August 2023

initiative for applied artificial intelligence

A Guide for Large Language Model Make-or-Buy Strategies: Business and Technical Insights

Contents

Executive Summary

Key Business Highlights:

Rational Approaches to Large Language Model Make-or-Buy Decisions

Firms that employ large language models (LLMs) can create significant value and achieve sustainable competitive advantage. However, the decision of whether to **make-or-buy LLMs** is a complex one and should be informed by consideration of strategic value, customization, intellectual property, security, costs, talent, legal expertise, data, and trustworthiness. It is also necessary to thoroughly evaluate available open-source and closed-source LLM options, and to understand the advantages and disadvantages of fine-tuning existing models versus pre-training models from scratch.

Depending on the strategic value and the degree of customization needed, firms have **six possible approaches** to consider when making LLM make-or-buy decisions:

1) **Buy end-to-end application without LLM controllability**
2) **Buy an application with limitedly controllable LLM** – Procure the application including LLM as a component with some transparency and control
3) **Make application, buy controllable LLM** – Internal development of application on top of procured LLMs controllable via APIs
4) **Make application, fine-tune LLM** – Internal development of application and fine-tuning of LLM based on procured or open-source pre-trained LLMs
5) **Make application, pre-train LLM –** Internal development of application and pre-training of LLM from scratch
6) **Stop**

Key Technical Highlights:

Future-shaping Trends for Informed Make-or-Buy Decisions

Beyond fundamental LLM techniques such as the transformer model architecture, pre-training, and instruction tuning, there are important emerging trends that will further enhance LLM performance and adaptability in widespread domain-specific tasks. These include the development of more efficient model architectures and dataset designs, integration of memory mechanisms inspired by cognitive science, incorporation of multimodality, enhancements in factuality, and improved reasoning capabilities for autonomous task completion.

New possibilities to strike balances between open- and closed-source models, and between large and small language models, present promising opportunities. A growing open-source ecosystem is helping organizations to optimize costs and achieve the best outcomes by leveraging the strengths of each type of model. Likewise, smaller language models have demonstrated efficacy in specific tasks, challenging the notion that bigger models are always superior. Embracing this diverse range of models can promote more efficient and effective language model implementation.

Gaining a comprehensive understanding of these trends is vital for firms wanting to make well-informed decisions and avoid misconceptions about LLMs when planning long-term budgets and infrastructure design.

1. Introduction

At the start of this decade, the concept of generative AI was known only to a few enthusiasts and visionaries. Yet in just a few years, it has become increasingly evident that generative AI, and particularly techniques related to Large Language Models (LLMs), are to be a game-changer for individuals, businesses, and wider society.

Generative AI and the latest class of generative AI systems, driven by LLMs such as GPT-4, PaLM-2, and Llama 2, are capable of creating original content by learning from vast datasets. These 'foundation models' generalize knowledge from massive amounts of data and can be customized for a wide range of use cases. Some use cases require minimal fine-tuning and a lower volume of data, while others can be solved by providing just a task instruction with no examples (termed zero-shot learning) or a small number of examples (few-shot learning). These opportunities are empowering developers to build AI applications that were previously impossible and which have the potential to transform industries.

The significance of generative AI and LLMs cannot be overstated. By enabling the automation of many tasks that could previously only be performed by humans, generative AI will significantly increase efficiency and productivity across entire value chains and corporate functions, reducing costs and opening up new and exciting opportunities for growth. A study by McKinsey, for example, estimates that generative AI could add between $2.6 trillion and $4.4 trillion of value to the global economy annually and automate work activities that currently account for 60-70% of employees' time[1]. Firms that do not embrace AI are at risk of falling behind.

With the disruptive and extremely fast-paced acceleration of AI advancement, executives are confronted with some pressing questions: What value do generative AI, and in particular LLMs, have for my business? How can I utilize the benefits of LLMs? What are the risks of embedding LLMs into my organization? And what are LLMs, anyway? Indeed, it is becoming vital to understand how to effectively leverage this technology in products, services, corporate functions and processes, and how to apply LLMs to use cases where significant added value can be achieved.

This white paper seeks to guide readers on how to navigate this new era of LLMs, enabling firms to make rational, informed decisions and achieve sustainable competitive advantage. It is essential to understand both the business and technical aspects of incorporating LLMs into your organization. As such, we here address both aspects by first discussing make-or-buy decisions around the application of LLMs from a business perspective, followed by an overview of critical technical topics, including the latest trends in the field and domain-specific industrial applications of LLMs.

Whatever your company's stage of AI maturity, now is the time to leverage LLMs and drive innovation further.

1 McKinsey and Company (2023). The economic potential of generative AI: The next productivity frontier. https://www.mckinsey.com/capabilities/mckinsey-digital/our-insights/The-economic-potential-of-generative-AI-The-next-productivity-frontier#business-and-society

Q How do you view the impact of the recent trend of generative AI?

A *"Strategically, this has changed the way we work and what our focus areas are. The output quality and ease of use will shape both our professional and our private lives."*

- Dr. Andreas Liebl, Managing Director and Founder, appliedAI Initiative GmbH

Glossary

Generative AI
A field of artificial intelligence that focuses on creating models capable of generating novel content, such as text, code, images, or music, that resembles human-created content.

Foundation Model
A large neural network model that captures and generalizes knowledge from massive data. A starting point for further customization and a fundamental building block for specific downstream tasks.

Large Language Model (LLM)
A powerful neural network algorithm designed to understand and generate human-like language, typically trained on a vast amount of text data and considered a type of foundation model. [See later Info Box 'Large language models as foundation models'].

Transformers
A type of neural network architecture that has revolutionized natural language processing tasks by efficiently capturing long-range dependencies in sequential data such as sentences or paragraphs, making it a suitable building block for large language models.

Pre-training
The initial phase of training a neural network model. The model learns from a large dataset, allowing it to capture general knowledge and patterns.

Fine-tuning
The process of adapting a pre-trained neural network model to perform specific tasks by training it on task-specific data. This allows the model to specialize its knowledge and improve its performance on specific applications.

Few-shot learning
A technique whereby an AI model learns to perform a new task with a small number of examples, making it possible to teach the model something new without needing much training data.

Zero-shot learning
A technique whereby an AI model can understand and perform a task with no specific examples or training on that task, relying instead on general knowledge it has learned from related tasks.

2. To Make or To Buy: Leveraging Large Language Models in Business

2.1. Getting Prepared for Large Language Model Make-or-Buy Decisions

2.1.1. Understanding the Large Language Model Tech Stack

Effectively utilizing LLMs in business requires consideration of several factors that will affect decisions to either leverage external closed-source models via APIs, develop LLMs in-house, or take some form of intermediary approach. There is no clear-cut answer to how to make these decisions but a systematic approach requires taking into account LLMs and their applications and informing make-or-buy decisions by expanding from a sole application perspective to one that encompasses LLMs.

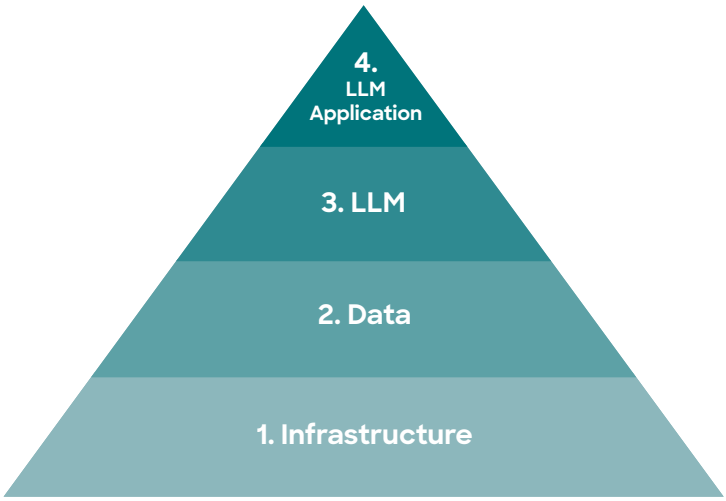


Figure 1. The tech stack for large language models

To achieve this, the first step is to assess which capabilities and internal resources are available and, in turn, which tech stack should be addressed. The LLM tech stack is generally understood to consist of four layers as presented in Figure 1.

The bottom layer is the **infrastructure** required (such as necessary hardware or cloud platforms). This includes the systems and processes needed to develop, train, and run LLMs, such as high-performance computation (HPC) optimized for AI and Deep Learning. Anticipated use cases and their scalability influence the overall infrastructure decision.

The second layer is the **data** volume and quality required. The amount of data needed strongly depends on approaches to use and customization of LLMs (e.g., pre-training vs. fine-tuning), so data quality and data curation are always crucial for LLM success. Firms can invest in data curation and preprocessing techniques such as data cleaning, normalization, and augmentation, to enhance data quality and consistency. Implementing rigorous quality control measures during the data collection and labeling process can also improve data reliability.

On the third layer is the **LLM**, which will eventually form the basis for idiosyncratic applications. LLMs can be open- or closed-source (cf. Chapter 2.1.3. and Chapter 3.1.2.). Firms should aim to create synergies between value-adding use cases as part of a systematic make-or-buy strategy.

The fourth and top layer is LLM **applications.** These applications can either build upon end-to-end applications or rely on an external third-party API. The make-or-buy decision for the application layer depends on the specifics of the lower layers. For example, if a firm lacks high-quality data, then "make" is unlikely to be a feasible option here.

2.1.2. Understanding Key Factors in Large Language Model Make-or-Buy Decisions

Besides the LLM tech stack, there are other factors that should considered in make-or-buy decisions for LLMs, including the following:

1) **Strategic value.** Ensuring that the deployment of LLMs is in line with the overall corporate strategy is of utmost importance in make-or-buy decisions. The main reason for developing an LLM in-house is that it can provide high strategic value with high scalability and value creation, enabling a firm to achieve sustainable competitive advantage. By building LLMs internally, organizations can establish and maintain proprietary knowledge and in-house expertise, creating an intellectual asset. This intellectual property can contribute to long-term competitive advantage as it becomes increasingly difficult for competitors to replicate or imitate. Competitive advantage can also be achieved through LLM fine-tuning, depending on the quality and value of the training data. As fine-tuning approaches are relatively inexpensive, this presents a promising value-creation opportunity for firms with data assets. In contrast, when LLMs are developed and trained externally, they are available to a wider market and available to competitors, meaning no sustainable competitive advantage can be achieved. Moreover, having in-house LLM development capabilities fosters innovation and a culture of continuous learning in that it enables firms to stay at the forefront of technological advancements.

