electronics

Review

A Survey of Large Language Models: Evolution, Architectures, Adaptation, Benchmarking, Applications, Challenges, and Societal Implications

Seyed Mahmoud Sajjadi Mohammadabadi [1,2], Burak Cem Kara [1], Can Eyupoglu [1,3], Can Uzay [1], Mehmet Serkan Tosun [4] and Oktay Karakuş [5,*]

[1] Battle Born AI, Nevada Center for Applied Research, University of Nevada, Reno, NV 89557, USA; mahmoud.sajjadi@unr.edu (S.M.S.M.); burakcemkara@battleborn.ai (B.C.K.); can.eyupoglu@msu.edu.tr or caneyupoglu@gmail.com or caneyupoglu@battleborn.ai (C.E.); canuzay@battleborn.ai (C.U.)
[2] Department of Computer Science and Engineering, University of Nevada, Reno, NV 89557, USA
[3] Department of Computer Engineering, Turkish Air Force Academy, National Defence University, İstanbul 34149, Türkiye
[4] Department of Economics, University of Nevada, Reno, NV 89557, USA; tosun@unr.edu
[5] School of Computer Science and Informatics, Cardiff University, Cardiff CF24 4AG, UK
* Correspondence: karakuso@cardiff.ac.uk

Abstract

This survey provides an in-depth review of large language models (LLMs), highlighting the significant paradigm shift they represent in artificial intelligence. Our purpose is to consolidate state-of-the-art advances in LLM design, training, adaptation, evaluation, and application for both researchers and practitioners. To accomplish this, we trace the evolution of language models and describe core approaches, including parameter-efficient fine-tuning (PEFT). The methodology involves a thorough survey of real-world LLM applications across the scientific, engineering, healthcare, and creative sectors, coupled with a review of current benchmarks. Our findings indicate that high training and inference costs are shaping market structures, raising economic and labor concerns, while also underscoring a persistent need for human oversight in assessment. Key trends include the development of unified multimodal architectures capable of processing varied data inputs and the emergence of agentic systems that exhibit complex behaviors such as tool use and planning. We identify critical open problems, such as detectability, data contamination, generalization, and benchmark diversity. Ultimately, we conclude that overcoming these complex technical, economic, and social challenges necessitates collaborative advancements in adaptation, evaluation, infrastructure, and governance.

Keywords: large language models; transformer architectures; parameter-efficient fine-tuning; prompt engineering; multimodal models; LLM Benchmarks; inference cost; sector-wise applications; generative AI; economic impact

Academic Editor: Arkaitz Zubiaga

Received: 8 August 2025
Revised: 27 August 2025
Accepted: 27 August 2025
Published: 9 September 2025

Citation: Sajjadi Mohammadabadi, S.M.; Kara, B.C.; Eyupoglu, C.; Uzay, C.; Tosun, M.S.; Karakuş, O. A Survey of Large Language Models: Evolution, Architectures, Adaptation, Benchmarking, Applications, Challenges, and Societal Implications. *Electronics* **2025**, *14*, 3580. https://doi.org/10.3390/electronics14183580

1. Introduction

Large language models (LLMs) represent a paradigm shift in artificial intelligence, extending far beyond traditional natural language processing (NLP). Foundational models such as OpenAI's GPT series [1–3], Google's PaLM [4], and Meta's LLaMA [5] show novel abilities to understand, reason, and generate human language in fields ranging from software engineering and scientific discovery to education and creative arts. At their core, these capabilities highlight the power of self-supervised learning (SSL) at scale,

where models acquire deep, generalizable knowledge from petabytes of unlabeled text and multimodal data. The pace of innovation in the LLM landscape is extraordinary, exemplified by the rapid introduction of new architectures, training methodologies, and applications. This explosive growth, while exciting, has created a fragmented and complex body of knowledge that poses a challenge for researchers, practitioners, and policymakers to navigate. As LLMs become more deeply integrated into societal infrastructure, a clear and comprehensive understanding of their underlying technologies, capabilities, and limitations is not only beneficial but essential. This survey is motivated by the need to organize this rapidly evolving field and provide an accessible overview of the state-of-the-art. This includes not only their core linguistic capabilities but also their rapid evolution into autonomous agents capable of tool use and planning.

Objectives, Scope, and Methodology

The main objective of this survey is to offer a comprehensive and multifaceted overview of the LLM domain. It aims to equip both novice and experienced researchers with a solid foundation in key concepts while also highlighting the nuanced challenges and future directions that define the research frontier. To meet these goals, this survey provides a clear and structured overview of the LLM field through the following goals:

- Evolution of language models, reviewing the rise of Transformer-based architectures by tracing key innovations and paradigm shifts from early rule-based systems to modern foundation models (see Section 2).
- Establish a taxonomy of popular LLM architectures, including encoder-only, decoder-only, encoder-decoder (sequence-to-sequence), and multimodal models, detailing their design principles, capabilities, and typical use cases (see Section 3).
- Describe the core training and adaptation methodologies, including large-scale self-supervised pre-training, task-specific fine-tuning, and adaptation techniques such as reinforcement learning from human feedback (RLHF) and parameter-efficient fine-tuning (PEFT), supporting efficient and scalable deployment (see Section 4).
- Review benchmarks and evaluation methods used to assess model performance across tasks, including reasoning, factual correctness, robustness, and linguistic understanding (see Section 5).
- Survey real-world applications of LLMs across diverse domains—including scientific discovery, software engineering, healthcare, and the emergence of agentic AI as a key application area—where models act as reasoning engines for autonomous systems (see Section 6).
- Examine the economic implications of LLM development and deployment, including training and inference costs, infrastructure dependencies, labor market shifts, and growing inequalities in access and benefits (see Section 7).
- Highlight emerging challenges and open research questions, including hallucination, ethical risks, resource efficiency, and the broader societal impacts of LLM deployment (see Section 8).

2. Evolution of Language Modeling

Language modeling has transformed remarkably, from early rule-based systems to today's Transformer-based models. This progression reflects a series of paradigm shifts, from manually crafted symbolic rules to statistical models, then to deep neural networks, and ultimately to large-scale self-supervised training on vast corpora. The Transformer [6] architecture brought a breakthrough, enabling models to scale effectively and allowing models to perform well on many different tasks. As illustrated in Table 1 and Figure 1,

the development of language models spans four major eras, each distinguished by key innovations in how language is represented and learned.

Table 1. Progression Please confirm the alignment change. of Language Modeling Paradigms.

Timeline	Dominant Models	Key Strengths	Notable Limitations
Pre-1990s: Rule-Based	ELIZA [7], PARRY [8], A.L.I.C.E. [9], SHRDLU [10]	Simulates conversation via handcrafted rules; early human-computer interaction	No learning; brittle; poor generalization; no understanding; limited context
1990s–2000s: Statistical	n-gram [11], HMM [12], CRF [13]	Data-driven; foundational for early speech/MT; robust to noise	Fixed context (n); limited long-range dependencies; no semantics
2000s-2020s: Neural Networks	RNN [14], LSTM [15], GRU [16], Word2Vec [17], GloVe [18]	Learns distributed representations; models variable-length sequences	Sequential bottlenecks; poor parallelization; struggles with long-term context
Late 2010s–Present: Transformers	BERT [19], GPT series [1–3], DeepSeek [20] T5 [21], LLaMA [5], PaLM [4]	Scalable self-attention; contextual understanding; few/zero-shot ability; handles long-range dependencies	High computational cost; hallucination; bias; interpretability challenges

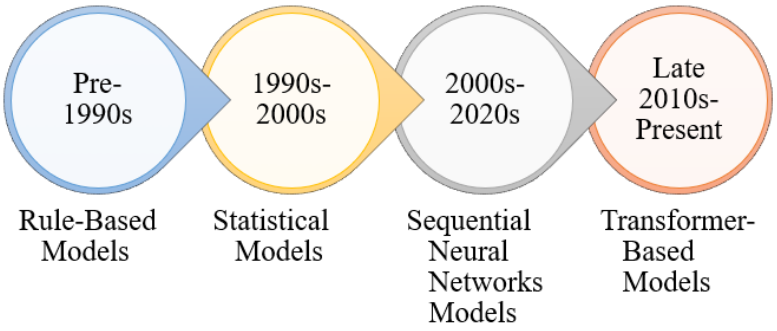

Figure 1. Timeline of language modeling evolution from rule-based systems to modern Transformer-based LLMs, highlighting major paradigm shifts and representative techniques.

2.1. Rule-Based Models (Pre–1990s)

Early NLP systems were primarily rule-based. Models such as ELIZA [7] and SHRDLU [10] relied on explicitly defined rules, pattern matching, and hardcoded grammars. These systems were capable of simulating structured dialogue but lacked genuine language comprehension and the ability to generalize, making them fragile and domain-limited.

2.2. Statistical Models (1990s–2000s)

The 1990s marked a shift toward statistical approaches, where language generation was modeled probabilistically. Techniques such as *n*-gram models [11], hidden Markov models (HMMs) [12], and conditional random fields (CRFs) [13] enabled more data-driven and robust methods, leading to advances in tasks like machine translation and speech recognition. Nonetheless, these models were constrained by limited context windows and an inability to capture deeper semantic or long-range dependencies.

2.3. Sequential Neural Language Models (2000s–2020s)

Deep learning dominated NLP throughout the 2010s. Recurrent neural networks (RNNs) [14], along with their variants such as LSTMs [15] and GRUs [16], enabled sequential processing and better context modeling. Embedding-based models like Word2Vec [17] and GloVe [18] introduced distributed word representations that captured semantic similarity in high-dimensional space. Despite these breakthroughs, early neural models had

limitations: they processed inputs sequentially, were hard to parallelize, and struggled with very long-term dependencies due to vanishing gradients.

2.4. Transformer-Based Models (Late 2010s–Present)

The advent of the Transformer architecture, with its attention mechanism, revolutionized NLP by allowing efficient, parallelizable modeling of long-range dependencies. Models such as BERT [19], GPT [1–3], T5 [21], PaLM [4], and LLaMA [5] pushed the boundaries of what language models could achieve. This approach enabled pre-training on large unlabeled data, followed by task-specific fine-tuning, with capabilities like few-shot learning. However, these models face challenges: they need a lot of computational power, sometimes generate incorrect information (hallucinate), and raise ethical concerns about bias, misuse, and transparency.

3. Model Architectures

The development of LLMs has followed a variety of architectural designs, each optimized for specific types of tasks and data modalities. As shown in Figure 2, the Transformer architecture introduced by Vaswani et al. [6] served as the foundation for most modern large language models. Table 2 further compares major LLM architectures, highlighting representative models and their typical use cases. Below, we highlight prominent model types, discussing their structural characteristics and typical use cases.

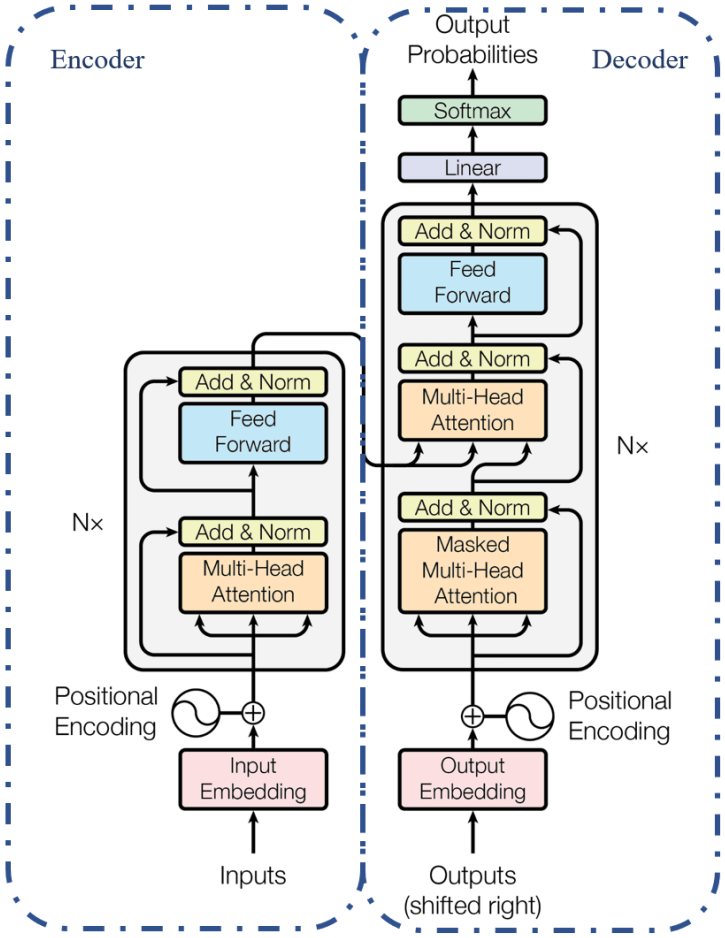

Figure 2. Overview of the Transformer architecture (encoder-decoder model) as introduced by Vaswani et al. [6].

Table 2. Comparison of Major LLM Architectures.

Architecture Type	Representative Models	Typical Use Cases
Encoder-Only	BERT [19], RoBERTa [22], ALBERT [23]	Text classification, NER, extractive QA, sentiment analysis
Decoder-Only	GPT-2/3/4 [2], LLaMA [5], PaLM [24], DeepSeek-V3 [20]	Text generation, dialogue systems, in-context learning
Encoder–Decoder (Seq2Seq)	T5 [21], BART [25]	Translation, summarization, abstractive QA, text rewriting
Multimodal	DeepSeek-VL [26], GPT-4o [27]	Image captioning, visual question answering, cross-modal retrieval

3.1. Decoder-Only Models

Decoder-only models, such as the GPT series (GPT-2, GPT-3, GPT-4) [1–3], PaLM [24], LLaMA [5], and DeepSeek-V3 [20], are autoregressive architectures using only the decoder blocks of the Transformer architecture. Their core operational principle is unidirectional context processing; they generate text token by token, from left to right. Each new token depends on the tokens generated before it. This works by using a masked self-attention mechanism that makes sure, when predicting the token at position i, the model only looks at tokens before position i. The standard self-supervised objective for these models is next token prediction (NTP), also named causal language modeling (CLM). Due to their inherent structure, these models excel at free-form text generation, dialogue systems, content creation, and any task requiring coherent and contextually aware linguistic output. Their proficiency in few-shot and zero-shot in-context learning directly results from this generative pre-training.