2) **Customization**. Developing LLMs in-house typically allows for greater customization, meaning that LLMs can be tailored to requirements and firm-specific use cases. This point mostly holds for fine-training models with unique internal data. In comparison to off-the-shelf products, customized LLMs allow for greater flexibility while also maintaining full ownership (cf. Chapter 3.2 *"Domain-Specific Application of Large Language Models in Industrial Scenarios"* for more technical information). While using external non-customized LLMs will mean lower costs, it is important to note that potentially sensitive data must be shared with the external partner.

3) **Intellectual property (IP).** LLMs, especially those sourced from the external market, are trained on extensive datasets that may include copyrighted materials or proprietary information. As a result, there may be concerns regarding ownership and usage rights of generated content. Firms must therefore establish clear policies and agreements that address IP rights concerning LLM-generated

content. These policies should outline ownership of content, licensing or usage restrictions, and provisions for protecting sensitive information. Collaborative efforts involving third parties should ensure that these issues are considered during contracting. It should be noted, however, that there is still a great deal of uncertainty around IP rights stemming from content created through generative AI.

4) **Security.** LLMs can require the processing of extremely sensitive business information. Firms should conduct a thorough risk assessment for each use case to ex-ante identify and address potential security issues. For highly sensitive data it is typically recommended to host the LLM within a firm insular network. If this is not possible, collaborating with reputable external LLM providers who adhere to stringent security standards and are transparent about their security practices is crucial. For data falling under the GDPR, firms must ensure that all data is stored and processed on servers within Europe.

5) **Costs.** Developing LLMs in-house is a costly endeavor. It first requires significant investment in terms of hiring a highly-skilled workforce, including ML engineers and NLP specialists, who tend to command high salaries. The development process itself is then time-consuming and resource-intensive, involving extensive research, data collection, model training, and iterative improvement cycles, all of which demand considerable computing power and infrastructure investment. Ongoing maintenance, updates, licenses, and support require continuous investment to ensure optimal performance and reliability. Last, it is important to consider the opportunity costs of allocating internal resources to LLM development over core business activities. While in-house development offers several benefits, it diverts attention and resources from other strategic initiatives and potentially delays time-to-market, which can lead to increased opportunity costs. Executives should therefore carefully evaluate financial implications and weigh costs against potential benefits before deciding to develop LLMs in-house. Fine-tuning may be a more suitable approach in many cases, with substantially lower costs. To address high development costs, organizations could explore ways to streamline the labeling and development cycles. Leveraging pre-existing labeled datasets or partnering with external data providers can reduce the need for extensive manual labeling, saving time and resources. Additionally, adopting cloud-based solutions for data storage and processing can offer scalability and cost-efficiency, enabling organizations to handle large volumes of data more effectively.

6) **Talent.** The scarcity of experienced professionals in fields such as data science, ML, and NLP often make it difficult to establish a skilled in-house team, especially for SMEs confronted with resource constraints. In Europe, the competition for top talent is fierce, with SMEs and large firms alike facing recruitment difficulties and talent shortage. Additionally, extremely rapid development in the field of LLMs necessitates continuous learning and professional development, meaning companies should make significant investments in training and upskilling their workforce. Overcoming these hurdles requires a strategic approach that can include fostering partnerships with academic institutions, collaborating with external partners, offering competitive salaries, and creating a stimulating work environment that promotes innovation. Firms already confronted with talent scarcity may decide to source their LLM solutions from the market to save direct and indirect talent-related costs and to utilize their talent resources for other projects. In-house fine-tuning models often constitute a middle course that can strike a balance between acquiring off-the-shelf products and developing models from scratch.

7) **Legal expertise.** Developing LLMs in-house requires firms to seek legal expertise to navigate an increasingly complex regulatory landscape. For instance, the proposed EU AI Act, which focuses on preventing harm to health, safety, and fundamental human rights, would involve a risk-based approach whereby AI systems would be assigned to a risk class. High-risk systems such as LLMs would need to meet stricter requirements than low-risk systems. Firms pursuing in-house

development of LLMs must ensure they follow all regulatory requirements and thus obtain increasingly complex legal expertise. If this is not available in-house, or if firms want to reduce their general liability, they may instead decide to buy an LLM from the market and ensure the provider is fully liable, i.e., that the specific use case is in line with applicable laws and regulations. Additionally, by considering risk classification early in the decision-making process and making timely decisions, firms can avoid unnecessary expenditures and undesired legal consequences.

8) **Data.** Data is of utmost importance for LLM performance. LLMs rely on vast amounts of diverse data to understand language patterns, enhance accuracy, and generate coherent and appropriate responses. However, biases inherent to the data can pose challenges. For example, LLMs might inadvertently learn and perpetuate biases present in training data. Efforts are being made to identify and mitigate such biases. Diverse and inclusive training data is crucial to ensure fairness and reduce perpetuation or amplification of existing biases, and regular monitoring and user feedback are vital for detecting and rectifying biases. By evaluating LLM outputs and actively seeking user input, developers can improve systems' fairness and mitigate biases. Data is equally important for the process of fine-tuning LLMs. By fine-tuning with domain-specific data, LLMs can acquire specialized knowledge and language patterns related to the target task, enabling them to generate responses that align with the specific requirements of the use case. Moreover, fine-tuning also helps address biases and improve fairness in LLM responses. By fine-tuning with datasets that are explicitly designed to be diverse, inclusive, and representative, developers can reduce biases and ensure that the LLM performs more equitably.

Q What, in your opinion, is the most critical challenge or risk that the European industry needs to address when adopting LLMs for practical use cases?

A *"Among the most critical challenges for the industry when adopting LLMs is the alignment with existing and upcoming regulations, such as the EU AI Act. At the same time, this challenge is also an opportunity to honor our customers' trust in their data with our own standards and approach, and to get them on board with the change. This alignment includes meeting data management requirements, model evaluation, testing, monitoring, disclosure of computational and energy requirements, and downstream documentation. In terms of data privacy, companies from Europe need to be cautious about sharing sensitive data with LLMs hosted by foreign entities and comply with GDPR regulations. To address this challenge, potential mitigation measures include developing robust data anonymization techniques, implementing secure and private computing methods, encouraging local LLM development to reduce reliance on foreign models, and working with regulators to establish clear guidelines and frameworks for the responsible use of AI."*

- Dr. Stephan Meyer, Head of Artificial Intelligence, Munich Re Group

9) **Trustworthiness.** Trustworthiness is of paramount importance when employing LLMs. In-house development of LLMs allows firms to have full control over the entire process, enabling them to build LLMs in line with their values and ethical considerations. This control fosters trustworthiness by ensuring that LLMs are aligned with firms' mission and vision. Moreover, in-house development enables transparency and explainability. Firms can document and communicate development methodologies, data sources, and training processes, allowing users to better understand and evaluate LLM outputs. By mitigating biases and ensuring fairness, firms can build trust among users, assuring them that the LLMs provide accurate and unbiased information. Alternatively, when buying LLMs from the market, especially from established suppliers, firms may benefit from the fact that the acquired LLM has undergone rigorous testing, evaluation, and compliance checks to ensure it meets industry standards and regulatory requirements. Again, the fine-tuning of models often constitutes a compromise between trustworthiness and effort.

Together, these factors should be viewed holistically and acted on as such, rather than being addressed in isolation.

2.1.3. Understanding (Dis-)advantages of Open- vs. Closed-source Large Language Models

Make-or-buy decisions regarding LLMs require thorough evaluation of available options, which include open-source and closed-source LLMs. Generally, the current market environment is dominated by closed-source, API-based LLMs, yet there is an ever-growing number of open-source options. The figure below provides an overview of notable open- and closed-source LLMs released between 2019 and June 2023 [1].

As Figure 2 shows, there is a wide range of options for open-source and closed-source LLMs[1]. Available open-source options tend to allow for greater transparency and auditability over their proprietary counterparts. With open-source models, researchers and developers can access the underlying code, model architecture, and training data, such that they can understand the inner workings of the model and identify potential biases or ethical concerns. Indeed, whereas transparency is a crucial aspect of open-source LLMs, closed-source LLMs are most often a black box with opaque underlying functioning. When a model's code and data are made openly available, developers can scrutinize and verify its behavior, ensuring it aligns with desired ethical standards. This transparency can also help to address concerns about algorithmic biases and discriminatory outputs. Researchers and the wider community can work together to identify and rectify these issues, leading to fairer, more trustworthy language models.

Several prominent open-source LLM initiatives have emerged, each making significant contributions to the field. As well as early versions of OpenAI's GPT (Generative Pre-trained Transformer), an influential open-source LLM initiative is Hugging Face's Transformers library, which provides a comprehensive set of pre-trained models including various architectures such as GPT, BERT, and RoBERTa. The library also offers tools and utilities for training, fine-tuning, and deploying models, making it easier for developers to leverage the power of LLMs in their applications. The Transformers library has gained widespread popularity due to its user-friendly interface, extensive documentation, and support from a vibrant community. Several other open-source LLM projects and libraries exist, such as Fairseq, Tensor2Tensor, and AllenNLP.

1 See also Chapter 3.1. for a more comprehensive analysis as well as detailed lists of available options from a technical perspective, in particular for the trend of maximizing the benefits by incorporating both large closed-source LLMs and a combination of large and small, specialized open-source LLMs.

In turn, closed-source LLMs often leverage significant computational resources and proprietary datasets during their training, allowing them to perform at extremely high levels on a range of language tasks. The investment in infrastructure and data acquisition made by companies can result in LLMs that surpass the capabilities of open-source models. However, firms are especially concerned about data protection and information security when closed-source LLMs are running as software as a service (API-based model), an approach increasingly used by vendors. Customization of closed-source models means that firms need to transfer their often highly sensitive data to the vendor for fine-tuning.

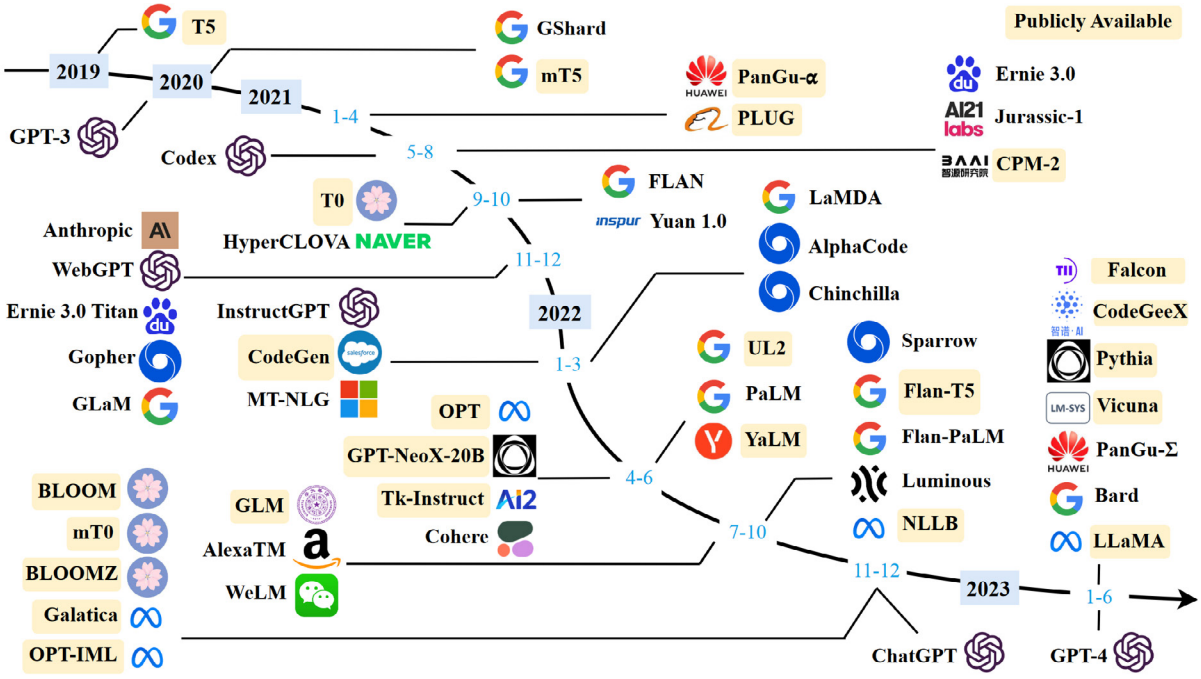


Figure 2. Open-source and closed-source large language models with over 10 billion parameters released between 2019 and June 2023 [1]

2.1.4. Understanding (Dis-)advantages of Fine-tuning vs. Pre-training Models from Scratch

Another critical aspect in make-or-buy decisions regarding LLMs relates to an in-depth understanding of the advantages and disadvantages of fine-tuning existing models versus pre-training models from scratch, specifically considered from a business perspective.

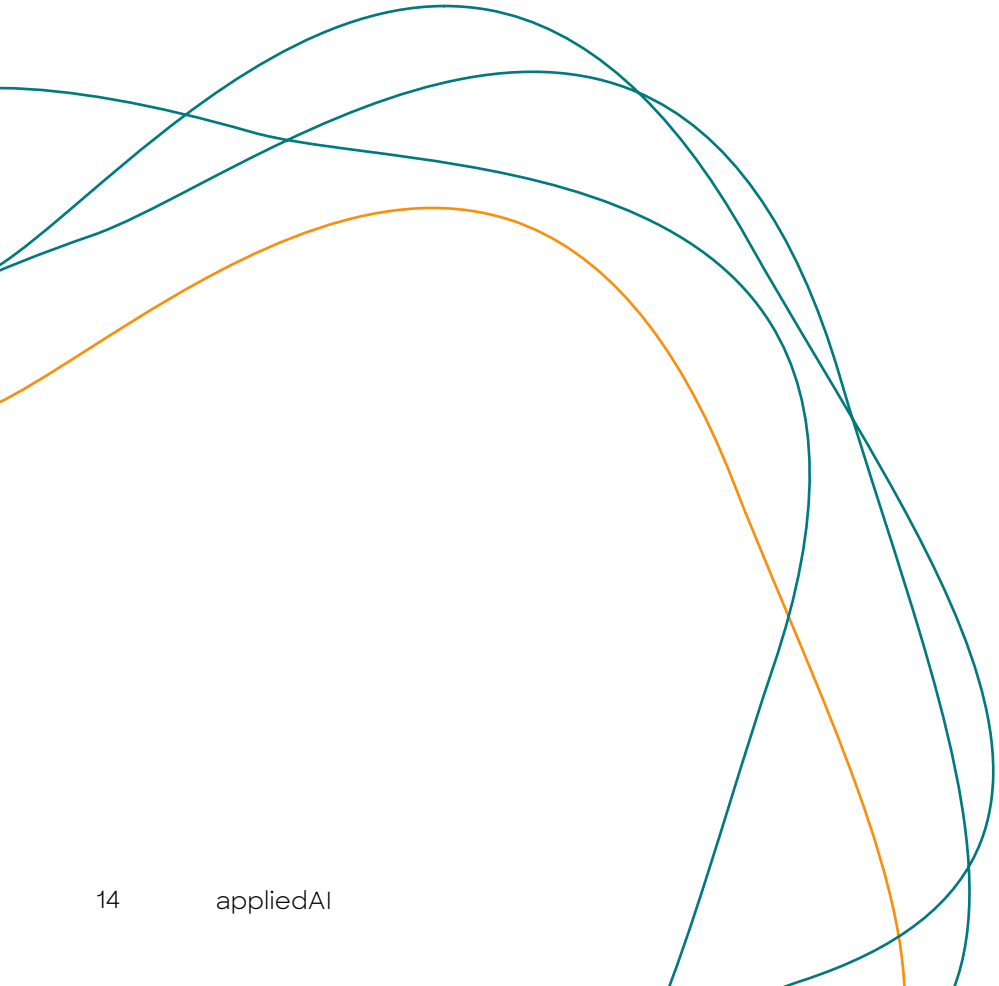
Fine-tuning pre-trained LLMs generally incurs significantly lower costs compared to building them from scratch. Depending on the underlying data structure and volume, fine-tuning costs can be relatively low, ranging from a few hundred to a few thousand US dollars. In fine-tuning, a pre-trained model is already available, eliminating the need for resource-intensive pre-training on vast amounts of data and large amounts of computational power. This translates to significant savings in resources, time, and electricity consumption.

Conversely, pre-training LLMs from scratch involves substantial costs at various stages of the process which combined can reach millions of dollars. For example, the training costs for OpenAI's GPT-3 are estimated to be $5 million, while models with more

training parameters are estimated to exceed these costs. Pre-training LLMs from scratch demands an enormous amount of computational power, specialized hardware, and extensive infrastructure, all of which add heavy costs. Another consideration is that the pre-training process can take weeks or even months to complete, adding to the costs of computational resources and electricity.

There are also notable differences in data acquisition and annotation costs. Fine-tuning LLMs typically requires a smaller labeled dataset for the target task, which can be less expensive to obtain, annotate, and curate than the comprehensive and diverse datasets required for pre-training an LLM from scratch. The costs of acquiring and labeling a large-scale dataset can be substantial, and manipulation of such assets requires substantial domain expertise and significant human effort.

Overall, then, there are usually cost advantages to fine-tuning LLMs compared to pre-training them from scratch. However, it is essential to consider the specific requirements of each use case, including the scale of the target task, availability of data, and potential risks, to determine the most appropriate approach based on available resources and objectives. Ultimately, decisions about this question will depend on the business cases and financial resources a firm is willing to invest. See also Chapter *3.2. Domain-Specific Application of Large Language Models in Industrial Scenarios* for relevant discussions from a technical perspective.



2.2. Approaches for Large Language Model Make-or-Buy Decisions

After acknowledging the LLM tech stack and relevant key factors and business considerations, there are six generic approaches that firms can follow when making LLM make-or-buy decisions:

1) **Buy end-to-end application without LLM controllability**
   When evaluating use cases of low strategic value and limited customization requirements for both the application and the LLM, acquiring a pre-built end-to-end application is typically the most convenient solution, with the LLM operating merely as a hidden component. Given the highly tailored nature of the LLM to the application and its scope, explicit customization and controllability are unnecessary and likely not allowed by the vendor.

2) **Buy an application with limitedly controllable LLM – Procure the application including the LLM as a component with some transparency and control**
   This approach of procuring an application along with controllable LLMs applies to use cases that demand minimal adjustments or can be deployed immediately. It is worth noting that in scenarios where customization needs are low, it may be less necessary to control the underlying LLM and companies might instead focus on adapting only the user layer to meet their requirements. Nevertheless, case-specific requirements concerning the degree of customization, regulation, data security/secrecy, intellectual property (IP) concerns, and overall performance should be carefully considered. Another point of attention is the reusability of an LLM across applications in the company and how this might produce undesired dependencies and vendor-locking scenarios. This approach is only feasible in cases of low data confidentiality allowing transfer to external providers.

3) **Make application, buy controllable LLM – Internal development of application on top of procured LLMs via APIs, e.g., Azure OpenAI Services**
   An alternative to the above approach is to focus exclusively on the internal development of the application while sourcing and integrating externally sourced pre-trained or fine-tuned LLMs. This approach is particularly suitable for use cases that demand medium to high levels of LLM customization and is especially relevant when internal resources such as computing power, capacity, or skills are not sufficiently available. Additionally, budget constraints can also drive the decision to adopt this strategy. However, as with approach 2, considerations regarding customization, regulation, data security/secrecy, and IP, as well as overall performance and model reusability, need to be carefully taken into account, and vendors should be carefully scrutinized.