3.2. Encoder-Only Models

Encoder-only models, such as BERT (bidirectional encoder representations from Transformers) [19] and its variants like RoBERTa [22] and ALBERT [23], are composed exclusively of Transformer encoder blocks. Unlike their autoregressive counterparts, these models process the whole input sequence, allowing for deep bidirectional context understanding. The self-attention mechanism in encoders is not masked, meaning every token can attend to every other token in the sequence (both to its left and right). This makes them exceptionally well-suited for comprehension-based Natural Language Understanding (NLU) tasks such as text classification, sentiment analysis, named entity recognition (NER), and extractive question answering. Their main pre-training task is usually Masked Language Modeling (MLM), where some input tokens are randomly hidden, and the model learns to predict these hidden tokens using the surrounding visible context.

3.3. Sequence-to-Sequence Models

Sequence-to-sequence (seq-to-seq) models, including T5 (text-to-text transfer Transformer) [21] and BART (bidirectional and auto-regressive Transformers) [25], utilize the complete Transformer architecture, comprising both an encoder and a decoder stack. The encoder processes the input sequence to build a rich, contextualized representation, which is then passed to the decoder to generate the target output sequence. This architecture is highly effective for transformation tasks that map an input sequence to a different output sequence. Prominent applications include machine translation, text summarization (where a long document is mapped to a shorter summary), and abstractive question answering. Their self-supervised pre-training often involves denoising objectives; for instance, T5 is trained by hiding parts of the input text and teaching the model to restore the original, complete text.

3.4. Multimodal Models

Multimodal models, such as DeepSeek-VL [26] and GPT-4o [27], are designed to handle and integrate multiple data modalities like text, images, and sometimes audio or video. These architectures extend the Transformer backbone to process and fuse heterogeneous inputs, enabling tasks such as cross-modal retrieval, visual question answering, and image captioning.

3.5. Mixture of Experts (MoE) Models

Another innovation for scaling LLMs efficiently is the Mixture of Experts (MoE) architecture [28]. Unlike traditional dense models, where all parameters are used for every input, MoE models consist of numerous smaller "expert" sub-networks and a router network. For any given input token, the router dynamically selects a small subset of experts to process it. This conditional computation allows MoE models to have a massive number of parameters while keeping the inference cost low, as only a fraction of the model is activated for each token. Prominent examples like Mistral's Mixtral 8x7B [29] and Google's Gemini [30] models leverage this architecture to achieve state-of-the-art performance with significantly reduced computational overhead compared to dense models of similar size [31].

4. Training and Adaptation

The lifecycle of an LLM does not end with its initial pre-training. LLMs undergo several stages of adaptation to show their full potential and tailor their vast, generalized knowledge to specific applications and user expectations. This section details the core training paradigms and methodologies that transform foundational models into specialized, efficient, and aligned tools. We first outline the stages of pre-training, fine-tuning, and prompt engineering, then cover alignment and efficiency strategies, focusing on parameter-efficient fine-tuning (PEFT) and transfer learning. An overview of these adaptation methods is illustrated in Figure 3.

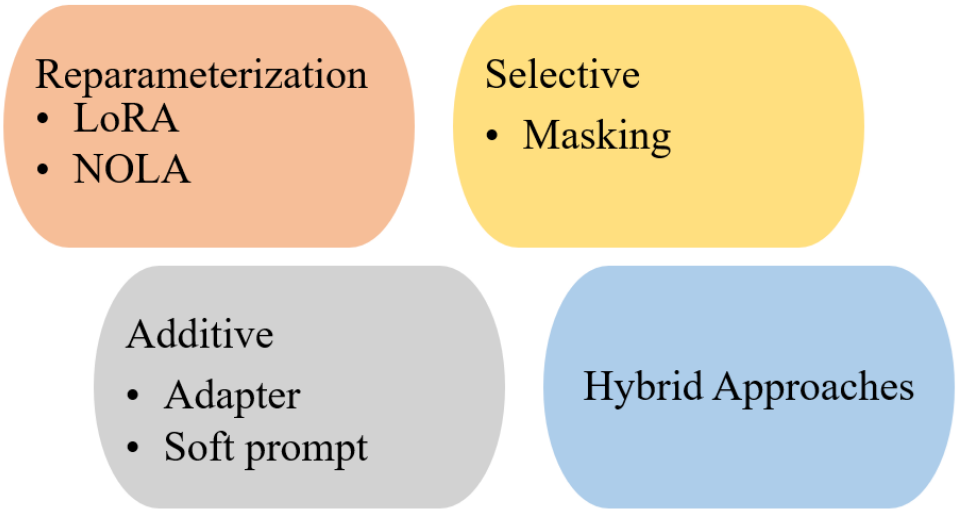

Figure 3. Overview of adaptation methods for large language models.

4.1. Pre-Training

Pre-training is a self-supervised phase in which the model learns general language patterns and world knowledge from large-scale, unlabeled text corpora. This one-time process uses objectives such as NTP or MLM and forms the foundational backbone of modern large language models.

4.1.1. Fine-Tuning

Fine-tuning updates all model parameters using supervised learning on a task-specific dataset. While effective, it is costly for large LLMs, requiring substantial compute and storage for each task, motivating more efficient alternatives.

4.1.2. Prompt Engineering and In-Context Learning

Prompt engineering guides LLM behavior by crafting inputs without changing model weights. It uses instructions and examples (few-shot prompting) to leverage in-context learning [3]. Though efficient and flexible, its performance depends on prompt design and context window limits.

4.1.3. Instruction Tuning

Instruction tuning is a specialized form of fine-tuning to enhance an LLM's ability to follow natural language instructions and generalize to unseen tasks. This is carried out by training the model on a large, diverse set of functions presented in instructional formats (e.g., "Summarize the following text," "Translate this sentence to French," "Write a Python function that computes the factorial"). Seminal models such as FLAN [32] and T0 [33] have shown that instruction tuning significantly improves zero-shot performance across a wide range of tasks, making models more usable and steerable.

4.1.4. Reinforcement Learning from Human Feedback (RLHF)

RLHF is a powerful technique for aligning LLM outputs with complex, subjective human preferences, such as helpfulness, honesty, and harmlessness. It was notably used to train models like InstructGPT and ChatGPT-3 [34]. The RLHF process typically involves three steps: (1) Supervised fine-tuning (SFT), where a pre-trained model is fine-tuned on a high-quality dataset of human-written examples to establish a baseline behavior. (2) Reward model training, where a separate model learns to score outputs based on a dataset of human-ranked model responses. (3) Reinforcement learning optimization, where the SFT model is further improved using an RL algorithm, which uses the reward model's scores to guide the LLM toward generating outputs that align with human preferences.

4.2. Parameter-Efficient Fine-Tuning

Parameter-efficient fine-tuning (PEFT) methods have emerged as a solution to the prohibitive costs of full fine-tuning. PEFT techniques aim to adapt a pre-trained LLM by updating only a small fraction of its parameters while keeping the vast majority of the original model weights frozen. This dramatically reduces computational and storage costs, making it feasible to adapt a single pre-trained model to multiple tasks. These methods vary significantly in their approach, leading to important trade-offs between the number of trainable parameters, impact on inference speed, memory usage, and downstream task performance, as summarized in Table 3. The choice of PEFT strategy often depends on the specific constraints of the application, such as available computing power and desired generalization. PEFT methods can be broadly categorized as follows.

Table 3. Comparison of Popular Parameter-Efficient Fine-Tuning (PEFT) Methods.

PEFT Method	Core Mechanism	Key Characteristics	Trainable Params
Adapters [35]	Injects small, trainable "adapter" modules between frozen Transformer layers.	Trade-off: Effective generalization but adds inference latency due to new modules. Requires architectural modification.	Low (∼0.1–5%)
Prompt Tuning [36]	Prepends learnable "soft prompt" embeddings to the input sequence.	Trade-off: Highest parameter efficiency but performance can be less stable and sensitive to prompt length. No inference latency.	Very Low (<0.1%)
Prefix-Tuning [37]	Prepends learnable prefixes to the hidden states of each Transformer layer.	More expressive and stable than prompt tuning, but slightly more complex. No added inference latency during generation.	Very Low (<0.1%)
LoRA [38]	Freezes base model weights and injects trainable low-rank matrices to approximate weight updates.	Trade-off: Balances high downstream performance with efficiency. No inference latency as matrices can be merged. Highly effective and widely adopted.	Low (∼0.1–1%)

4.2.1. Adapter-Based Methods

This approach involves injecting small, trainable neural network modules, known as "adapters," within the layers of the pre-trained Transformer. During fine-tuning, only the parameters of these newly added adapters are trained, while the original LLM weights remain frozen. Adapters are typically designed as bottleneck architectures, with a down-projection, a non-linearity, and an up-projection, significantly reducing the number of trainable parameters compared to the main model [35].

4.2.2. Prompt-Based Methods (Soft Prompts)

Prompt-based Methods differ from discrete prompt engineering by learning continuous task-specific vectors, or soft prompts, that are prepended to the model's input. In *prompt tuning*, a small set of learnable embedding vectors is added to the input token embeddings; only these vectors are updated during training [36]. Prefix tuning, a more expressive variant, prepends continuous vectors to the hidden states of each Transformer layer, enabling more direct control over internal activations [37].

4.2.3. Reparameterization-Based Methods

Low-rank adaptation (LoRA) is one of the most popular and effective PEFT techniques [38]. It assumes that weight updates during adaptation lie in a low-rank subspace. Instead of updating the full weight matrix W, LoRA introduces a trainable low-rank decomposition $\Delta W = BA$, where A and B are much smaller matrices and $r \ll \min(d_{\text{in}}, d_{\text{out}})$. The base weights W are frozen, and only A and B are trained. At inference, the update is merged as $W' = W + BA$, adding no latency. The trainable parameters are a small fraction of W. QLoRA improves efficiency further by quantizing the base model to 4-bit and applying LoRA on top [39].

5. Benchmarking and Evaluation

The effectiveness of an LLM's training or adaptation process ultimately depends on how well its capabilities are assessed. Once a model has been pre-trained and fine-tuned, it must be systematically evaluated to verify its performance across a range of tasks. Robust

benchmarking not only guides the model development process but also helps identify limitations, biases, or regressions that may have emerged during the adaptation process. Thus, evaluation is a critical step in the LLM lifecycle, closely tied to its development and deployment.

5.1. Benchmarking

Benchmarking is a foundational aspect of evaluating LLMs, as it provides standardized datasets and metrics to quantify performance across tasks, domains, and model scales. With the growing deployment of LLMs in real-world applications, benchmarking has evolved from static accuracy measurements to multidimensional, general-purpose evaluations. Several prominent benchmarks have been developed to assess capabilities such as factual recall, reasoning, linguistic understanding, and robustness. In the subsections below, we summarize four major benchmarks, each contributing unique insights into LLM behavior. While these are often used together in evaluation pipelines, their emphases and methodologies differ significantly. A comparison of major LLM benchmarks is provided in Table 4.

Table 4. Comparison of Major LLM Benchmarks.

Benchmark	Focus	Dataset Size/Scope
MMLU [40]	Academic QA and reasoning across disciplines	57 subjects, ∼15K Multiple-Choice Questions
BIG-bench [41]	Emergent abilities and generalization (e.g., humor, ethics, logic)	200+ tasks, community-contributed
SuperGLUE [42]	Challenging NLU tasks (coreference, inference, etc.)	8 tasks (e.g., RTE, WSC, COPA)
HELM [43]	Multi-dimensional LLM evaluation (accuracy, fairness, robustness, bias)	42 scenarios × 8 metrics × 30+ models

In the subsections below, we summarize four major benchmarks, each contributing unique insights into LLM behavior. While these are often used together in evaluation pipelines, their emphases and methodologies differ significantly.

5.1.1. MMLU

The Massive Multitask Language Understanding (MMLU) benchmark evaluates knowledge and reasoning across 57 tasks from diverse academic and professional fields, including mathematics, medicine, history, and law. Designed for few-shot evaluation, MMLU measures how well LLMs generalize to unseen subject areas, making it a critical tool for gauging real-world utility beyond training distributions [40].

5.1.2. BIG-Bench

BIG-bench (Beyond the Imitation Game) is a collaborative, community-driven benchmark suite consisting of over 200 tasks. These tasks are designed to evaluate emergent abilities of language models, such as humor understanding, arithmetic reasoning, and moral judgment. BIG-bench emphasizes the identification of novel generalization capabilities that arise only in models of sufficient scale [41].

5.1.3. SuperGLUE

SuperGLUE is an advanced successor to the GLUE benchmark, targeting more challenging language understanding problems, such as coreference resolution, causal reasoning, and multisentence inference. It includes human performance baselines and provides fine-

grained diagnostics for task-specific errors, making it suitable for evaluating fine-tuned or general-purpose LLMs [42].

5.1.4. HELM

Unlike task-specific benchmarks, the Holistic Evaluation of Language Models (HELM) framework takes a meta-evaluation approach by comparing multiple LLMs across a variety of dimensions—accuracy, calibration, robustness, fairness, bias, toxicity, and efficiency. HELM highlights trade-offs in model design and deployment, promoting more transparent and holistic assessment across use cases and deployment scenarios [43].

5.2. Evaluation

Evaluating the quality of generated text from NLP models is critical for tasks such as summarization, translation, and other text-generation applications. This section reviews commonly used automatic metrics, ROUGE and BLEU, with their definitions, computations, strengths, and limitations.

5.2.1. ROUGE

ROUGE (Recall-Oriented Understudy for Gisting Evaluation) is a family of recall-focused metrics primarily used for automatic summarization evaluation but also applicable to other generation tasks where coverage of reference content is important [44]. ROUGE metrics compare the overlap between a reference text (often one or more human-generated references) and a system-generated text. Higher scores indicate greater overlap and, presumably, better content coverage.

Each ROUGE variant reports *Precision* (proportion of overlapping units in the generated text), *Recall* (proportion of overlapping units in the reference text), and F_1 (their harmonic mean). ROUGE's definitions of precision and recall operate over counts of overlapping textual units rather than binary classification counts. The main ROUGE variants include

- ROUGE-N: Evaluates the overlap of n-grams between generated and reference texts, counting matched n-grams (with frequency clipping). Commonly used for unigrams (ROUGE-1) and bigrams (ROUGE-2), it emphasizes recall, specifically how many reference n-grams are covered, making it suitable for summarization evaluation.
- ROUGE-L: Uses the longest common subsequence (LCS) between generated and reference texts to capture in-sequence overlap without requiring contiguous matches. It computes precision and recall over LCS length (and often their harmonic mean), with a sentence-level variant (ROUGE-Lsum) for multisentence inputs.
- ROUGE-S (Skip-Bigram): Matches word pairs in order but not necessarily adjacent, allowing more flexible overlap detection than strict n-grams. It counts skip-bigram matches to assess loosely ordered content overlap, though it remains surface-based without deeper semantic matching.

5.2.2. BLEU

BLEU (Bilingual Evaluation Understudy) is a precision-oriented metric for machine translation that measures clipped n-gram overlap with reference translations and applies a brevity penalty to discourage overly short outputs [45]. While effective for translation by penalizing extraneous content, BLEU's reliance on exact n-gram matches can limit its correlation with human judgments in tasks with high lexical variability.

Different metrics suit different tasks. ROUGE is well-suited for summarization due to its emphasis on recall and content coverage. BLEU, with its precision focus and brevity penalty, remains standard for machine translation. BLEURT and similar learned metrics are

valuable when capturing paraphrasing, fluency, and semantic nuance is essential—though they often require greater computational resources. Using multiple human references improves evaluation reliability by accounting for acceptable variation in outputs. Consistent preprocessing (e.g., tokenization, casing, punctuation) is also critical, as minor inconsistencies can skew scores. Since metric values vary by dataset and domain, relative comparisons between models are generally more informative than absolute numbers. Ultimately, automatic metrics should be complemented by human evaluation, especially in high-stakes or user-facing applications where coherence, factuality, and usefulness are paramount.

6. Applications of LLMs

LLMs have rapidly evolved from experimental systems within research into powerful technologies transforming various industries. Their multimodal reasoning, zero- and few-shot learning, and generative capabilities support numerous practical applications across domains (Figure 4). Table 5 summarizes the major sectors where LLMs are being deployed, highlighting representative applications and examples.

Figure 4. Overview of LLM applications across diverse sectors.

Table 5. Applications of Large Language Models Across Sectors.

Category	Sector/Use Cases	Description/Example Functions	References
STEM & Research	Scientific Research: hypothesis generation, experiment design, writing	LLMs like Elicit and SciBot support knowledge synthesis, planning, and scientific writing.	[1,46–50]
	Healthcare & Life Sciences: scribing, drug discovery, literature review	LLMs generate EHR notes, simulate molecular interactions, and summarize biomedical texts.	[51–54]
	Software Engineering: code generation, debugging, HDL design, and privacy-aware analytics	LLMs like Copilot and CodeLLaMA assist in programming and hardware logic synthesis.	[55–60]

Table 5. *Cont.*

Category	Sector/Use Cases	Description/Example Functions	References
Enterprise & Business	Finance & Banking: fraud detection, chatbots, reporting	Analyzes transactions, powers financial assistants, and automates compliance summaries.	[61–63]
	Manufacturing & Supply Chain: forecasting, log analysis, training	Forecast demand, interpret logs, and support engineering education via LLM-based tutors.	[64–66]
	Legal & Regulatory: legal search, contracts, compliance monitoring	Used in tools like CoCounsel and Harvey AI for legal reasoning and risk detection.	[67,68]
Creative & Social Domains	Creative Industries: writing, art/music, design ideation	LLMs power story generation, compose music (e.g., MuseNet), and assist with architecture sketches.	[69–73]
	Education: conversational tutoring, engagement	Supports inclusive, always-available learning environments with natural interaction.	[74,75]
	Training: content customization, feedback, real-time assessment	Used in platforms like Khanmigo and Duolingo AI for tailored learning experiences and skill development.	[74–77]
Autonomous Systems	LLM Agents: task chaining, API interaction, digital automation	Auto-GPT and LangChain enable agents to reason, use tools, and automate workflows.	[78,79]

6.1. Software Engineering and Design

LLMs play a critical role in modern software development and hardware design:

- Code Generation: Tools like GitHub Copilot and CodeLLaMA suggest or generate code from natural language [55,56].
- Debugging and Refactoring: LLMs assist developers by offering bug fixes and code improvements [57].
- EDA and HDL Translation: In chip design, LLMs automate the generation of HDL and streamline design verification [58,59].

6.2. Healthcare and Life Sciences

LLMs are increasingly integrated into healthcare settings for administrative automation, clinical decision-making, and research support. Examples include:

- EHR Data Synthesis: LLMs excel at analyzing complex and unstructured Electronic Health Records (EHRs). They can extract information and generate concise patient summaries, with performance that can outperform human experts [80].
- Diagnostic Assistance: These models can serve as powerful diagnostic aids by processing clinical notes and patient histories to suggest potential differential diagnoses in fields such as radiology [81]. Specialized models, such as Med-PaLM 2, have demonstrated expert-level performance on medical competency exams, underscoring their potential to augment clinical workflows [82].
- Personalized Treatment Planning: By aligning a patient's clinical profile with the latest evidence-based guidelines and medical literature, LLMs can assist in providing personalized treatment plans [83,84].

- Medical Scribing and Documentation: LLMs transcribe doctor–patient conversations into structured EHR notes, reducing clerical workload and allowing clinicians to focus more on patient care [51,52].
- Drug Discovery: They assist in identifying drug targets and predicting molecular interactions using biomedical data [53,54].
- Literature Synthesis: LLMs extract and summarize findings from large corpora of medical papers, enabling faster insights [51].

6.3. Finance and Banking

The financial sector applies LLMs for automation, analysis, and risk mitigation:

- Investment Analysis and Strategy: LLMs perform sentiment analysis on financial news, social media, and earnings reports to identify market trends and inform investment strategies. These models process vast amounts of unstructured text data in real-time to provide quantitative insights that support algorithmic trading and portfolio management [85].
- Compliance and Fraud Detection: Models analyze transactions and communications for anomalies indicative of fraud or regulatory violations [61].
- Chatbots and Virtual Assistants: Customer service is enhanced by LLMs that provide 24/7 support, reducing operational costs [62].
- Financial Reporting: LLMs generate and summarize reports, accelerating analyst workflows [63].

6.4. Manufacturing and Supply Chain

LLMs optimize complex engineering and logistics systems:

- Forecasting and Optimization: Demand prediction and supply chain optimization benefit from LLM-generated insights [64].
- Quality Control: Natural language interfaces aid in interpreting maintenance logs or sensor data [65].
- Engineering Education: LLMs provide customized support and tutoring for technical training [66].

6.5. Scientific Research and Discovery

LLMs are accelerating scientific progress:

- Hypothesis Generation and Literature Review: LLMs rapidly synthesize findings from thousands of papers [46]. For instance, tools like Elicit [47] and Semantic Scholar [86] leverage Transformer models to extract key claims, compare methodologies, and trace citations across thousands of papers in seconds. More recent models like Google's Gemini [30] can perform deep research by synthesizing information across multiple documents, analyzing data, and generating novel hypotheses, effectively acting as AI research assistants [87].
- Experiment Design and Analysis: Beyond understanding prior work, LLMs can support the planning and interpretation of experiments. For example, models like ChatGPT [1] and SciPIP [88] have been used to suggest experimental conditions, recommend statistical techniques, and simulate expected outcomes based on prior data [46]. In computational chemistry, LLMs have even been integrated into pipelines to optimize reaction conditions and propose novel molecular structures [48], and new frameworks are using them to make the entire automated machine learning (AutoML) process more explainable and user-friendly through natural language [89].
- Scientific Writing: LLMs assist researchers in drafting abstracts, summarizing findings, and organizing research manuscripts in line with academic standards. Tools such

as PaperPal and Writefull utilize LLMs to enhance clarity, suggest citations, and correct grammar in real time. In addition, citation-aware models like SciBot [49] can automatically insert references and generate BibTeX entries based on context.

6.6. Education and Corporate Training

LLMs are transforming how knowledge is delivered and assessed:

- Personalized Learning: LLMs dynamically tailor educational content to match a learner's proficiency, interests, and preferred learning style. For instance, platforms like Khanmigo (by Khan Academy) use GPT-based models to deliver adaptive math explanations for students at varying levels [74].
- Assessment and Feedback: LLMs can evaluate student responses, provide constructive feedback, and even detect misconceptions. Tools such as Gradescope AI integrate LLMs to automate short-answer grading and generate formative feedback, freeing instructors to focus on higher-level instruction [76,77].
- Virtual Tutoring: LLMs act as intelligent tutors that offer instant, 24/7 support across a wide range of topics. For example, Duolingo's GPT-4-powered AI tutor provides personalized conversational practice in language learning, correcting errors and explaining grammar contextually [75].

6.7. Creative and Content Industries

From text to music, LLMs are reshaping creative workflows and augmenting human expression across multiple domains.

- Writing and Journalism: LLMs like GPT-4 are used by outlets such as BuzzFeed to generate article drafts, headlines, and marketing copy [69]. These models accelerate content creation while allowing human editors to refine tone and accuracy.
- Sports Media and Entertainment: Domain-specific applications are emerging that showcase how LLMs can augment commentary, analysis, and fan engagement. Data-driven football match commentaries that combine real-time statistics with fluent narrative structures help enrich live sports coverage [90]. Similarly, natural language explanations of machine learning models of footballing actions bridge the gap between complex analytics and interpretable insights for coaches and analysts [91].
- Visual and Performing Arts: Multimodal systems such as OpenAI's MuseNet [71] and DALL-E [72] generate music and artwork from textual prompts, enabling new forms of artistic experimentation [70]. Artists use these tools for inspiration, rapid prototyping, or hybrid collaborations.
- Design and Architecture: Tools like Autodesk Forma integrate LLMs and generative models to assist with early-stage ideation and layout generation [73].

6.8. Legal and Regulatory Sectors

LLMs are increasingly being adopted to streamline legal workflows, reduce costs, and improve access to legal resources.

- Legal Research: LLMs like Casetext's CoCounsel use GPT-4 to retrieve relevant statutes, case law, and legal summaries within seconds [67].
- Contract Review: Tools such as Harvey AI assist law firms by analyzing contracts, flagging potential risks, and summarizing clauses in plain language [68].

6.9. Autonomous AI Agents

Emerging AI agents powered by LLMs are capable of reasoning over tasks, interacting with tools, and autonomously executing complex digital operations.

- Chain Tasks and APIs: LLM agents like Auto-GPT and LangChain-based systems can autonomously decompose goals into subtasks and interact with external APIs, file systems, or browsers to complete them [78,79].

7. Economic Implications of LLM Development and Deployment

The rapid advancement and widespread adoption of LLMs are generating far-reaching economic impacts. These models are reshaping industries, creating new markets, and introducing substantial economic challenges [92]. The LLM ecosystem is shaped by three primary cost centers: the massive, front-loaded capital required for pre-training; the ongoing operational expense of inference; and the significant, often-underestimated human capital investment in data curation and alignment [93] (see Figure 5). As shown in Table 6, training costs for frontier models have escalated into the hundreds of millions of dollars, reinforcing high barriers to entry and driving market concentration.

This section examines the key economic dimensions of LLM development and deployment, beginning with their foundational cost structure and extending to their broader impacts on labor markets, fiscal policy, and economic inequality.

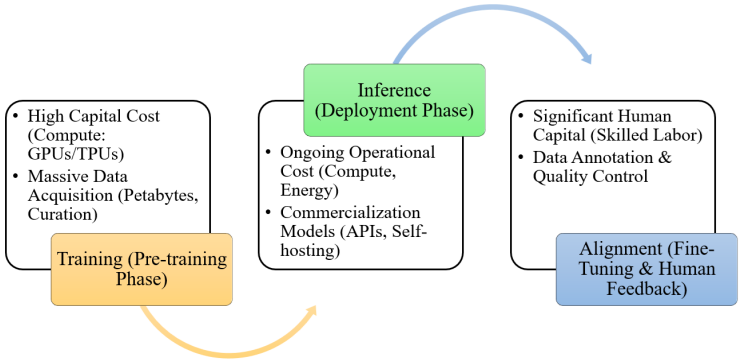

Figure 5. LLM lifecycle: Training, Inference, and Alignment.

Table 6. Illustrative Cost Estimates for LLM Training and Inference.

Item	Estimated Cost/Metric
Training GPT-3 (175B)	∼$4.6M USD (2020) [94]
Training PaLM (540B)	∼$3∼12M USD (2022) [95]
Training Gemini 1.0 Ultra	∼$192M USD (2025) [96]
Inference Cost per 1M tokens (OpenAI API, June 2025)	$0.10 (GPT-4.1 nano input) to $8.00 (GPT-4.1 output) [97]

7.1. The Foundational Costs: From Training to Deployment

The economics of LLMs are anchored by immense costs spanning the entire model lifecycle, from initial training to final deployment (see Table 7). These expenditures concentrate development in a handful of well-resourced organizations and create technical and economic trade-offs at each stage.

- Training Costs: The initial pre-training of a foundation model is the most expensive phase, representing a significant front-loaded capital expenditure. It requires massive computational power, typically involving thousands of high-end GPUs or TPUs running continuously for weeks or months. The costs have escalated dramatically; while GPT-2 (1.5 billion parameters, 2019) cost an estimated $50,000 to train, Google's PaLM (540 billion parameters, 2022) is estimated to have cost around $8 million, and the Megatron-Turing NLG 530B model over $11.35 million [98]. These costs are driven by the sheer scale of the model (billions or trillions of parameters) and the vast

datasets (trillions of tokens) required to achieve state-of-the-art performance. This has concentrated development in industry, which produced 32 significant machine learning models in 2022 compared to just three from academia.