4) **Make application, fine-tune LLM – Internal development of application and fine-tuning of LLM based on procured or open-source pre-trained LLMs**
   This approach involves utilizing existing pre-trained LLM models, along with specific fine-tuning frameworks or services, and combining them with internal development efforts to build applications and fine-tune models using internal data for targeted use cases. The quantity and quality of open-source pre-trained LLMs are continuously rising, but the licenses of these pre-trained models can impose significant limitations on their commercial use. For fine-tuning, several providers such as AWS, Google, NVIDIA, H2O, and others already offer such services, and various open-source fine-tuning services are already available. The level of internal development required depends on both the sophistication of the fine-tuning components and the quality of the underlying pre-trained LLM, as well as the availability of in-house data. While fine-tuning models is comparatively inexpensive, data quality is often a major bottleneck. Nevertheless, this approach offers a viable option for achieving sufficient customization and quality of LLMs, while maintaining control over internal data processing and LLM hosting. This can become particularly important in certain use cases, ensuring sustainable competitive advantage.

5) **Make application, pre-train LLM – Internal development of application and pre-training of LLM from scratch**
This approach involves full end-to-end development ("make"), building the application itself as well as pre-training LLMs in-house from scratch. The broader the applicability of an LLM and the greater the value it can generate, the better it is to pursue the "make" approach. This option is also advisable in highly sensitive use cases where relying on externally sourced models is not an option. Although very costly, developing LLMs from scratch might be the best option for achieving optimal customization and quality, and for ensuring a sustainable competitive advantage.

6) **Stop**
If the use case holds limited strategic value, it is advisable to assign resources to use cases of higher strategic significance.

Figure 3 provides a guide of which approach to use, organized by level of strategic value of an application and the degree of customization needed.

Q What are your thoughts on the potential impact of large language models in the semiconductor industry, and how do you see that affecting your company?

A *"In the semiconductor industry there are main value potentials: improving our processes and creating customer value. One area where this potential can be realised is in knowledge retrieval throughout research and development and manufacturing processes, leading to enhanced speed and stability, for example in the case of equipment maintenance. This reduces our dependency on specific experts with the right domain knowledge being present 24/7 to solve critical issues and helps us train new experts faster. Moreover, by providing top-notch customer support for our highly technical products, we can deliver a better customer experience while increasing the scalability associated with such service. Additionally, there is significant room for improving productivity in support functions, ranging from generating product documentation to marketing and beyond -- lots of potential."*

- Simon-Pierre Genot, Senior Manager AI Strategy, Infineon Technologies

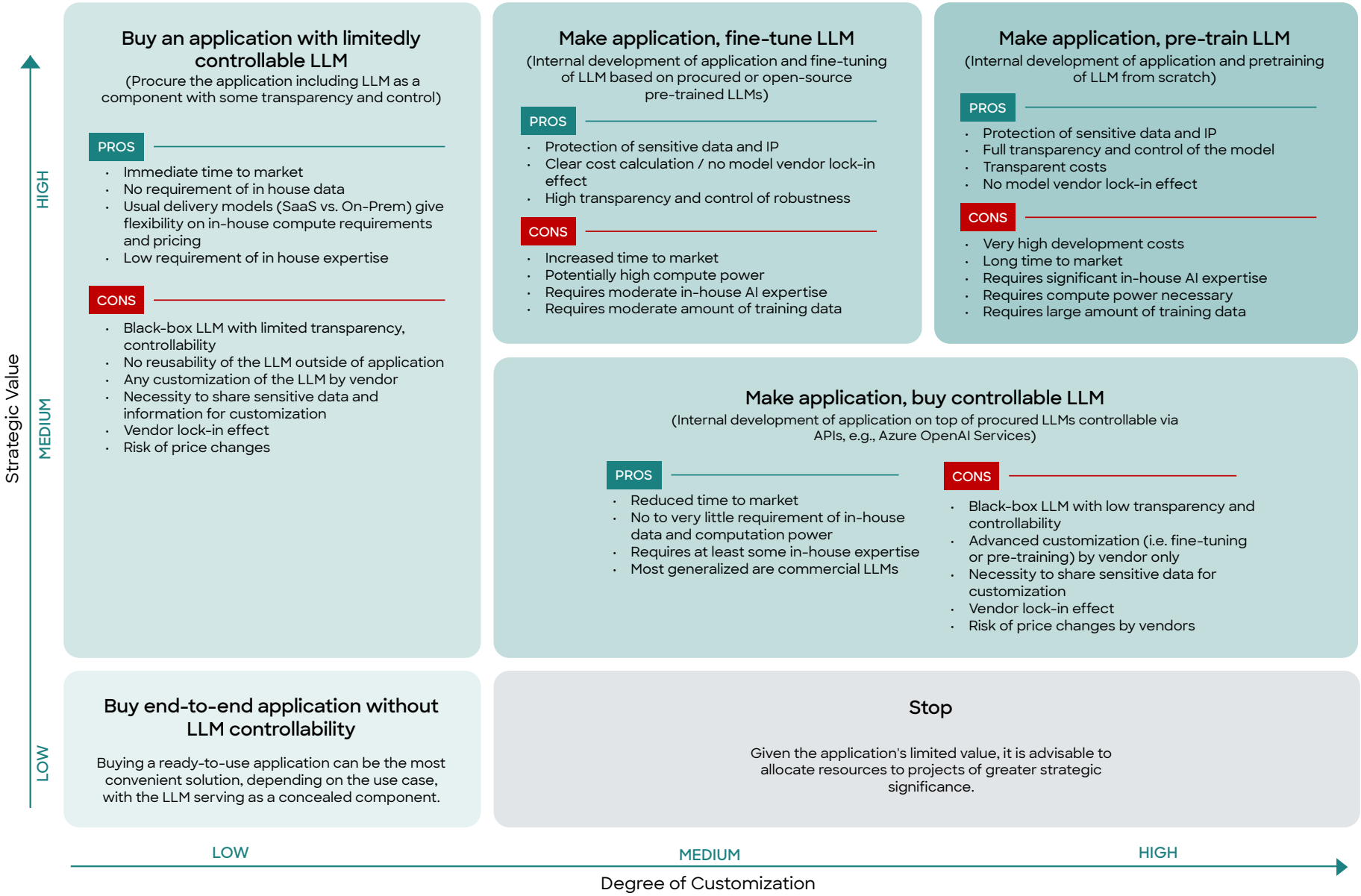


Figure 3. Pros and cons of in-house LLM application development (“make or buy”)

3. Critical Techniques and Trends in the Field of Large Language Models: From Landscape to Domain-specific Applications

3.1. Navigating the Landscape of Large Language Models in the Generative AI Era

3.1.1. Key Techniques, Architectures, and Types of Data

LLMs are an integral part of the generative AI era. They are complex systems that can process natural language input and generate human-like responses. Navigating the landscape of LLMs in this era requires an understanding of key techniques as well as the types of data used in these models. In this section, we will discuss some of the fundamental aspects of LLMs that enable them to function seamlessly, before describing some key trends observed in this fast-developing field.

Transformer as the Base Architecture that Handles Contextual Meanings

One of the most popular techniques used in LLMs is the transformer model architecture, introduced by Vaswani *et al.* in 2017 [2]. Transformers are neural networks that can process sequences of data such as text while being able to handle long-range dependencies and understand context. They do this by implementing an 'attention' mechanism that allows the model to process an entire input sequence all at once and capture the relative importance of each input token to every other token in the context. This enables the LLM to understand the complicated relationships between words, phrases, etc., even when they are far apart in the input sequence. Furthermore, the transformer architecture offers a key advantage over previous recurrent neural network models in that it is highly parallelizable, facilitating large-scale training on distributed hardware. The basic transformer architecture has been used or adapted in some of the most powerful and popular LLMs, such as GPT-3, T5, and BERT.

Pre-training as a Key Procedure to Equip the Model with Fundamental Knowledge

Another key technique used in LLMs is pre-training, which involves training a model on a large corpus of text data before fine-tuning it on a specific task. This technique has been shown to improve the performance of LLMs on a variety of downstream tasks such as translating languages, answering questions, and generating text. Pre-training can be conducted using a variety of objectives including language modeling, where the model is trained to predict the next word in a sequence, and masked language modeling, where some of the input tokens are masked and the model must predict their original values.

Instruction Tuning & RLHF: Aligning with Human Preference

Instruction tuning is a fundamental concept in training LLMs. Early work focused on fine-tuning LLMs on various publicly available NLP datasets and evaluating their performance on different NLP tasks. More recent work, such as OpenAI's InstructGPT, has been built on human-created instructions and demonstrates success in processing diverse user instructions [3] Subsequent works like Alpaca and Vicuna have explored open-domain instruction fine-tuning using open-source LLMs. Alpaca, for example, used a dataset of 50k instructions, while Vicuna leveraged 70k user-shared conversations from ShareGPT.com. These efforts have advanced instruction tuning and its applicability in real-world settings.

Another technique, Reinforcement Learning from Human Feedback (RLHF), aims to use methods from reinforcement learning to optimize language models with human feedback [4]. Its core training process involves pre-training a language model, training a reward model, and fine-tuning the language model with reinforcement learning. The reward model is calibrated with human preferences and generates a scalar reward that represents these preferences. While RLHF is promising, to date it has notable limitations such as the potential for models to output factually inaccurate text.

Types of Data

LLMs are typically trained on extensive datasets primarily composed of textual material from web pages, books, and social media. However, as will be explained in a later section, they can also utilize data from other sources as long as it can be converted to a sequence of tokens with a known set of 'vocabulary'. Hence LaTeX formulas, musical notes, and programming languages like

Python, Java, and C++ may all be adopted as training data [5]-[7]. This enables the model to generate novel mathematical or physical formulas, reason with them, compose music, and generate code to address bugs and enhance program efficiency, thereby streamlining the development process. Additionally, LLMs can leverage SMILE or SELFIES chemical structures for drug design, DNA or protein sequences for predicting protein structures, or genetic mutations related to diseases [8]-[11].