- Inference Costs: While training is a formidable one-time cost, inference—the process of using a trained model to generate outputs—is a persistent operational expense that can cumulatively surpass the initial training cost for widely used services. The core economic challenge is balancing the conflicting demands of latency and throughput. For example, an interactive, low-latency configuration for PaLM 540B achieves a Model FLOPS Utilization (MFU) of only 14%, while a high-throughput configuration reaches 76% MFU, a five-fold difference in computational efficiency and cost. This is rooted in technical bottlenecks like the massive memory footprint of the model weights and the KV cache, which can total 3 TB for a 540B parameter model, and the inherently sequential nature of autoregressive decoding that limits parallelism [99]. Optimizing inference efficiency through techniques like model quantization (e.g., using INT8 weights reduced PaLM's per-token latency by 23%), multiquery attention, and specialized hardware is a critical area of research and economic concern [99].
- Data Acquisition and Curation: While much of the data used for pre-training is scraped from the public web (e.g., Common Crawl), creating high-quality, clean, and diverse datasets is a significant undertaking. Furthermore, the data required for alignment stages like supervised fine-tuning (SFT) and reinforcement learning from human feedback (RLHF) represents a substantial and often underestimated "hidden cost" driven by expensive, high-skill human labor. This phase requires thousands of hours of work from skilled labelers to generate demonstrations and rank model outputs to create preference datasets. These human-powered data generation efforts can add millions of dollars to the total development cost, an expense not captured in compute-based cost estimates like the $8 million figure for PaLM [98]. This human capital investment is a critical barrier to entry and a key component of a model's total cost of ownership.
- Hardware Dependency: The development of LLMs has been largely dependent on the availability of powerful GPUs, with NVIDIA commanding a dominant market share [100]. This has created a hardware bottleneck where access to cutting-edge accelerators is a primary determinant of an organization's ability to compete at the frontier of AI research.
- Cloud Infrastructure Dominance: Major cloud providers such as Amazon Web Services (AWS), Microsoft Azure, and Google Cloud Platform (GCP) are central to the LLM ecosystem. They provide the scalable, on-demand computing infrastructure necessary for both training and hosting LLMs. Strategic partnerships, such as Microsoft's investment in OpenAI, highlight how cloud providers are positioning themselves as indispensable platforms for the AI economy, capturing a significant portion of the value generated by LLM applications [101].
- Scalability and Deployment Trade-offs: Organizations face a critical decision between using third-party LLM APIs (e.g., OpenAI, Anthropic) and deploying their models (whether open-source or custom-built). Using APIs offers lower upfront costs and easier access but can lead to high long-term operational expenses and concerns over data privacy and model control. Self-hosting provides more control but requires significant investment in infrastructure and expertise. This trade-off is a central economic consideration for businesses integrating LLMs into their operations.

The economics of LLMs are inextricably linked to the underlying hardware and cloud infrastructure, a market dominated by a few key players.

Table 7. Breakdown of Economic Costs in LLM Development. MFU: Model FLOPS Utilization.

Cost Category	**Main Drivers**	**Key Technical Bottlenecks**	**Example Estimate**
Training (Pre-training)	Compute, GPU/TPU clusters, massive datasets	Model size scaling, training FLOPs, hardware availability	$8 M (PaLM 540B)
Inference (Deployment)	Continuous compute, energy, latency constraints	Memory bandwidth, KV cache size, parallelism limits	29 ms/token @ 76% MFU
Data Curation & Alignment	Human labor, annotation costs, quality control	RLHF ranking, SFT prompt generation, skilled reviewers	Millions USD

7.2. Market Consolidation and Commercialization

The high cost and technical complexity of developing frontier foundation models have concentrated power in the hands of a few dominant technology firms. This growing consolidation raises concerns about unequal access to advanced AI capabilities, reduced competition, and the broader economic consequences of an increasingly centralized LLM ecosystem [102]. As shown in Table 8, the economic structure of the LLM market reflects deep imbalances in capital, compute, and data access, reinforcing barriers for smaller actors and exacerbating inequality.

- Market Concentration: The development of state-of-the-art LLMs—such as GPT-4, Gemini, and Claude—is currently viable only for a small group of corporations, including Google, OpenAI (partnered with Microsoft), Meta, and Anthropic, who possess the necessary capital, proprietary data, and large-scale compute infrastructure [98,103,104]. This concentration of model development capabilities has sparked growing concerns over an emerging "AI oligopoly," in which a few firms dominate foundational AI technologies, limit open innovation, and shape the trajectory of the LLM ecosystem to serve proprietary interests [104].
- Commercialization and Access: These dominant firms primarily commercialize LLMs through usage-based APIs, which offer high performance but at costs often unaffordable for smaller businesses. In contrast, the open-source ecosystem (e.g., LLaMA, Mistral) provides alternatives, but these require in-house expertise and infrastructure [5]. Small and medium-sized enterprises (SMEs) face critical barriers—including limited budgets, talent shortages, and lack of cloud resources—that hinder their ability to adopt AI effectively [105–107]. As a result, there is a growing productivity gap between large firms rapidly scaling AI and smaller businesses struggling to compete [108].
- Economic Inequality and the Global AI Divide: The uneven diffusion of LLM benefits could intensify economic inequality [109]. Domestically, workers, firms, and regions with access to advanced AI tools may gain disproportionate advantages in productivity and profitability. Internationally, countries with limited access to AI development infrastructure risk falling further behind economically and technologically, exacerbating global inequalities [110].
- The Open-Source Ecosystem as a Counterbalance: In response to the market concentration driven by high development costs, the open-source community has emerged as a powerful force for democratizing AI. Pioneered by models like Meta's LLaMA series [5,111,112] and further advanced by organizations like EleutherAI [113], the movement provides access to high-performance foundation models with permissive licenses. This enables researchers, startups, and smaller enterprises to innovate, conduct

research, and build applications without being locked into proprietary API ecosystems. Open-source models foster transparency, enable security auditing, and allow for deep customization and fine-tuning on private data, capabilities often restricted by commercial vendors. While this approach significantly lowers the barrier to entry, it still requires substantial in-house computational resources and expertise to effectively deploy and maintain these models [114].

Table 8. Structural Drivers of Inequality in the LLM Economy.

Factor	Impact on Market Dynamics	Barriers for SMEs and Global South	Reference(s)
Market Consolidation	AI capabilities concentrated in a few tech giants	High entry cost excludes academia, small firms	[98,104]
API Commercialization	Usage-based pricing favors large-scale customers	Per-token cost unsustainable for startups/NGOs	[105,108]
Infrastructure Lock-in	Cloud platforms vertically integrate compute and model access	Self-hosting requires GPU access, engineering talent	[100,101]
Global Access Divide	Uneven distribution of AI benefits	Limited infrastructure, talent pipeline, and compute funding	[109,110]

7.3. Labor Market Disruption and Socioeconomic Inequality

LLMs are poised to cause significant shifts in the labor market, acting as both a transformative and destructive technology [115]. While they augment human capabilities, they also threaten to automate cognitive tasks, with complex and potentially divergent outcomes for wage and wealth inequality [116].

In many professional domains, LLMs are being deployed as "co-pilots" or assistants that enhance human productivity [117]. Programmers use tools like GitHub Copilot to write code more efficiently, writers use LLMs for brainstorming and drafting, and analysts leverage them for summarizing complex documents. Studies suggest these tools can yield substantial productivity gains, particularly for less-experienced workers [118]. Conversely, LLMs' advanced capabilities threaten to automate tasks previously considered exclusive to human cognition [119]. Roles involving routine, text-based work—such as customer service, data entry, and paralegal support—are especially vulnerable to displacement [120]. The long-term economic impact will depend on the rate of automation versus the creation of new roles that emerge to develop, manage, and collaborate with AI systems. The ultimate distributional consequences of LLM deployment are shaped by who controls the technology and how it is governed [121]. Deeper analysis reveals several compounding socioeconomic factors:

- Wage and Skill Polarization: The integration of LLMs may exacerbate wage inequality. Workers with skills complementary to AI (e.g., prompt engineering, AI ethics, system integration) may see wage increases, while those performing tasks easily automated may face downward wage pressure [122]. This necessitates broad societal efforts focused on reskilling and upskilling the workforce to adapt to an AI-driven economy [110].
- Wage vs. Wealth Inequality: AI adoption could have opposing effects on inequality. A calibrated task-based model using UK household data suggests that while AI may reduce wage inequality by displacing some high-income workers, it could substantially increase wealth inequality. This occurs as capital owners and those whose productivity is complemented by AI capture a larger share of economic gains, highlighting a

difficult trade-off for policymakers between fostering growth and managing wealth disparities [123].

- Demographic and Fiscal Pressures: The economic impacts of LLMs intersect with major demographic trends. In aging high-income economies, AI may compensate for shrinking workforces but could also reduce momentum for immigration policies that support fiscal stability [124]. As modeled by Tosun, demographic shifts directly influence public spending on education and human capital. Failure to adapt fiscal policy could amplify intergenerational pressures and undercut the public investments needed to prepare the workforce for an AI-driven economy [125].
- Population Aging: High-income economies experiencing demographic decline may increasingly rely on LLMs to sustain productivity. However, without inclusive reskilling initiatives and adaptive migration policies, these technologies risk shrinking the tax base, amplifying intergenerational fiscal pressures, and undermining public investment in education [126].
- Geographic Disparities: LLMs offer the potential to revitalize rural and underserved areas through applications like telehealth and remote education. However, this promise is contingent on equitable access to broadband infrastructure and local training, without which AI could worsen the rural-urban economic divide [127].

Ultimately, the historical record suggests that technological advances, AI included, do not yield equitable outcomes by default. As West highlights in his Brookings commentary [128], AI has the potential to exacerbate income inequality by displacing mid-skill jobs and concentrating economic gains among capital owners and advanced tech workers. Without deliberate redistribution mechanisms and inclusive system design, LLM adoption may deepen existing structural divides. More broadly, West warns that without appropriate safeguards, the productivity gains enabled by AI are likely to accrue to those with control over capital and digital infrastructure. Acemoglu's analysis [129] similarly emphasizes that unregulated innovation tends to reinforce rather than reduce socioeconomic disparities. Complementing these perspectives, Acemoglu and Johnson argue that it is the institutional context—who governs and controls the technology—that ultimately shapes whether innovation expands opportunity or entrenches elite dominance [121].

8. Recent Trends and Open Issues

Recent advancements in LLMs have accelerated both research and deployment across a wide spectrum of domains. This section outlines prominent emerging trends, alongside unresolved challenges, spanning algorithmic, ethical, and social dimensions (see Figure 6).

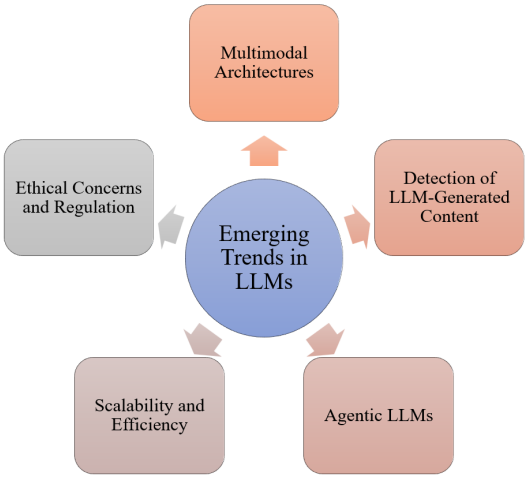

Figure 6. Emerging trends in LLM research.

8.1. Multimodal and Unified Architectures

A dominant research trend is the shift towards unified multimodal architectures designed to natively process and reason across modalities. Pioneering systems like DeepMind's Flamingo [130], and more recent models such as GPT-4o [131,132] and Gemini 1.5 [30], exemplify this trend by integrating visual, textual, and audio inputs within a unified architectural framework. This unified method differs from earlier modular pipelines, which typically relied on separate, specialized models for each modality that were then loosely connected. By leveraging interleaved attention layers and large-scale alignment strategies, modern unified models can perform complex tasks ranging from image captioning to audio-based reasoning. Even with their exciting potential, multimodal LLMs face several key limitations. It's difficult to align different data types within these models properly, and training them demands substantial resources [133]. More research is crucial to see how well these models truly generalize beyond controlled tests and to check how robust they are with different types of inputs.

8.2. Detection of LLM-Generated Content

The widespread availability of content created by LLMs raises urgent concerns about authenticity, plagiarism, and misinformation. Recent studies show that it's becoming increasingly difficult to tell the difference between human-written and LLM-generated text in news articles, scientific papers, and social media posts [134,135]. Because of this, detecting AI-generated content has become a crucial area of research, with methods ranging from watermarking [136] to statistical [137] and neural-based approaches [138], often relying on foundational mathematical methods like the generation of discrete orthogonal matrices [139]. To improve resilience against tampering, researchers are developing advanced methods such as multimodal quantum watermarking for images, which has shown high robustness against noise, geometric, and cropping attacks [140]. However, human performance in identifying LLM-generated content remains close to random chance [134]. Moreover, adversarial attacks and prompt engineering often evade detection, posing a severe risk in educational and scientific contexts [141]. There is an ongoing need for robust detection benchmarks like DetectRL [142], and for integrating these into regulatory and content verification pipelines.

8.3. Agentic LLMs and Tool-Augmented Reasoning

Beyond simple text generation, a significant trend is the development of LLMs that exhibit emergent agentic behaviors, such as sophisticated planning, memory, and tool use. Tools such as Auto-GPT [78] and agents built with LangChain [143] let LLMs do more than just write. They can find information, use other software (APIs), and connect different tasks to complete complex jobs [144]. These agent-like LLMs represent a new way for these models to think and interact with the world. However, there are several significant challenges. These LLM agents can have trouble making stable plans, sometimes invent facts when using tools (tool hallucinations), and forget or corrupt information they're supposed to remember [145]. To make these agents more reliable, researchers are working on ways to "ground" their knowledge in reality, create modular control systems, and clearly define how the agents understand their actions and current state [146].

8.4. Scalability and Efficiency

The growing size and deployment of LLMs raise concerns about compute, energy, and accessibility. Techniques such as parameter-efficient fine-tuning (PEFT) [38] and quantization (e.g., QLoRA) [39] aim to reduce training and inference costs. Nonetheless, efficient models still face trade-offs in robustness, generalization, and downstream performance.

Furthermore, aligning open models without access to large-scale human feedback datasets remains a major bottleneck [147].

8.5. Ethical Concerns, Regulation, and Societal Impact

As LLMs become common in education, journalism, and science, their unsupervised use brings significant societal risks that require careful consideration and robust governance frameworks [148]. Key challenges include:

- Authenticity and Misinformation: It is becoming nearly impossible to tell the difference between human-written and AI-generated text, which threatens academic honesty, public trust in information, and the integrity of the digital ecosystem [149].
- Copyright and Data Provenance: A critical legal and ethical challenge stems from the common practice of training LLMs on vast datasets scraped from the internet, which often include copyrighted materials such as books, articles, and artwork without permission from the creators. This practice has led to numerous high-profile lawsuits from authors, artists, and news organizations who argue that it constitutes mass copyright infringement [150,151]. The central debate revolves around the doctrine of "fair use," with technology companies arguing that training is a transformative use, while rights holders contend it devalues their intellectual property. This highlights the urgent need for transparency about where training data comes from and the development of ethical data-sourcing practices [152,153].
- Cultural and Linguistic Homogenization: The widespread use of LLMs raises concerns about language becoming too uniform, the potential loss of diverse cultural nuances, and a "flattening" of emotional expression in generated content [154,155].
- Regulation and Governance: Although new rules, such as the EU AI Act [156], are beginning to address these concerns, their enforcement and auditability remain limited. This underscores the ongoing need for human oversight in the auditing process and clear, domain-specific guidelines for how these models are used within different communities.

8.6. Open Problems

Several foundational challenges exist across multiple emerging trends in LLM research. Addressing these issues is essential for future progress. Figure 7 illustrates the key open problems outlined below.

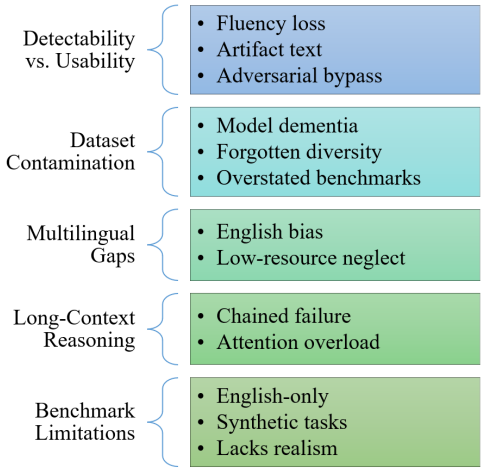

Figure 7. Overview of key open problems in LLM research.

- Detectability vs. Usability Trade-off. The challenge of detectability involves reliably identifying whether a piece of content was generated by an AI model. This

field encompasses several technical strategies, including (a) watermarking, which embeds a secret, statistically detectable signal into the generated output [157,158]; (b) provenance tracking, which involves cryptographic methods to verify the origin of content [159,160]; and (c) model fingerprinting, which identifies the unique stylistic artifacts of a specific model [161,162]. A core open problem is that these detection approaches often degrade output fluency or introduce stylistic characteristics that can hinder creative or assistive writing [163,164]. Furthermore, the robustness of these detectors is often undermined by paraphrasing or adversarial prompt attacks, raising questions about their sustained utility [165,166].

- Dataset Contamination and Model Collapse. Data contamination refers to the unintentional inclusion of test data in a model's training set, leading to inflated performance metrics and an inaccurate assessment of true generalization capabilities [167]. This problem manifests in two key ways: (a) benchmark contamination, where evaluation datasets are inadvertently scraped from the web and included in pre-training corpora, and (b) model collapse, a phenomenon where models trained on the synthetic outputs of previous models suffer a progressive loss of quality as the diversity of human-like language degrades and rare semantic patterns are lost [168]. Furthermore, paraphrased benchmarks can circumvent conventional data decontamination processes, thereby inflating performance estimates [169]. This highlights the critical need for contamination-resilient evaluation and dataset curation methodologies [170].
- Multilingual and Cross-Domain Generalization. Generalization in LLMs refers to the ability to perform effectively on new, unseen data and tasks that differ significantly from the training distribution [171]. A critical open problem is poor generalization in specific contexts, particularly in multilingual and cross-domain settings. Existing benchmarks are overwhelmingly English-centric, leaving low-resource languages and domain-specific tasks underrepresented [172,173]. When applying multilingual LLMs to long non-English contexts, performance can drop dramatically (e.g., from 96% in English to as low as 36% in Somali on multitarget retrieval tasks [172]), highlighting serious equity and inclusivity gaps.
- Long-Context Reasoning and Retrieval. Even models with extremely large context windows struggle with complex multistep reasoning across long texts. Issues like multimatching and logic-based retrieval tasks require chained reasoning and exceed existing attention and chain-of-thought capabilities unless decomposed into numerous steps [174,175]. Furthermore, simply increasing context length often yields diminishing returns or even performance degradation due to "hard negatives" or distracting information [176].
- Benchmark Diversity and Realism. Current benchmarks are often synthetic or English-centered. While the Needle-in-a-Haystack (NIAH) test assesses memory [177], it does not adequately measure deep comprehension or robust reasoning [178,179]. Emerging benchmarks (e.g., RULER [180], PangeaBench [181]) aim to address these gaps but are still limited in scope and cultural reach. A more comprehensive evaluation suite must cover multilingual, multimodal, and real-world reasoning challenges.

In summary, LLMs have achieved remarkable capabilities, but they remain fragile in areas related to authenticity, longevity, inclusivity, and reasoning fidelity. Overcoming these interconnected challenges will require rigorous benchmarking, contamination-aware dataset pipelines, multilingual and multimodal evaluation designs, and more nuanced controller architectures that can robustly manage complexity without sacrificing performance or fairness.

9. Conclusions

This survey has examined core aspects of large language model development, covering adaptation techniques, evaluation metrics, diverse applications, economic dynamics, and emerging research trajectories. We highlighted how parameter-efficient fine-tuning methods offer practical avenues for adapting vast models by updating only a minimal subset of parameters. The discussion also addressed the inherent strengths and limitations of automated evaluation metrics, emphasizing the persistent need for human-in-the-loop evaluation to capture complexities such as coherence, factual accuracy, and linguistic fluency. We also showcased the transformative potential of LLMs through their various cross-domain applications. Economically, our analysis revealed that significant upfront training costs, ongoing inference expenses, and substantial human labor profoundly shape market structures and raise concerns regarding inequality and labor disruption. Our survey concluded by exploring current trends, including multimodal LLMs and tool-augmented agents, while identifying persistent open challenges such as detectability, data contamination, and generalization. Addressing these complex issues requires collaborative progress across multiple fronts. Future work must focus not only on advancing adaptation and evaluation techniques but also on developing novel infrastructure, such as frameworks for decentralized training and federated evaluation, to mitigate the centralizing pressures of high computational costs. Concurrently, robust governance mechanisms are essential, including standards for data transparency, independent model audits, and clear licensing regimes to ensure responsible and equitable LLM deployment.

Author Contributions: Conceptualization, S.M.S.M., B.C.K., C.E., C.U., M.S.T. and O.K.; methodology, S.M.S.M., B.C.K. and C.E.; writing—original draft preparation, S.M.S.M., B.C.K., C.E. and O.K.; writing—review and editing, S.M.S.M., B.C.K., C.E. and O.K.; visualization, S.M.S.M., B.C.K. and C.E.; supervision, C.E., M.S.T. and O.K. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: No new data were created in this study. Data sharing is not applicable to this article.

Conflicts of Interest: The authors declare that they have no conflicts of interest. This research has not received any specific grant from public funding agencies or commercial or not-for-profit sectors.

Abbreviations

The following abbreviations are used in this manuscript:

AI	Artificial Intelligence
AWS	Amazon Web Services
BERT	Bidirectional Encoder Representations from Transformers
BIG-bench	Beyond the Imitation Game benchmark
BLEU	Bilingual Evaluation Understudy
CLM	Causal Language Modeling
COPA	Choice of Plausible Alternatives
CRF	Conditional Random Fields
EDA	Electronic Design Automation
EHR	Electronic Health Record
FLOPs	Floating Point Operations per Second
GCP	Google Cloud Platform

GPU	Graphics Processing Unit
GRU	Gated Recurrent Unit
HDL	Hardware Description Language
HELM	Holistic Evaluation of Language Models
HMM	Hidden Markov Model
KV	Key-Value
LCS	Longest Common Subsequence
LLM	Large Language Model
LoRA	Low-Rank Adaptation
LSTM	Long Short-Term Memory
MFU	Model FLOPS Utilization
MLM	Masked Language Modeling
MMLU	Massive Multitask Language Understanding
NER	Named Entity Recognition
NIAH	Needle-in-a-Haystack
NLG	Natural Language Generation
NLP	Natural Language Processing
NLU	Natural Language Understanding
NTP	Next Token Prediction
PEFT	Parameter-Efficient Fine-Tuning
QA	Question Answering
RLHF	Reinforcement Learning from Human Feedback
RNN	Recurrent Neural Network
ROUGE	Recall-Oriented Understudy for Gisting Evaluation
RTE	Recognizing Textual Entailment
Seq2Seq	Sequence-to-Sequence
SME	Small and Medium-sized Enterprises
SSL	Self-Supervised Learning
STEM	Science, Technology, Engineering, and Mathematics
TPU	Tensor Processing Unit
VL	Vision-Language
WSC	Winograd Schema Challenge

References

1. Radford, A.; Narasimhan, K.; Salimans, T.; Sutskever, I. Improving Language Understanding by Generative Pre-Training. OpenAI Technical Report 2018. Available online: https://cdn.openai.com/research-covers/language-unsupervised/language_understanding_paper.pdf (accessed on 20 June 2025).
2. Radford, A.; Wu, J.; Child, R.; Luan, D.; Amodei, D.; Sutskever, I. Language models are unsupervised multitask learners. *OpenAI Blog* **2019**, *1*, 9.
3. Brown, T.; Mann, B.; Ryder, N.; Subbiah, M.; Kaplan, J.D.; Dhariwal, P.; Neelakantan, A.; Shyam, P.; Sastry, G.; Askell, A.; et al. Language models are few-shot learners. *Adv. Neural Inf. Process. Syst.* **2020**, *33*, 1877–1901.
4. Chowdhery, A.; Narang, S.; Devlin, J.; Bosma, M.; Mishra, G.; Roberts, A.; Barham, P.; Chung, H.W.; Sutton, C.; Gehrmann, S.; et al. Palm: Scaling language modeling with pathways. *J. Mach. Learn. Res.* **2023**, *24*, 1–113.
5. Touvron, H.; Lavril, T.; Izacard, G.; Martinet, X.; Lachaux, M.A.; Lacroix, T.; Rozière, B.; Goyal, N.; Hambro, E.; Azhar, F.; et al. Llama: Open and efficient foundation language models. *arXiv* **2023**, arXiv:2302.13971. [CrossRef]
6. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, Ł.; Polosukhin, I. Attention is all you need. In Proceedings of the 31st Conference on Neural Information Processing Systems 2017, Long Beach, CA, USA, 4–9 December 2017; Advances in Neural Information Processing Systems; Volume 30.
7. Weizenbaum, J. ELIZA—A computer program for the study of natural language communication between man and machine. *Commun. ACM* **1966**, *9*, 36–45. [CrossRef]
8. Colby, K.M.; Weber, S.; Hilf, F.D. Artificial paranoia. *Artif. Intell.* **1971**, *2*, 1–25. [CrossRef]
9. Wallace, R.S. *The Anatomy of ALICE*; Springer: Berlin/Heidelberg, Germany, 2009.
10. Winograd, T. *Procedures as a Representation for Data in a Computer Program for Understanding natural Language*; MIT Press: Cambridge, MA, USA, 1971.