The scope extends further to encompass various other modalities like audio, video, signal data (such as wireless network signals or depth sensing signals) [12]-[16], relational or graph database data (such as stock prices or knowledge graphs) [17], [18], as well as digital signatures and file bytes (such as blockchain transactions or image file bytes) [19],[20]. This huge range of usable data sources allows the models to perform tasks such as speech recognition, action recognition, video summarization, robotic movement planning, knowledge graph completion, stock price prediction, blockchain transaction, or wireless network transmission anomaly detection, as well as image classification. While training models on diverse data types can pose challenges related to pre-processing and standardization, it offers significant benefits as it can unlock new applications and solutions across various domains. The ability to process and generate sequential data from multiple modalities expands the potential impact and use cases of LLMs, fostering innovation and problem-solving in numerous fields (Figure 4).

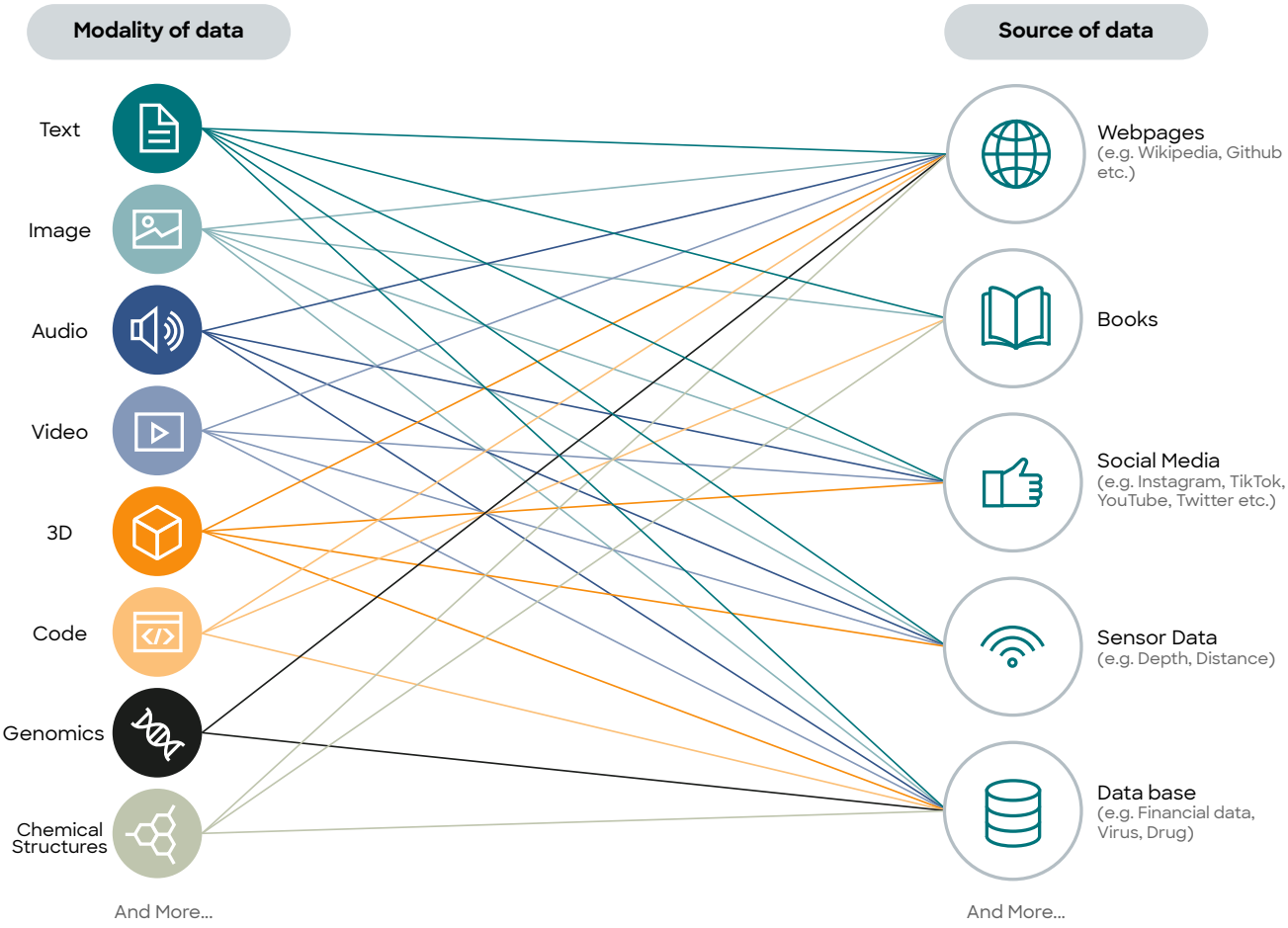


Figure 4. Sample data modalities and data sources involved in recent large language models. Note that both the types of data modalities and the types of data sources are continuously increasing.

Large Language Models as Foundation Models

LLMs possess the remarkable ability to generalize knowledge across diverse contexts, aligning them closely with the concept of ***foundation models*** [21],[22]. Foundation models capture relevant information as a versatile "foundation" for various purposes, distinguishing them from traditional approaches. They demonstrate the characteristic of ***emergence,*** with behaviors implicitly induced rather than explicitly constructed. LLMs excel in solving diverse tasks that go beyond their original language modeling training [23],[24]. These tasks can be accomplished just using natural language prompts, without the need for explicit training. This ***in-context learning*** capability allows LLMs to perform tasks such as machine translation, arithmetic, code generation, answering questions, and more [25],[26]. In a ***zero-shot learning*** scenario, the model relies solely on the task descriptions given in the prompt [27]-[30], while in a ***few-shot learning*** scenario, a small number of correct answer samples are incorporated into the prompts [31]-[33]. Meanwhile, the use of ***chain-of-thought (CoT)*** prompting, which provides step-by-step instructions to guide the model's answer generation, has been shown to boost the model's reasoning capabilities and overall performance [34]-[36]. These highlight the generality and adaptability of LLMs as foundation models.

Homogenization is another key characteristic of foundation models and refers to the unifying and consolidating of methodologies across modeling approaches, research fields, and modalities [21]. For example, model architectures such as BERT, RoBERTa, GPT, and others have been adopted as the base architecture for most state-of-the-art NLP models. This trend extends beyond the field of natural language processing, with similar transformer-based approaches being applied in diverse domains such as DNA sequencing and chemical molecule generation. In addition, based on similar principles, foundation models may be built across modalities. Multimodal models, which combine data in the form of texts, audio, images, etc., offer a valuable fusion of information for tasks spanning multiple modes. This convergence of methodologies and models has streamlined disparate techniques, leveraging the power of transformers as a core component. Homogenization has facilitated cross-field research, enabling LLMs to excel in diverse applications such as drug discovery, robotic reasoning, and media generation. Foundation models provide a base of generalized knowledge that transcends specific tasks and domains, revolutionizing the generative AI landscape.

To summarize, LLMs are powerful neural network algorithms in the field of natural language processing. Key techniques used in LLMs include transformer architecture, pre-training, instruction tuning, and RLHF. LLMs are trained on massive amounts of data gathered from a huge range of sources and modalities. As foundation models, they are proficient at generalizing knowledge from vast amounts of text and showing zero- or few-shot learning capabilities as well as impressive reasoning skills, particularly when combined with techniques like chain-of-thought prompting. LLMs can accurately complete a wide range of tasks including understanding language, generating text, and handling diverse types of sequences. Understanding the techniques, architectures, and types of data used in LLMs as well as their characteristics as foundation models is essential for navigating the current and future landscape of generative AI.

Beyond the Fundamentals: Key Trends That Shape the Future

In the ever-evolving realm of LLMs, several key trends have emerged to resolve previous inadequacies such as heavy costs, hallucinations, and reasoning fallacies. These limitations have posed considerable challenges to the industrialization of LLMs. Consequently, the research and development related to these trends will play a pivotal role in expanding LLM utilization. The trends surpass foundational aspects and provide fresh perspectives into the evolving characteristics of LLMs, unlocking exciting opportunities for exploration and innovation, and laying the groundwork for future advancements.

Efficient Model Architectural Design

A significant recent advancement in LLM research pertains to enhancing model efficiency. Efforts have been made to reduce time and space complexities associated with LLMs. One such innovation is Receptance Weighted Key Value (RWKV), which optimizes model architecture and resource utilization without compromising performance [37]. Another notable trend relevant to model architecture design regards techniques that allow models to efficiently handle longer input sequences (e.g., LongNet [38], Unlimiformer [39], mLongT5 [40]), thereby enabling LLMs to process and understand more comprehensive and context-rich information at once.

Effective and Precise Dataset Creation

Another burgeoning area of focus is the effective generation of training and instruction tuning data, leveraging methods such as WizardLM to evolve complex instructions from simple ones, enhancing the speed of data generation as well as the diversity of the contents [41]. Other approaches like MiniPile [42] or INGENIOUS [43] aim to achieve competitive performance with a small number of examples. Additionally, the innovative approach of Domain Reweighting with Minimax Optimization (DoReMi) estimates the optimal proportion of language from different domains in a dataset, such that LLMs can better adapt to diverse data sources and enhance their capacity for generalization [44].

Reconsideration of Model Scaling Laws: Bigger ≠ Better

The LLM field has traditionally emphasized a positive correlation between model scale and performance improvement. Yet recent studies challenge this notion by presenting evidence of inverse scaling, whereby increased model size leads to worse task performance [45] (Figure 5). This phenomenon arises due to factors including undesirable patterns in the training data and deviation from a pure next-word prediction task. These findings have sparked a shift in understanding the behavior of larger-scale models and have highlighted the need for careful consideration of training objectives and data selection. Relatedly, exploration of smaller language models (SLMs) [46]-[48] has demonstrated their efficacy in specific tasks such as procedural planning and domain-specific question-answering. Approaches like PlaSma focus on equipping SLMs with procedural knowledge and counterfactual planning capabilities, enabling them to rival or surpass the performance of larger models [49]. Similarly, Dr. LLaMA leverages LLMs to enhance SLMs through generative data augmentation, yielding improved performance in domain-specific question answering tasks [50]. These developments challenge the conventional belief that bigger models are inherently superior and highlight the importance of carefully tailored data and objectives for training language models. By adopting a more nuanced understanding of model scaling laws, researchers and practitioners can harness the potential of smaller as well as larger language models to meet the demands of diverse applications and domains.

Alternative Alignment Approaches

Another focus of current research is how best to align LLMs with human preferences, with the goal of improving model performance and interaction quality. Traditional approaches such as the aforementioned Reinforcement Learning from Human Feedback (RLHF) have relied on optimizing LLMs using reward scores from a human-trained reward model [3], [4]. These approaches have shown effectiveness, but come with computational complexity and heavy memory requirements. Recent advancements introduce approaches

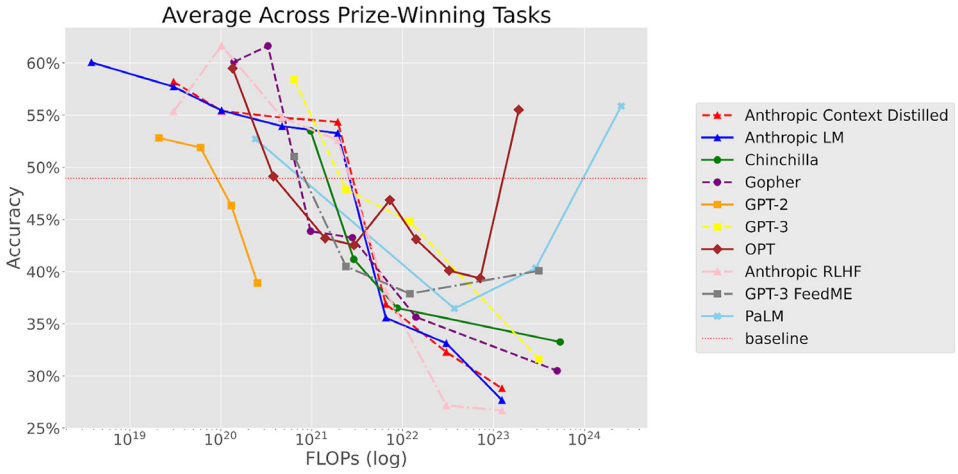


Figure 5. Larger models may not necessarily perform better for tasks deviating from next-word prediction. FLOPs correspond to the amount of computation consumed during model pre-training, which correlates with model size as well as factors such as training time or data quantity. Training FLOPs are used rather than model size alone because computation is considered a better proxy for model performance in the original paper[45].

such as Sequence Likelihood Calibration with Human Feedback ([51]) and Reward Ranking from Human Feedback (RRHF) [52], which address earlier shortcomings by calibrating a language model's sequence likelihood through ranking of desired versus undesired outputs. Another method, termed Less Is More for Alignment (LIMA) [53], aims to achieve comparable performance without reinforcement learning by more efficiently fine-tuning models on only 1,000 carefully curated prompts and responses. These examples present a simpler and more efficient approach to aligning LLM output probabilities with human preferences, facilitating integration of LLMs into practical applications and enhancing their value.

Incorporation of Cognitively Inspired Memory Mechanisms

Yet another emerging trend in this field is the incorporation of cognitively inspired memory mechanisms into LLMs, which takes inspiration from current understanding of human memory functioning [54]–[56]. This development aims to improve training efficiency, generalization across tasks, and long-term interaction capabilities. For example, to address the forgetting phenomenon, in which a model's performance on previously completed tasks deteriorates, researchers have proposed Decision Transformers with Memory (DT-Mem) which integrates an internal working memory module into LLMs [57]. By storing, blending, and retrieving information for different tasks, this proposed mechanism enhances training efficiency and generalization. Researchers are also investigating deficiencies of long-term memory in LLMs, referring to models' limited capacity to sustain interactions over extended periods. One proposed solution is MemoryBank, a novel memory mechanism tailored for LLMs [58]. Inspired by the Ebbinghaus Forgetting Curve theory, MemoryBank enables LLMs to summon relevant memories and continuously update their memory based on time elapsed and the significance of the memory. By emulating human memory storage mechanisms and allowing for long-term memory retention, LLMs could overcome the limitations of forgetting and sustain meaningful longer-term interactions.

Magnifying Multimodality

As described earlier, a clear trend in the continuously evolving field of LLMs is the incorporation of more and more modalities and the improvement of multimodal training [14], [36], [59]–[61]. Researchers have developed approaches like ImageBind, which learns a joint embedding across multiple

modalities such as images, text, audio, depth, thermal, and inertial measurement unit data, making cross-modal retrieval, composition, detection, and generation possible [62]. ULIP-2, a multimodal pre-training framework, addresses scalability and comprehensiveness issues in gathering multimodal data for 3D understanding by leveraging LLMs to automatically generate holistic language counterparts [63]. It has achieved remarkable improvements in zero-shot classification and real-world benchmarks without manual annotation efforts. Such advancements expand LLM capabilities, enabling them to understand and generate across multiple modalities and perform complex tasks in diverse domains.

From Explainability to Tractability and Controllability

Novel approaches have also been developed to enhance the explainability, tractability, and controllability of LLMs and relevant applications [64]–[67]. For example, Control-GPT leverages the precision of LLMs like GPT-4 in generating code snippets for text-to-image generation [68]. By querying GPT-4 to write graph-generating codes and using the generated sketches alongside text instructions, Control-GPT enhances instruction-following and greatly improves the controllability of image generation. Another approach, Backpacks, introduces a neural architecture that combines strong modeling performance with interpretability and control [69]. Backpacks learn multiple sense vectors for each word and represent a word as a context-dependent combination of sense vectors, allowing for interpretable interventions to change the model's behavior. Additionally, GeLaTo proposes using tractable probabilistic models, such as distilled hidden Markov models, to impose lexical constraints in autoregressive text generation [70]. GeLaTo achieves state-of-the-art performance on constrained text generation benchmarks, surpassing strong baselines. Advances like these not only provide insights into the workings of LLMs but also enable greater control and customization, enhancing their performance in computer vision and text generation tasks.

Hallucination Fixes, Knowledge Augmentation, Grounding, and Continual Learning

One of the most prominent trends in recent research is the concerted effort to tackle hallucination and factual inaccuracy, two major stumbling blocks to LLM industrialization [71]–[74]. Researchers have pursued multiple approaches to tackle these problems [75]–[86]. One approach involves analyzing and mitigating self-contradictions in LLM-generated text by designing frameworks that constrain LLMs to generate appropriate sentence pairs [87]. Another aims to enhance the factual correctness and verifiability of LLMs by enabling them to generate text with citations [88]. This involves building benchmarks for citation evaluation and developing metrics that correlate with human judgment.

Additionally, researchers have introduced frameworks that augment LLMs with structured or graph knowledge bases (‘grounding’) to improve factual correctness and reduce hallucination. One approach, Chain of Knowledge (CoK), incorporates structured knowledge bases that provide accurate facts and reduce hallucination [89]. Another technique, Parametric Knowledge Guiding (PKG) [84], equips LLMs with a knowledge-guiding module that accesses relevant knowledge at runtime without modifying the model's parameters. These advances in hallucination avoidance, knowledge augmentation, grounding, and continual learning contribute to improving the reliability and accuracy of generated text across domains and tasks.

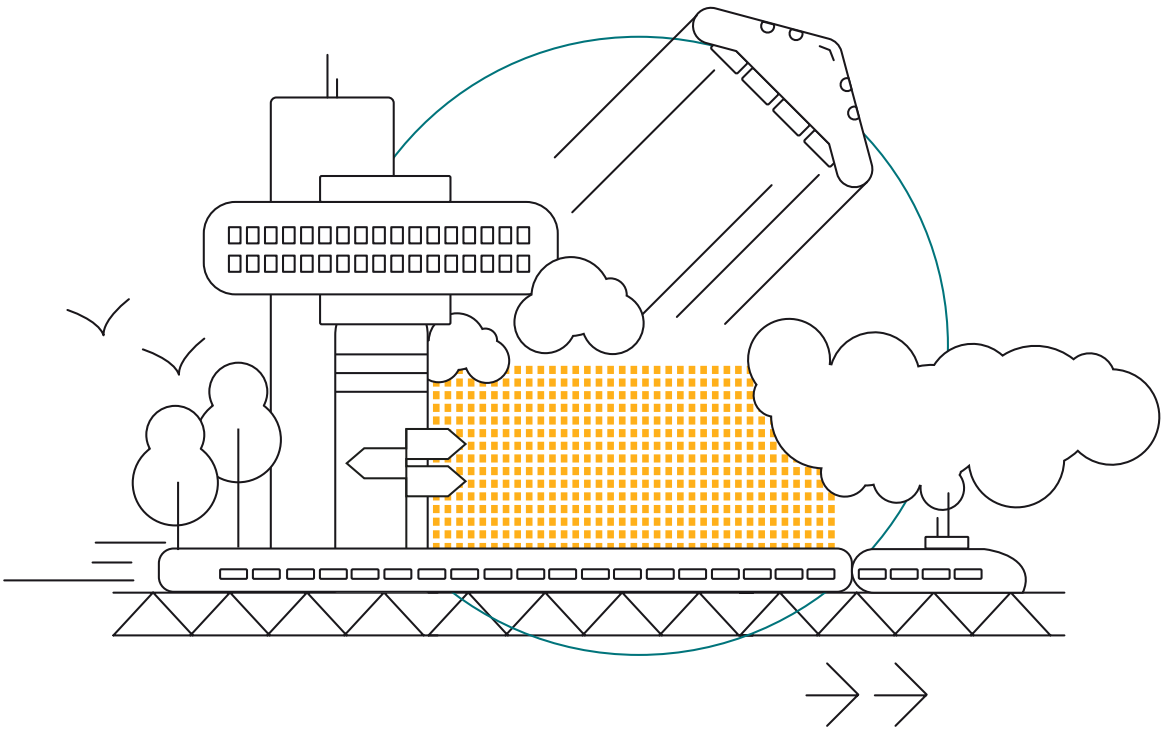
Human-like Reasoning and Problem Solving

This trend focuses on enhancing the reasoning ability of LLMs [90]–[97]. Researchers have introduced innovative frameworks such as Tree of Thoughts (ToT), which enable LLMs to explore and strategically plan intermediate steps toward problem-solving [98]. This approach encourages LLMs to make deliberate decisions, evaluate choices, and consider multiple reasoning paths, rather than just a single one. Another proposed method, Self-Notes, allows LLMs to deviate from the input context, enhancing context memory

and enabling multi-step reasoning [99]. Additionally, OlaGPT introduces a framework to simulate human cognitive abilities, including attention, memory, reasoning, and learning [100]. OlaGPT incorporates an active learning mechanism to strengthen problem-solving abilities by recording and referring to previous mistakes and expert opinions. These developments in reasoning abilities pave the way for LLMs to tackle complex problems more effectively, bridging the gaps between their current capabilities and human reasoning.