11. Jelinek, F. *Statistical Methods for Speech Recognition*; MIT Press: Cambridge, MA, USA, 1998.
12. Rabiner, L.R. A tutorial on hidden Markov models and selected applications in speech recognition. *Proc. IEEE* **2002**, *77*, 257–286. [CrossRef]
13. Lafferty, J.; McCallum, A.; Pereira, F. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In Proceedings of the ICML, Williamstown, MA, USA, 28 June–1 July 2001; Volume 1, p. 3.
14. Elman, J.L. Finding structure in time. *Cogn. Sci.* **1990**, *14*, 179–211. [CrossRef]
15. Hochreiter, S.; Schmidhuber, J. Long short-term memory. *Neural Comput.* **1997**, *9*, 1735–1780. [CrossRef] [PubMed]
16. Cho, K.; Van Merriënboer, B.; Gulcehre, C.; Bahdanau, D.; Bougares, F.; Schwenk, H.; Bengio, Y. Learning phrase representations using RNN encoder-decoder for statistical machine translation. *arXiv* **2014**, arXiv:1406.1078. [CrossRef]
17. Mikolov, T.; Chen, K.; Corrado, G.; Dean, J. Efficient estimation of word representations in vector space. *arXiv* **2013**, arXiv:1301.3781. [CrossRef]
18. Pennington, J.; Socher, R.; Manning, C.D. Glove: Global vectors for word representation. In Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), Doha, Qatar, 25–29 October 2014; pp. 1532–1543.
19. Devlin, J.; Chang, M.W.; Lee, K.; Toutanova, K. Bert: Pre-training of deep bidirectional transformers for language understanding. In Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Minneapolis, MN, USA, 2–7 June 2019; Volume 1, pp. 4171–4186.
20. Liu, A.; Feng, B.; Xue, B.; Wang, B.; Wu, B.; Lu, C.; Zhao, C.; Deng, C.; Zhang, C.; Ruan, C.; et al. Deepseek-v3 technical report. *arXiv* **2024**, arXiv:2412.19437.
21. Raffel, C.; Shazeer, N.; Roberts, A.; Lee, K.; Narang, S.; Matena, M.; Zhou, Y.; Li, W.; Liu, P.J. Exploring the limits of transfer learning with a unified text-to-text transformer. *J. Mach. Learn. Res.* **2020**, *21*, 1–67.
22. Liu, Y.; Ott, M.; Goyal, N.; Du, J.; Joshi, M.; Chen, D.; Levy, O.; Lewis, M.; Zettlemoyer, L.; Stoyanov, V. Roberta: A robustly optimized bert pretraining approach. *arXiv* **2019**, arXiv:1907.11692.
23. Lan, Z.; Chen, M.; Goodman, S.; Gimpel, K.; Sharma, P.; Soricut, R. Albert: A lite bert for self-supervised learning of language representations. *arXiv* **2019**, arXiv:1909.11942.
24. Anil, R.; Dai, A.M.; Firat, O.; Johnson, M.; Lepikhin, D.; Passos, A.; Shakeri, S.; Taropa, E.; Bailey, P.; Chen, Z.; et al. Palm 2 technical report. *arXiv* **2023**, arXiv:2305.10403. [CrossRef]
25. Lewis, M.; Liu, Y.; Goyal, N.; Ghazvininejad, M.; Mohamed, A.; Levy, O.; Stoyanov, V.; Zettlemoyer, L. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. *arXiv* **2019**, arXiv:1910.13461.
26. Lu, H.; Liu, W.; Zhang, B.; Wang, B.; Dong, K.; Liu, B.; Sun, J.; Ren, T.; Li, Z.; Yang, H.; et al. Deepseek-vl: Towards real-world vision-language understanding. *arXiv* **2024**, arXiv:2403.05525.
27. Hurst, A.; Lerer, A.; Goucher, A.P.; Perelman, A.; Ramesh, A.; Clark, A.; Ostrow, A.; Welihinda, A.; Hayes, A.; Radford, A.; et al. Gpt-4o system card. *arXiv* **2024**, arXiv:2410.21276. [CrossRef]
28. Shazeer, N.; Mirhoseini, A.; Maziarz, K.; Davis, A.; Le, Q.; Hinton, G.; Dean, J. Outrageously large neural networks: The sparsely-gated mixture-of-experts layer. *arXiv* **2017**, arXiv:1701.06538.
29. Jiang, A.Q.; Sablayrolles, A.; Roux, A.; Mensch, A.; Savary, B.; Bamford, C.; Chaplot, D.S.; Casas, D.d.l.; Hanna, E.B.; Bressand, F.; et al. Mixtral of experts. *arXiv* **2024**, arXiv:2401.04088. [CrossRef]
30. Team, G.; Georgiev, P.; Lei, V.I.; Burnell, R.; Bai, L.; Gulati, A.; Tanzer, G.; Vincent, D.; Pan, Z.; Wang, S.; et al. Gemini 1.5: Unlocking multimodal understanding across millions of tokens of context. *arXiv* **2024**, arXiv:2403.05530. [CrossRef]
31. Fedus, W.; Zoph, B.; Shazeer, N. Switch transformers: Scaling to trillion parameter models with simple and efficient sparsity. *J. Mach. Learn. Res.* **2022**, *23*, 1–39.
32. Wei, J.; Bosma, M.; Zhao, V.Y.; Guu, K.; Yu, A.W.; Lester, B.; Du, N.; Dai, A.M.; Le, Q.V. Finetuned language models are zero-shot learners. *arXiv* **2021**, arXiv:2109.01652.
33. Sanh, V.; Webson, A.; Raffel, C.; Bach, S.H.; Sutawika, L.; Alyafeai, Z.; Chaffin, A.; Stiegler, A.; Scao, T.L.; Raja, A.; et al. Multitask prompted training enables zero-shot task generalization. *arXiv* **2021**, arXiv:2110.08207.
34. Ouyang, L.; Wu, J.; Jiang, X.; Almeida, D.; Wainwright, C.; Mishkin, P.; Zhang, C.; Agarwal, S.; Slama, K.; Ray, A.; et al. Training language models to follow instructions with human feedback. *Adv. Neural Inf. Process. Syst.* **2022**, *35*, 27730–27744.
35. Houlsby, N.; Giurgiu, A.; Jastrzebski, S.; Morrone, B.; De Laroussilhe, Q.; Gesmundo, A.; Attariyan, M.; Gelly, S. Parameter-efficient transfer learning for NLP. In Proceedings of the International Conference on Machine Learning, PMLR, Long Beach, CA, USA, 9–15 June 2019; pp. 2790–2799.
36. Lester, B.; Al-Rfou, R.; Constant, N. The power of scale for parameter-efficient prompt tuning. *arXiv* **2021**, arXiv:2104.08691. [CrossRef]
37. Li, X.L.; Liang, P. Prefix-tuning: Optimizing continuous prompts for generation. *arXiv* **2021**, arXiv:2101.00190. [CrossRef]
38. Hu, E.J.; Shen, Y.; Wallis, P.; Allen-Zhu, Z.; Li, Y.; Wang, S.; Wang, L.; Chen, W. Lora: Low-rank adaptation of large language models. *ICLR* **2022**, *1*, 3.

39. Dettmers, T.; Pagnoni, A.; Holtzman, A.; Zettlemoyer, L. Qlora: Efficient finetuning of quantized llms. *Adv. Neural Inf. Process. Syst.* **2023**, *36*, 10088–10115.
40. Hendrycks, D.; Burns, C.; Basart, S.; Zou, A.; Mazeika, M.; Song, D.; Steinhardt, J. Measuring massive multitask language understanding. *arXiv* **2020**, arXiv:2009.03300.
41. Srivastava, A.; Rastogi, A.; Rao, A.; Shoeb, A.A.M.; Abid, A.; Fisch, A.; Brown, A.R.; Santoro, A.; Gupta, A.; Garriga-Alonso, A.; et al. Beyond the imitation game: Quantifying and extrapolating the capabilities of language models. *arXiv* **2022**, arXiv:2206.04615. [CrossRef]
42. Wang, A.; Pruksachatkun, Y.; Nangia, N.; Singh, A.; Michael, J.; Hill, F.; Levy, O.; Bowman, S. Superglue: A stickier benchmark for general-purpose language understanding systems. *Adv. Neural Inf. Process. Syst.* **2019**, *32*. [CrossRef]
43. Liang, P.; Bommasani, R.; Lee, T.; Tsipras, D.; Soylu, D.; Yasunaga, M.; Zhang, Y.; Narayanan, D.; Wu, Y.; Kumar, A.; et al. Holistic evaluation of language models. *arXiv* **2022**, arXiv:2211.09110. [CrossRef]
44. Lin, C.Y. Rouge: A package for automatic evaluation of summaries. In Proceedings of the Text Summarization Branches Out, Barcelona, Spain, 25–26 July 2004; pp. 74–81.
45. Papineni, K.; Roukos, S.; Ward, T.; Zhu, W.J. Bleu: A method for automatic evaluation of machine translation. In Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics, Philadelphia, PA, USA, 7–12 July 2002; pp. 311–318.
46. Eger, S.; Cao, Y.; D'Souza, J.; Geiger, A.; Greisinger, C.; Gross, S.; Hou, Y.; Krenn, B.; Lauscher, A.; Li, Y.; et al. Transforming Science with Large Language Models: A Survey on AI-assisted Scientific Discovery, Experimentation, Content Generation, and Evaluation. *arXiv* **2025**, arXiv:2502.05151.
47. Whitfield, S.; Hofmann, M.A. Elicit: AI literature review research assistant. *Public Serv. Q.* **2023**, *19*, 201–207. [CrossRef]
48. Ramos, M.C.; Collison, C.J.; White, A.D. A review of large language models and autonomous agents in chemistry. *Chem. Sci.* **2025**, *16*, 2514–2572. [CrossRef]
49. Frincu, I. In Search of the Perfect Prompt. Master's Thesis, Aalto University, Espoo, Finland, 2023.
50. Borhani, M.; Soofiani, N.M.; Ebrahimi, E.; Asadollah, S. First Report on Growth and Reproduction of Turcinoemacheilus bahaii (Esmaeili, Sayyadzadeh, Özulug, Geiger and Freyhof, 2014), in Zayandeh Roud River, Iran. *Austin Environ. Sci.* **2017**, *2*, 1014. [CrossRef]
51. Nazi, Z.A.; Peng, W. Large language models in healthcare and medical domain: A review. *Informatics* **2024**, *11*, 57. [CrossRef]
52. Mohammadabadi, S.M.S.; Peikani, M.B. Identification and classification of rheumatoid arthritis using artificial intelligence and machine learning. In *Diagnosing Musculoskeletal Conditions Using Artifical Intelligence and Machine Learning to Aid Interpretation of Clinical Imaging*; Elsevier: Amsterdam, The Netherlands, 2025; pp. 123–145.
53. Zheng, Y.; Koh, H.Y.; Yang, M.; Li, L.; May, L.T.; Webb, G.I.; Pan, S.; Church, G. Large language models in drug discovery and development: From disease mechanisms to clinical trials. *arXiv* **2024**, arXiv:2409.04481. [CrossRef]
54. Mohammadabadi, S.M.S.; Seyedkhamoushi, F.; Mostafavi, M.; Peikani, M.B. Examination of AI's role in Diagnosis, Treatment, and Patient care. In *Transforming Gender-Based Healthcare with AI and Machine Learning*; CRC Press: Boca Raton, FL, USA, 2024; pp. 221–238.
55. Huynh, N.; Lin, B. Large Language Models for Code Generation: A Comprehensive Survey of Challenges, Techniques, Evaluation, and Applications. *arXiv* **2025**, arXiv:2503.01245.
56. Wong, M.F.; Guo, S.; Hang, C.N.; Ho, S.W.; Tan, C.W. Natural language generation and understanding of big code for AI-assisted programming: A review. *Entropy* **2023**, *25*, 888. [CrossRef]
57. Liu, B.; Jiang, Y.; Zhang, Y.; Niu, N.; Li, G.; Liu, H. An Empirical Study on the Potential of LLMs in Automated Software Refactoring. *arXiv* **2024**, arXiv:2411.04444. [CrossRef]
58. Zhong, R.; Du, X.; Kai, S.; Tang, Z.; Xu, S.; Zhen, H.L.; Hao, J.; Xu, Q.; Yuan, M.; Yan, J. Llm4eda: Emerging progress in large language models for electronic design automation. *arXiv* **2023**, arXiv:2401.12224. [CrossRef]
59. Mohammadabadi, S.M.S.; Entezami, M.; Moghaddam, A.K.; Orangian, M.; Nejadshamsi, S. Generative artificial intelligence for distributed learning to enhance smart grid communication. *Int. J. Intell. Netw.* **2024**, *5*, 267–274.
60. Mohammadabadi, S.M.S.; Liu, Y.; Canafe, A.; Yang, L. Towards distributed learning of pmu data: A federated learning based event classification approach. In Proceedings of the 2023 IEEE Power & Energy Society General Meeting (PESGM), Orlando, FL, USA, 16–20 July 2023; IEEE: New York, NY, USA, 2023; pp. 1–5.
61. Li, Y.; Wang, S.; Ding, H.; Chen, H. Large language models in finance: A survey. In Proceedings of the Fourth ACM International Conference on AI in Finance, Brooklyn, NY, USA, 27–29 November 2023; pp. 374–382.
62. Peddinti, S.R.; Katragadda, S.R.; Pandey, B.K.; Tanikonda, A. Utilizing large language models for advanced service management: Potential applications and operational challenges. *J. Sci. Technol.* **2023**, *4*. [CrossRef]
63. Lopez-Lira, A.; Kwon, J.; Yoon, S.; Sohn, J.y.; Choi, C. Bridging language models and financial analysis. *arXiv* **2025**, arXiv:2503.22693.
64. Sriram, A. Comparative Forecasting in Retail Supply Chains Using Machine Learning and Large Language Models. Master's Thesis, State University of New York at Binghamton, Binghamton, NY, USA, 2025.