LLM-guided Artificial General Intelligence

Researchers have also recently endeavored to develop artificial general intelligence on top of LLMs [101]–[103]. Voyager, an embodied lifelong learning agent powered by LLMs, autonomously explores and acquires skills in Minecraft without human intervention [104]. It uses an automatic curriculum, an ever-growing skill library, and an iterative prompting mechanism to enhance its abilities. Voyager demonstrates exceptional proficiency in Minecraft, outperforming prior state-of-the-art methods on various metrics. Another approach, LLMs as Tool Makers (LATM), allows LLMs to create their own reusable tools for problem-solving, eliminating dependency on existing tools [105]. LATM consists of two phases, tool making and tool using, which together enable LLMs to generate tools for different tasks and achieve cost effectiveness. LATM has been validated across complex reasoning tasks. Additionally, Augmenting Autotelic Agents with Large Language Models (LMA3) introduces a language model augmented autotelic agent that leverages a pre-trained language model to represent, generate, and learn diverse and abstract human-relevant goals [106]. LMA3 demonstrates the ability to learn a wide range of skills without hand-coded goal representations or curricula in a text-based environment. Such innovations promote the development of artificial general intelligence by empowering LLMs to autonomously acquire skills, create tools, and pursue diverse goals.



3.1.2. Major Closed-source Models and Open-source Alternatives

In the landscape of LLMs, there are several major closed-source (often proprietary) models and a growing number of open-source alternatives that offer powerful capabilities for various natural language processing tasks. These models have been developed by leading industry players, open-source developers, and research institutions, and they continue to push the boundaries of what LLMs can achieve. In this section, we will explore some prominent closed-source models and the growing area of open-source alternatives.

Closed-source Models

Prior to GPT-3, most LLMs were openly available. However, with GPT-3 and similar models that excel in next word prediction, there has been a shift towards proprietary closed-source models. These are predominantly developed by major industry players such as OpenAI, Google, and Microsoft. Table 1 presents a selected list of these models.

ChatGPT is often considered a service rather than a standalone model as it incorporates GPT-3.5 or GPT-4 (for the Plus version). Likewise, Google's experimental conversational AI service Bard initially utilized a lightweight and optimized version of LaMDA (Language Model for Dialogue Application) but later transitioned to a more advanced language model called PaLM 2. Bing Chat, powered by a customized version of OpenAI's ChatGPT, integrates Microsoft's search engine to deliver human-like conversational responses and improve overall user experience. Another commercial chatbot, ERNIE Bot, is built upon Ernie 3.0-Titan. These conversational AI services, not listed in Table 1, build upon proprietary closed-source models to provide contextually relevant conversations and deliver engaging user experience.

Table 1. Selected list of closed-source models after 2020.

Country	Developer & Provider	Model	Parameters	Release
US	OpenAI	GPT-3	175B	Jun 2020
US	OpenAI	InstructGPT	1.3B, 6B, 175B	Jan 2022
US	OpenAI	GPT-3.5	175B	Mar 2022
US	OpenAI	GPT-4	Unknown	Mar 2023
US	Microsoft	phi-1	1.3B	Jun 2023
US	Google	LaMDA	137B	May 2021
US	Google	GLaM	1.2T	Dec 2021
US	Google	PaLM	540B	Apr 2022
US	Google	PaLM-E	562B	Mar 2023
US	Google	PaLM-2	340B	May 2023
US/UK	Google DeepMind	Gopher	280B	Dec 2021
US/UK	Google DeepMind	Chinchilla	70B	Mar 2022
US	Amazon	AlexaTM	20B	Aug 2022
US	NVIDIA	Megatron Turing NLG	530B	Oct 2021
US	Bloomberg	BloombergGPT	50B	Mar 2023
US	Anthropic	Claude	52B	Dec 2021
US	Anthropic	Claude 2	Unknown	Jul 2023
US	Cohere	Cohere	Unknown	Nov 2021
China	Baidu & Peng Cheng Lab.	ERNIE 3.0 Titan	260B	Dec 2021
China	Beijing Academy of Artificial Intelligence	Wu Dao 2.0	175B	May 2021
China	Huawei	PanGu-Σ	1T	Mar 2023
Israel	AI21	Jurassic-1	178B	Sept 2021
Israel	AI21	Jurassic-2	Unknown	Mar 2023
South Korea	Naver Corp	HyperCLOVA	204B	May 2021
Germany	Aleph Alpha	Luminous	13B, 30B, 70B	Nov 2021

Open-Source Alternatives

In the first half of 2023 especially, there has been a surge of open-source LLMs, paving the way for fresh avenues of innovation and collaboration. In the early stages of this surge, the open-source landscape consisted mostly of research-only models such as LLaMA, Alpaca, and their subsequent iterations, including Dolly 1.0, GPT4All, GALPACA, Baize, Koala, Vicuna, LLaVA, WizardLM, StableVicuna, ImageBind, etc. These models allowed researchers to study and explore the capabilities and potentials of LLMs (Figure 6 and Table 2).

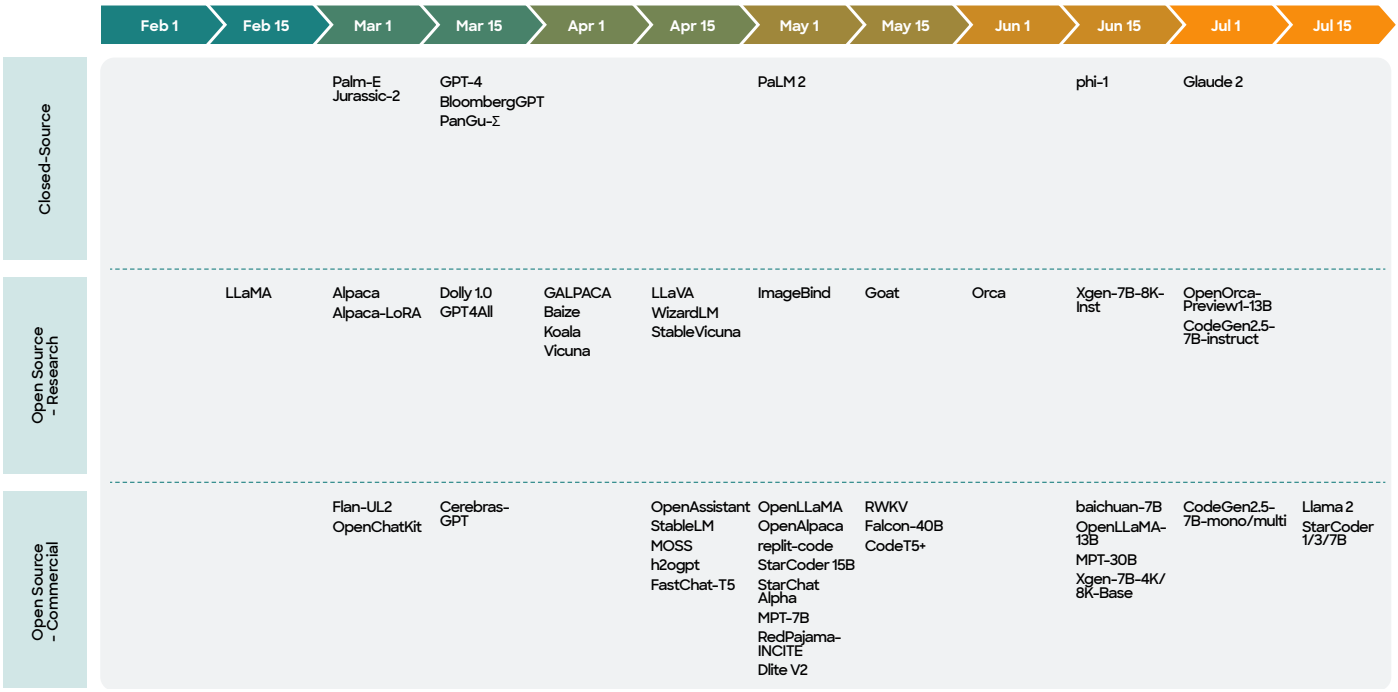


Figure 6. Major large language models released between February and July 2023

Table 2. Selected list of open-source non-commercial models.