65. Brundage, M.P.; Sharp, M.; Pavel, R. Qualifying evaluations from human operators: Integrating sensor data with natural language logs. *PHME Soc. Eur. Conf.* **2021**, *6*, 9. [CrossRef]
66. Bakas, N.P.; Papadaki, M.; Vagianou, E.; Christou, I.; Chatzichristofis, S.A. Integrating llms in higher education, through interactive problem solving and tutoring: Algorithmic approach and use cases. In Proceedings of the European, Mediterranean, and Middle Eastern Conference on Information Systems, Dubai, United Arab Emirates, 11–12 December 2023; Springer: Cham, Switzerland, 2023; pp. 291–307.
67. McHugh, B.; Myers, D.; Patel, A. AI Co-Counsel: An Attorney's Guide to Using Artificial Intelligence in the Practice of Law Symposium. *Akron Law Rev.* **2024**, *57*, 3.
68. Khikmatillaeva, M. Beyond Chatbots: How specialized AI tools are reducing legal workloads. *FARS Int. J. Educ. Soc. Sci. Humanit.* **2025**, *13*, 133–154.
69. Cheng, S. When Journalism meets AI: Risk or opportunity? *Digit. Gov. Res. Pract.* **2025**, *6*, 1–12. [CrossRef]
70. Dhariwal, P.; Jun, H.; Payne, C.; Kim, J.W.; Radford, A.; Sutskever, I. Jukebox: A generative model for music. *arXiv* **2020**, arXiv:2005.00341. [CrossRef]
71. Topirceanu, A.; Barina, G.; Udrescu, M. Musenet: Collaboration in the music artists industry. In Proceedings of the 2014 European Network Intelligence Conference, Wroclaw, Poland, 29–30 September 2014; IEEE: New York, NY, USA, 2014; pp. 89–94.
72. Marcus, G.; Davis, E.; Aaronson, S. A very preliminary analysis of DALL-E 2. *arXiv* **2022**, arXiv:2204.13807. [CrossRef]
73. Lu, Y. Artificial Intelligence Applied On Today's Urban and Architectural Conceptual Design-A Competition Case Study. Ph.D. Thesis, Politecnico di Torino, Torino, Italy, 2025.
74. Shetye, S. An evaluation of khanmigo, a generative ai tool, as a computer-assisted language learning app. *Stud. Appl. Linguist. TESOL* **2024**, *24*. [CrossRef]
75. Vega, J.; Rodriguez, M.; Check, E.; Moran, H.; Loo, L. Duolingo evolution: From automation to artificial intelligence. In Proceedings of the IEEE Colombian Conference on Applications of Computational Intelligence, Pamplona, Colombia, 17–19 July 2024; Springer: Berlin/Heidelberg, Germany, 2025; pp. 54–71.
76. Calamas, D. *Student and Instructor Feedback on an AI-Assisted Grading Tool*; American Society for Engineering Education: Washington, DC, USA, 2024.
77. Hidalgo-Reyes, J.; Alvarez, J.; Guevara-Chavez, L.; Cruz-Netro, Z.G. Gradescope as a Tool to Improve Assessment and Feedback in Engineering. In Proceedings of the 2025 Institute for the Future of Education Conference (IFE), Monterrey, Mexico, 28–30 January 2025; IEEE: New York, NY, USA, 2025; pp. 1–7.
78. Yang, H.; Yue, S.; He, Y. Auto-gpt for online decision making: Benchmarks and additional opinions. *arXiv* **2023**, arXiv:2306.02224. [CrossRef]
79. O'BRIEN, P.D.; Wiegand, M.E. Agent based process management: Applying intelligent agents to workflow. *Knowl. Eng. Rev.* **1998**, *13*, 161–174. [CrossRef]
80. Van Veen, D.; Van Uden, C.; Blankemeier, L.; Delbrouck, J.-B.; Aali, A.; Bluethgen, C.; Pareek, A.; Polacin, M.; Reis, E.P.; Seehofnerová, A.; et al. Clinical Text Summarization: Adapting Large Language Models Can Outperform Human Experts. *Res. Sq.* **2023**. [CrossRef]
81. Rao, A.; Kim, J.; Kamineni, M.; Pang, M.; Lie, W.; Dreyer, K.J.; Succi, M.D. Evaluating GPT as an adjunct for radiologic decision making: GPT-4 versus GPT-3.5 in a breast imaging pilot. *J. Am. Coll. Radiol.* **2023**, *20*, 990–997. [CrossRef]
82. Singhal, K.; Azizi, S.; Tu, T.; Mahdavi, S.S.; Wei, J.; Chung, H.W.; Scales, N.; Tanwani, A.; Cole-Lewis, H.; Pfohl, S.; et al. Large language models encode clinical knowledge. *Nature* **2023**, *620*, 172–180. [CrossRef]
83. Benary, M.; Wang, X.D.; Schmidt, M.; Soll, D.; Hilfenhaus, G.; Nassir, M.; Sigler, C.; Knödler, M.; Keller, U.; Beule, D.; et al. Leveraging large language models for decision support in personalized oncology. *JAMA Netw. Open* **2023**, *6*, e2343689. [CrossRef] [PubMed]
84. Clusmann, J.; Kolbinger, F.R.; Muti, H.S.; Carrero, Z.I.; Eckardt, J.N.; Laleh, N.G.; Löffler, C.M.L.; Schwarzkopf, S.C.; Unger, M.; Veldhuizen, G.P.; et al. The future landscape of large language models in medicine. *Commun. Med.* **2023**, *3*, 141. [CrossRef]
85. Wu, S.; Irsoy, O.; Lu, S.; Dabravolski, V.; Dredze, M.; Gehrmann, S.; Kambadur, P.; Rosenberg, D.; Mann, G. Bloomberggpt: A large language model for finance. *arXiv* **2023**, arXiv:2303.17564. [CrossRef]
86. Völker, T.; Pfister, J.; Koopmann, T.; Hotho, A. From Chat to Publication Management: Organizing your related work using BibSonomy & LLMs. In Proceedings of the 2024 Conference on Human Information Interaction and Retrieval, Sheffield, UK, 10–14 March 2024; pp. 386–390.
87. Rane, N.; Choudhary, S.; Rane, J. Gemini versus ChatGPT: Applications, performance, architecture, capabilities, and implementation. *J. Appl. Artif. Intell.* **2024**, *5*, 69–93. [CrossRef]
88. Wang, W.; Gu, L.; Zhang, L.; Luo, Y.; Dai, Y.; Shen, C.; Xie, L.; Lin, B.; He, X.; Ye, J. SciPIP: An LLM-based Scientific Paper Idea Proposer. *arXiv* **2024**, arXiv:2410.23166.

89. Sırt, M.; Eyüpoğlu, C. A user-friendly and explainable framework for redesigning AutoML processes with large language models. In Proceedings of the 33rd Signal Processing and Communications Applications Conference (SIU), Sile, Istanbul, Turkiye, 25–28 June 2025; pp. 1–4. [CrossRef]
90. Xia, H.; Yang, Z.; Zhao, Y.; Wang, Y.; Li, J.; Tracy, R.; Zhu, Z.; Wang, Y.F.; Chen, H.; Shen, W. Language and multimodal models in sports: A survey of datasets and applications. *arXiv* **2024**, arXiv:2406.12252. [CrossRef]
91. Rahimian, P.; Flisar, J.; Sumpter, D. Automated explanation of machine learning models of footballing actions in words. *J. Sport. Anal.* **2025**, *11*, 22150218251353089. [CrossRef]
92. Cottier, B.; Rahman, R.; Fattorini, L.; Maslej, N.; Besiroglu, T.; Owen, D. The rising costs of training frontier AI models. *arXiv* **2024**, arXiv:2405.21015. [CrossRef]
93. Liu, Y.; He, H.; Han, T.; Zhang, X.; Liu, M.; Tian, J.; Zhang, Y.; Wang, J.; Gao, X.; Zhong, T.; et al. Understanding llms: A comprehensive overview from training to inference. *Neurocomputing* **2025**, *620*, 129190. [CrossRef]
94. Gale, T.; Elsen, E.; Hooker, S. Do Neural Networks Really Need to Be So Big? 2020. MIT-IBM Watson AI Lab Blog. Available online: https://mitibmwatsonailab.mit.edu/research/blog/do-neural-networks-really-need-to-be-so-big/ (accessed on 1 August 2025).
95. Buchholz, K. The Extreme Cost of Training AI Models. 2024. Available online: https://www.forbes.com/sites/katharinabuchholz/2024/08/23/the-extreme-cost-of-training-ai-models/ (accessed on 20 June 2025).
96. for Human-Centered Artificial Intelligence (HAI), S.I. AI Index Report. 2024. Available online: https://hai.stanford.edu/research/ai-index-2024 (accessed on 20 June 2025).
97. OpenAI. API Pricing. 2025. Available online: https://openai.com/api/pricing/ (accessed on 20 June 2025).
98. Maslej, N.; Fattorini, L.; Brynjolfsson, E.; Etchemendy, J.; Ligett, K.; Lyons, T.; Manyika, J.; Ngo, H.; Niebles, J.C.; Parli, V.; et al. Artificial intelligence index report 2023. *arXiv* **2023**, arXiv:2310.03715. [CrossRef]
99. Pope, R.; Douglas, S.; Chowdhery, A.; Devlin, J.; Bradbury, J.; Heek, J.; Xiao, K.; Agrawal, S.; Dean, J. Efficiently scaling transformer inference. *Proc. Mach. Learn. Syst.* **2023**, *5*, 606–624.
100. IoT Analytics. The Leading Generative AI Companies. 2025. Available online: https://iot-analytics.com/leading-generative-ai-companies/ (accessed on 19 June 2025).
101. Zhang, M.; Yuan, B.; Li, H.; Xu, K. LLM-Cloud Complete: Leveraging cloud computing for efficient large language model-based code completion. *J. Artif. Intell. Gen. Sci. (JAIGS)* **2024**, *5*, 295–326. [CrossRef]
102. Vipra, J.; Korinek, A. Market concentration implications of foundation models. *arXiv* **2023**, arXiv:2311.01550. [CrossRef]
103. Bommasani, R.; Klyman, K.; Longpre, S.; Kapoor, S.; Maslej, N.; Xiong, B.; Zhang, D.; Liang, P. The 2023 Foundation Model Transparency Index. *Trans. Mach. Learn. Res.* **2025.** [CrossRef]
104. Ludwig, J.; Mullainathan, S.; Rambachan, A. *Large Language Models: An Applied Econometric Framework*; Technical Report; National Bureau of Economic Research: Cambridge, MA, USA, 2025.
105. Castro, D. AI Can Improve U.S. Small Business Productivity. Information Technology & Innovation Foundation. 2025. Available online: https://itif.org/publications/2025/04/08/ai-can-improve-us-small-business-productivity/ (accessed on 1 August 2025).
106. Newstardom Insights. The AI Accessibility Gap: Can Small Businesses Keep Up? 2024. Available online: https://newstardom.com/insights/the-ai-accessibility-gap-can-small-businesses-keep-up (accessed on 19 June 2025).
107. Jain, A.; Kakade, K.S.; Vispute, S.A. The Role of Artificial Intelligence (AI) in the Transformation of Small-and Medium-Sized Businesses: Challenges and Opportunities. In *Artificial Intelligence-Enabled Businesses: How to Develop Strategies for Innovation*; John Wiley & Sons: Hoboken, NJ, USA, 2025; pp. 209–226.
108. Korinek, A.; Vipra, J. Concentrating intelligence: Scaling and market structure in artificial intelligence. *Econ. Policy* **2025**, *40*, 225–256. [CrossRef]
109. Xie, Y.; Avila, S. The social impact of generative LLM-based AI. *Chin. J. Sociol.* **2025**, *11*, 31–57. [CrossRef]
110. Acemoglu, D.; Restrepo, P. Tasks, automation, and the rise in US wage inequality. *Econometrica* **2022**, *90*, 1973–2016. [CrossRef]
111. Touvron, H.; Martin, L.; Stone, K.; Albert, P.; Almahairi, A.; Babaei, Y.; Bashlykov, N.; Batra, S.; Bhargava, P.; Bhosale, S.; et al. Llama 2: Open foundation and fine-tuned chat models. *arXiv* **2023**, arXiv:2307.09288. [CrossRef]
112. Grattafiori, A.; Dubey, A.; Jauhri, A.; Pandey, A.; Kadian, A.; Al-Dahle, A.; Letman, A.; Mathur, A.; Schelten, A.; Vaughan, A.; et al. The llama 3 herd of models. *arXiv* **2024**, arXiv:2407.21783. [CrossRef]
113. Phang, J.; Bradley, H.; Gao, L.; Castricato, L.; Biderman, S. EleutherAI: Going Beyond "Open Science" to "Science in the Open". *arXiv* **2022**, arXiv:2210.06413. [CrossRef]
114. Sinha, S.; Lee, Y.M. Challenges with developing and deploying AI models and applications in industrial systems. *Discov. Artif. Intell.* **2024**, *4*, 55. [CrossRef]
115. Fossen, F.; Sorgner, A. Mapping the future of occupations: Transformative and destructive effects of new digital technologies on jobs. *Foresight* **2019**, *13*, 10–18. [CrossRef]
116. Wang, Y. The large language model (llm) paradox: Job creation and loss in the age of advanced ai. *Authorea Prepr.* **2023.** [CrossRef]
117. Durach, C.F.; Gutierrez, L. "Hello, this is your AI co-pilot"–operational implications of artificial intelligence chatbots. *Int. J. Phys. Distrib. Logist. Manag.* **2024**, *54*, 229–246. [CrossRef]

118. Dillon, E.W.; Jaffe, S.; Immorlica, N.; Stanton, C.T. *Shifting Work Patterns with Generative AI*; Technical report; National Bureau of Economic Research: Cambridge, MA, USA, 2025.
119. Wang, J.Y.; Sukiennik, N.; Li, T.; Su, W.; Hao, Q.; Xu, J.; Huang, Z.; Xu, F.; Li, Y. A Survey on Human-Centric LLMs. *arXiv* **2024**, arXiv:2411.14491. [CrossRef]
120. Niu, Q.; Liu, J.; Bi, Z.; Feng, P.; Peng, B.; Chen, K.; Li, M.; Yan, L.K.; Zhang, Y.; Yin, C.H.; et al. Large language models and cognitive science: A comprehensive review of similarities, differences, and challenges. *arXiv* **2024**, arXiv:2409.02387. [CrossRef]
121. Johnson, S.; Acemoglu, D. *Power and Progress: Our Thousand-Year Struggle Over Technology and Prosperity | Winners of the 2024 Nobel Prize for Economics*; Hachette: London, UK, 2023.
122. Wilmers, N. Generative AI and the Future of Inequality. *MIT Exploration of Generative AI*, 27 March 2024. [CrossRef]
123. Rockall, E.; Mendes Tavares, M.; Pizzinelli, C. *AI Adoption and Inequality*; International Monetary Fund: Washington, DC, USA, 2025.
124. Tosun, M.S. Ageing Robots to the Rescue, Technical Report, Oxford Institute of Population Ageing. 2023. Oxford Institute of Population Ageing Blog. Available online: https://www.ageing.ox.ac.uk/blog/ageing-robots-to-the-rescue (accessed on 20 June 2025).
125. Tosun, M.S. Endogenous fiscal policy and capital market transmissions in the presence of demographic shocks. *J. Econ. Dyn. Control* **2008**, *32*, 2031–2060. [CrossRef]
126. Tosun, M.S. *Global Aging and Fiscal Policy with International Labor Mobility: A Political Economy Perspective*; Technical Report, IZA Discussion Papers; International Monetary Fund: Washington, DC, USA, 2009.
127. Weeks, W.B.; Spelhaug, J.; Weinstein, J.N.; Ferres, J.M.L. Bridging the rural-urban divide: An implementation plan for leveraging technology and artificial intelligence to improve health and economic outcomes in rural America. *J. Rural. Health* **2024**, *40*. [CrossRef]
128. Woods, D.; Podhorzer, M. *AI's Impact on Income Inequality in the U.S.*; Technical Report; Brookings Institution: Washington, DC, USA, 2023.
129. Acemoglu, D.; Restrepo, P. Secular stagnation? The effect of aging on economic growth in the age of automation. *Am. Econ. Rev.* **2017**, *107*, 174–179. [CrossRef]
130. Alayrac, J.B.; Donahue, J.; Luc, P.; Miech, A.; Barr, I.; Hasson, Y.; Lenc, K.; Mensch, A.; Millican, K.; Reynolds, M.; et al. Flamingo: A visual language model for few-shot learning. *Adv. Neural Inf. Process. Syst.* **2022**, *35*, 23716–23736.
131. Shahriar, S.; Lund, B.D.; Mannuru, N.R.; Arshad, M.A.; Hayawi, K.; Bevara, R.V.K.; Mannuru, A.; Batool, L. Putting gpt-4o to the sword: A comprehensive evaluation of language, vision, speech, and multimodal proficiency. *Appl. Sci.* **2024**, *14*, 7782. [CrossRef]
132. Islam, R.; Moushi, O.M. Gpt-4o: The cutting-edge advancement in multimodal llm. *Authorea Prepr.* **2024**. [CrossRef]
133. Song, S.; Li, X.; Li, S.; Zhao, S.; Yu, J.; Ma, J.; Mao, X.; Zhang, W.; Wang, M. How to Bridge the Gap between Modalities: Survey on Multimodal Large Language Model. *IEEE Trans. Knowl. Data Eng.* **2025**, *7*, 5311–5329. [CrossRef]
134. Wu, J.; Yang, S.; Zhan, R.; Yuan, Y.; Chao, L.S.; Wong, D.F. A survey on LLM-generated text detection: Necessity, methods, and future directions. *Comput. Linguist.* **2025**, *51*, 275–338. [CrossRef]
135. Muñoz-Ortiz, A.; Gómez-Rodríguez, C.; Vilares, D. Contrasting linguistic patterns in human and llm-generated news text. *Artif. Intell. Rev.* **2024**, *57*, 265. [CrossRef]
136. Dathathri, S.; See, A.; Ghaisas, S.; Huang, P.S.; McAdam, R.; Welbl, J.; Bachani, V.; Kaskasoli, A.; Stanforth, R.; Matejovicova, T.; et al. Scalable watermarking for identifying large language model outputs. *Nature* **2024**, *634*, 818–823. [CrossRef] [PubMed]
137. Sun, Y.; He, J.; Cui, L.; Lei, S.; Lu, C.T. Exploring the deceptive power of llm-generated fake news: A study of real-world detection challenges. *arXiv* **2024**, arXiv:2403.18249.
138. Goswami, A.; Kaur, G.; Tayal, S.; Verma, A.; Verma, M. Analyzing the efficacy of Deep Learning and Transformer models in classifying Human and LLM-Generated Text. In Proceedings of the 2024 8th International Conference on Computing, Communication, Control and Automation (ICCUBEA), Pune, India, 23–24 August 2024; IEEE: New York, NY, USA, 2024; pp. 1–5.
139. Chan, K.H.; Ke, W.; Im, S.K. A general method for generating discrete orthogonal matrices. *IEEE Access* **2021**, *9*, 120380–120391. [CrossRef]
140. Xing, Z.; Lam, C.T.; Yuan, X.; Im, S.K.; Machado, P. Mmqw: Multi-modal quantum watermarking scheme. *IEEE Trans. Inf. Forensics Secur.* **2024**, *19*, 5181–5195. [CrossRef]
141. Pu, J.; Sarwar, Z.; Abdullah, S.M.; Rehman, A.; Kim, Y.; Bhattacharya, P.; Javed, M.; Viswanath, B. Deepfake text detection: Limitations and opportunities. In Proceedings of the 2023 IEEE symposium on security and privacy (SP), San Francisco, CA, USA, 21–25 May 2023; IEEE: New York, NY, USA, 2023; pp. 1613–1630.
142. Wu, J.; Zhan, R.; Wong, D.; Yang, S.; Yang, X.; Yuan, Y.; Chao, L. Detectrl: Benchmarking llm-generated text detection in real-world scenarios. *Adv. Neural Inf. Process. Syst.* **2024**, *37*, 100369–100401.
143. Topsakal, O.; Akinci, T.C. Creating large language model applications utilizing langchain: A primer on developing llm apps fast. In Proceedings of the International Conference on Applied Engineering and Natural Sciences, Konya, Turkey, 10–12 July 2023; Volume 1, pp. 1050–1056.

144. Cao, S.; Zhang, J.; Shi, J.; Lv, X.; Yao, Z.; Tian, Q.; Li, J.; Hou, L. Probabilistic tree-of-thought reasoning for answering knowledge-intensive complex questions. *arXiv* **2023**, arXiv:2311.13982.
145. Xu, H.; Zhu, Z.; Pan, L.; Wang, Z.; Zhu, S.; Ma, D.; Cao, R.; Chen, L.; Yu, K. Reducing tool hallucination via reliability alignment. *arXiv* **2024**, arXiv:2412.04141. [CrossRef]
146. Liu, B.; Li, X.; Zhang, J.; Wang, J.; He, T.; Hong, S.; Liu, H.; Zhang, S.; Song, K.; Zhu, K.; et al. Advances and challenges in foundation agents: From brain-inspired intelligence to evolutionary, collaborative, and safe systems. *arXiv* **2025**, arXiv:2504.01990. [CrossRef]
147. Bai, G.; Chai, Z.; Ling, C.; Wang, S.; Lu, J.; Zhang, N.; Shi, T.; Yu, Z.; Zhu, M.; Zhang, Y.; et al. Beyond efficiency: A systematic survey of resource-efficient large language models. *arXiv* **2024**, arXiv:2401.00625. [CrossRef]
148. Weidinger, L.; Mellor, J.; Rauh, M.; Griffin, C.; Uesato, J.; Huang, P.S.; Cheng, M.; Glaese, M.; Balle, B.; Kasirzadeh, A.; et al. Ethical and social risks of harm from language models. *arXiv* **2021**, arXiv:2112.04359. [CrossRef]
149. Epstein, Z.; Hertzmann, A.; Investigators of Human Creativity. Art and the science of generative AI. *Science* **2023**, *380*, 1110–1111. [CrossRef] [PubMed]
150. Desai, D.R.; Riedl, M. Between copyright and computer science: The law and ethics of generative aI. *Nw. J. Tech. Intell. Prop.* **2024**, *22*, 55. [CrossRef]
151. Opderbeck, D.W. Copyright in AI training data: A human-centered approach. *Okla. L. Rev.* **2023**, *76*, 951. [CrossRef]
152. Raza, S.; Ghuge, S.; Ding, C.; Dolatabadi, E.; Pandya, D. FAIR enough: Develop and assess a FAIR-compliant dataset for large language model training? *Data Intell.* **2024**, *6*, 559–585. [CrossRef]
153. Pahune, S.; Akhtar, Z.; Mandapati, V.; Siddique, K. The Importance of AI Data Governance in Large Language Models. *Big Data Cogn. Comput.* **2025**, *9*, 147. [CrossRef]
154. Liu, Z. Cultural bias in large language models: A comprehensive analysis and mitigation strategies. *J. Transcult. Commun.* **2025**, *3*, 224–244. [CrossRef]
155. Laakso, A. Ethical Challenges of Large Language Models-a Systematic Literature Review. Master's Thesis, University of Helsinki, Helsinki, Finland, 2023.
156. Caruana, M.M.; Borg, R.M. Regulating Artificial Intelligence in the European Union. In *The EU Internal Market in the Next Decade–Quo Vadis?*; Brill: Leiden, The Netherlands, 2025; p. 108.
157. Fairoze, J.; Garg, S.; Jha, S.; Mahloujifar, S.; Mahmoody, M.; Wang, M. Publicly-detectable watermarking for language models. *arXiv* **2023**, arXiv:2310.18491. [CrossRef]
158. Liu, A.; Pan, L.; Hu, X.; Li, S.; Wen, L.; King, I.; Yu, P.S. An unforgeable publicly verifiable watermark for large language models. *arXiv* **2023**, arXiv:2307.16230.
159. Li, L.; Bai, Y.; Cheng, M. Where am i from? identifying origin of llm-generated content. In Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing, Miami, FL, USA, 12–16 November 2024; pp. 12218–12229.
160. Antoun, W.; Sagot, B.; Seddah, D. From text to source: Results in detecting large language model-generated content. *arXiv* **2023**, arXiv:2309.13322.
161. Russinovich, M.; Salem, A. Hey, That's My Model! Introducing Chain & Hash, An LLM Fingerprinting Technique. *arXiv* **2024**, arXiv:2407.10887.
162. Zeng, B.; Wang, L.; Hu, Y.; Xu, Y.; Zhou, C.; Wang, X.; Yu, Y.; Lin, Z. Huref: Human-readable fingerprint for large language models. *Adv. Neural Inf. Process. Syst.* **2024**, *37*, 126332–126362.
163. Bao, G.; Zhao, Y.; Teng, Z.; Yang, L.; Zhang, Y. Fast-detectgpt: Efficient zero-shot detection of machine-generated text via conditional probability curvature. *arXiv* **2023**, arXiv:2310.05130.
164. Bao, G.; Rong, L.; Zhao, Y.; Zhou, Q.; Zhang, Y. Decoupling Content and Expression: Two-Dimensional Detection of AI-Generated Text. *arXiv* **2025**, arXiv:2503.00258.
165. Koike, R.; Kaneko, M.; Okazaki, N. Outfox: Llm-generated essay detection through in-context learning with adversarially generated examples. In Proceedings of the AAAI Conference on Artificial Intelligence, Vancouver, BC, Canada, 20–27 February 2024; Volume 38, pp. 21258–21266.
166. Li, Y.; Li, Q.; Cui, L.; Bi, W.; Wang, L.; Yang, L.; Shi, S.; Zhang, Y. Deepfake text detection in the wild. *arXiv* **2023**, arXiv:2305.13242. [CrossRef]
167. Min, B.; Ross, H.; Sulem, E.; Veyseh, A.P.B.; Nguyen, T.H.; Sainz, O.; Agirre, E.; Heintz, I.; Roth, D. Recent advances in natural language processing via large pre-trained language models: A survey. *ACM Comput. Surv.* **2023**, *56*, 1–40. [CrossRef]
168. Shumailov, I.; Shumaylov, Z.; Zhao, Y.; Gal, Y.; Papernot, N.; Anderson, R. The curse of recursion: Training on generated data makes models forget. *arXiv* **2023**, arXiv:2305.17493.
169. Yang, S.; Chiang, W.L.; Zheng, L.; Gonzalez, J.E.; Stoica, I. Rethinking benchmark and contamination for language models with rephrased samples. *arXiv* **2023**, arXiv:2311.04850. [CrossRef]
170. Li, D.; Sun, R.; Huang, Y.; Zhong, M.; Jiang, B.; Han, J.; Zhang, X.; Wang, W.; Liu, H. Preference Leakage: A Contamination Problem in LLM-as-a-judge. *arXiv* **2025**, arXiv:2502.01534. [CrossRef]

171. Wang, X.; Antoniades, A.; Elazar, Y.; Amayuelas, A.; Albalak, A.; Zhang, K.; Wang, W.Y. Generalization vs Memorization: Tracing Language Models' Capabilities Back to Pretraining Data. *arXiv* **2024**, arXiv:2407.14985. [CrossRef]
172. Agrawal, A.; Dang, A.; Nezhad, S.B.; Pokharel, R.; Scheinberg, R. Evaluating Multilingual Long-Context Models for Retrieval and Reasoning. *arXiv* **2024**, arXiv:2409.18006.
173. Ghosh, A.; Datta, D.; Saha, S.; Agarwal, C. The Multilingual Mind: A Survey of Multilingual Reasoning in Language Models. *arXiv* **2025**, arXiv:2502.09457.
174. Xu, P.; Ping, W.; Wu, X.; McAfee, L.; Zhu, C.; Liu, Z.; Subramanian, S.; Bakhturina, E.; Shoeybi, M.; Catanzaro, B. Retrieval meets long context large language models. In Proceedings of the The Twelfth International Conference on Learning Representations, Kigali, Rwanda, 1–5 May 2023.
175. Jin, B.; Yoon, J.; Han, J.; Arik, S.O. Long-context llms meet rag: Overcoming challenges for long inputs in rag. *arXiv* **2024**, arXiv:2410.05983. [CrossRef]
176. Villalobos, P.; Ho, A.; Sevilla, J.; Besiroglu, T.; Heim, L.; Hobbhahn, M. Position: Will we run out of data? Limits of LLM scaling based on human-generated data. In Proceedings of the Forty-First International Conference on Machine Learning, Vienna, Austria, 21–27 July 2024.
177. Nelson, E.; Kollias, G.; Das, P.; Chaudhury, S.; Dan, S. Needle in the haystack for memory based large language models. *arXiv* **2024**, arXiv:2407.01437. [CrossRef]
178. Mohammadabadi, S.M.S. From generative ai to innovative ai: An evolutionary roadmap. *arXiv* **2025**, arXiv:2503.11419. [CrossRef]
179. Dai, H.; Pechi, D.; Yang, X.; Banga, G.; Mantri, R. DENIAHL: In-Context Features Influence LLM Needle-In-A-Haystack Abilities. *arXiv* **2024**, arXiv:2411.19360.
180. Hsieh, C.P.; Sun, S.; Kriman, S.; Acharya, S.; Rekesh, D.; Jia, F.; Zhang, Y.; Ginsburg, B. RULER: What's the Real Context Size of Your Long-Context Language Models? *arXiv* **2024**, arXiv:2404.06654.
181. Yue, X.; Song, Y.; Asai, A.; Kim, S.; de Dieu Nyandwi, J.; Khanuja, S.; Kantharuban, A.; Sutawika, L.; Ramamoorthy, S.; Neubig, G. Pangea: A fully open multilingual multimodal llm for 39 languages. In Proceedings of the The Thirteenth International Conference on Learning Representations, Vienna, Austria, 1–7 May 2024.