Country	Developer & Provider	Model	Parameters	Release
US	Meta AI	OPT-175B	12M-175B	May 2022
US	Meta AI	LLaMA	7B-65B	Feb 2023
US	Meta AI	ImageBind	Unknown	May 2023
US, China	Microsoft, Peking U.	WizardLM	7B-65B	Apr 2023
US	Microsoft	Orca	13B	Jun 2023
US	Stanford University	Alpaca	7B	Mar 2023
US	Georgia Tech Research Institute	GALPACA	6.7B, 30B	Apr 2023
US, China	University of California, San Diego, Sun Yat-sen University, Microsoft Research Asia	Baize	7B-30B	Apr 2023
US	UC Berkeley	Koala	13B	Apr 2023
US	University of Wisconsin-Madison, Microsoft Research, Columbia University	LLaVA	13B	Apr 2023
US	Databricks	Dolly 1.0	6B	Mar 2023
US	Nomic AI	GPT4All	7B-13B	Mar 2023
US	LMSYS Org	Vicuna	13B	Apr 2023
US	CarperAI	StableVicuna	13B	Apr 2023
Singapore	National University of Singapore	Goat	7B	May 2023
France	BigScience	Bloom	176B	Nov 2022
Various	OpenOrca	OpenOrca-Preview1-13B	13B	Jul 2023

The open-source landscape has expanded considerably since then, with many models and datasets emerging that allow potential commercial usage[1]. Notable among these models are Cerebras-GPT, Pythia, Dolly 2.0, GPT4All-J, OpenAssistant, StableLM, H2OGPT, OpenLLaMA, OpenAlpaca, MPT-7B, and RedPajama-INCITE. Together with open-source models, open-source datasets like RedPajama-Data-1T and StarCoderData have been generated and their data curation methods published, further widening the possibilities for commercial applications. These models and datasets aim to provide accessible and customizable alternatives to proprietary commercial options (Tables 3-5).

Overall, these open-source models have helped create fertile ground for experimentation, enabling researchers, developers, and practitioners to explore novel applications and collaborate on advancing the field of generative AI without costly licensing agreements. The availability of these models has democratized access to sophisticated generative language modeling techniques, promoting a more inclusive and vibrant AI development ecosystem. Developers, researchers, and practitioners now have the opportunity to leverage and contribute to these open-source models, driving transformative breakthroughs in diverse applications.

Table 3. Selected list of open-source large language models that allow potential commercial usage.

Country	Developer & Provider	Model	Parameters	Release
US	EleutherAI	GPT-J-6B	6B	Jun 2021
US	EleutherAI	GPT-NeoX-20B	20B	Apr 2022
US	Google	UL2	20B	May 2022
US	Google	Flan T5	80M-11B	Oct 2022
US	Google	Flan UL2	20B	Mar 2023
US	Cerebras	Cerebras-GPT	111M-13B	Mar 2023
US	Nomic AI	GPT4All-J	6B	Apr 2023
US	EleutherAI	Pythia	70M-12B	Apr 2023
US	Databricks	Dolly 2.0	3B-12B	Apr 2023
US	H2O.ai	h2oGPT	12B	Apr 2023
US	LMSYS Org	FastChat-T5	3B	Apr 2023
US	AI Squared	Dlite V2	124M-1.5B	May 2023
US, Spain, UK	RWKV Foundation, EleutherAI, University of Barcelona, Charm Therapeutics, Ohio State University	RWKV	169M-14B	May 2023
US	MosaicML	MPT-7B, 30B	7B, 30B	May-Jun 2023
US	Together	RedPajama-INCITE	3B, 7B	May 2023
US	OpenLM Research, Stability AI	OpenLLaMA	3B, 7B, 13B	May-Jun 2023
US	Meta AI	Llama 2	7B-70B	Jul 2023
UK	Stability AI	StableLM-Alpha	3B-65B	Apr 2023
Germany	LAION AI	Open Assistant (Pythia family)	12B	Apr 2023
UAE	Technology Innovation Institute	Falcon	7B, 40B	May 2023
China	Baichuan	baichuan-7B	7B	Jun 2023

1 The models and datasets mentioned in the tables in this section have been curated based on various sources including providers' official announcements and Github repositories, Hugging Face model cards (https://huggingface.co/models) and open-source knowledge graphs or lists such as the Stanford foundation models ecosystem graph (https://crfm.stanford.edu/) and the Open LLMs Github repository (https://github.com/eugeneyan/open-llms). While these tables serve as a starting point for readers to explore commercially usable models and datasets, it is important to note that licenses for model weights, source codes, or datasets may vary across different branches and downstream products and may be subject to changes across different versions as they evolve. Also, the associated permissive licenses (e.g., CC BY-SA-4.0, Apache 2.0, BSD-3-Clause, MIT, OpenRAIL-M v1) may have different nuances concerning liability, warranty, patent use, copyright, etc. As a best practice, readers should always verify with the original providers the up-to-date licensing conditions of the models as well as those of the associated model weights, source codes, and datasets before engaging in extensive development and launching of commercial usage.

Table 4. Selected list of open-source code-oriented large language models that allow potential commercial usage.

Country	Developer & Provider	Model	Parameters	Release
US	Replit	Replit Code	2.7B	May 2023
US	Salesforce Research	CodeGen2	16B	May 2023
US	Salesforce Research	CodeT5+	16B	May 2023
US	Salesforce Research	Xgen-7B-4K/8K-Base	7B	Jun 2023
US	Salesforce Research	CodeGen2.5-7B-mono/ multi	7B	Jul 2023
France/US	Hugging Face & ServiceNow (BigCode Project)	SantaCoder	1.1B	Jan 2023
France/US	Hugging Face & ServiceNow (BigCode Project)	StarCoder	15B, 1B, 3B, 7B	May, Jul 2023
France/US	Hugging Face & ServiceNow (BigCode Project)	StarChat Alpha	16B	May 2023

Table 5. Selected list of open-source datasets that allow potential commercial usage.

Country	Developer & Provider	Model	Parameters	Release
US	EleutherAI	The Pile	825GB	Dec 2020
US	Anthropic	Helpful and Harmless	79.3MB	Apr 2022
US	Together	RedPajama-Data-1T	5TB	Apr 2023
US	Databricks	databricks-dolly-15k	13.1MB	Apr 2023
France/US	Hugging Face & ServiceNow (BigCode Project)	The Stack	6TB	Nov 2022- Feb 2023
France/US	Hugging Face & ServiceNow (BigCode Project)	StarCoderData	882GB	May 2023
Germany	LAION AI	LAION-5B	11.4TB	Jun 2022
Germany	LAION AI	OIG Dataset	44M	Mar 2023
Germany	LAION AI	OASST1 (OpenAssistant Conversations Dataset)	41.6MB	Apr 2023
UK	University College London	MiniPile	6GB	Apr 2023

Q Could you share your insights on the potential long-term impacts of increasingly advanced open-source LLMs on the European industry?

A *"The emergence of increasingly advanced open-source LLMs can have significant long-term impact on the European industry. It offers opportunities for European companies to leverage and build upon these models to develop innovative AI solutions. In contrast to proprietary (non-European) offerings, this reduces dependence on foreign technology, strengthens intellectual property, and facilitates regulatory compliance."*

- Dr. Stephan Meyer, Head of Artificial Intelligence, Munich Re Group

Best of Both Worlds

This growing open-source ecosystem offers the possibility of a balance between closed-source and open-source models. In a recent paper on the concept of 'FrugalGPT', for example, authors put forward the idea of integrating different types of LLMs to optimize costs and achieve better outcomes (Figure 7; [107]). By embracing both closed-source models and open-source alternatives, organizations can have the best of both worlds, for example, by leveraging GPT-4 as a high-level reasoning and planning engine and then using open-source models to complete specific tasks in contexts where their performance excels.

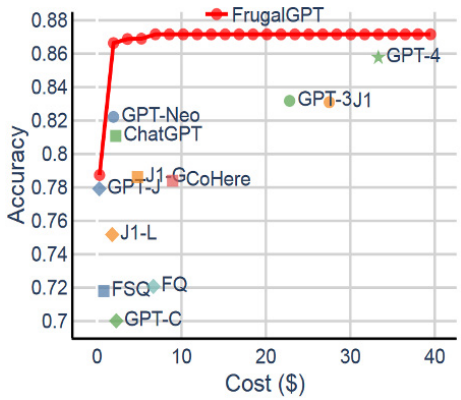


Figure 7. FrugalGPT demonstrates how to use large language models while reducing cost and improving performance [107].

In short, the current LLM landscape includes major closed-source models and an expanding range of open-source alternatives. These models offer powerful capabilities for various NLP tasks including chatbots, knowledge-base question-answering, language generation, and more. Closed-source, commercial models provide robust solutions backed by industry expertise, whereas the recent open-source explosion has led to the emergence of research-only models and commercially usable alternatives, allowing for increased collaboration and innovation. Striking a balance between closed-source and open-source models may help organizations benefit from the respective strengths of these different types of models and drive LLM progress.

3.1.3. Flourishing Large Language Model Applications, Extensions, and Relevant Frameworks

With the recent surge in enthusiasm around generative AI sparked by the launch of ChatGPT, there has been a rapid expansion in the number of possible applications, extensions, and frameworks that center around LLMS. These developments open up myriad new possibilities and pave the way for transformative advances. In this section, we explore some of the major recent developments in this exciting era.

Agentic AI
The term Agentic AI refers to artificial intelligence systems that can make autonomous decisions and take proactive action based on their understanding of a given situation. The emergence of LLMs has accelerated the development of agentic AI as such models can act as a reasoning engine or a core controller for intelligent planning and execution behavior. Applications such as AgentGPT, AgentLLM, Transformers Agent, Langchain Agents, and Auto-GPT demonstrate the capability to act as virtual agents, enabling autonomous decision-making and interaction in dynamic environments to complete target tasks. Agentic LLMs have the potential to revolutionize fields such as customer service, virtual assistants, and autonomous systems.

Coding Assistants and Coding-oriented Models
Another area of significant development is that of coding assistants and relevant models, which aim to enhance software development workflows. GitHub Copilot, StarCoder/StarChat, CodeGen2 [108], and CodeT5+ [109] are prominent examples. These coding assistants and models leverage LLMs to provide intelligent code suggestions and evaluations, generate code snippets or comments, and assist developers in improving and optimizing the quality and efficiency of code. By automating repetitive tasks and offering intelligent guidance, these tools boost developer productivity and facilitate rapid prototyping.

LLM Programming
The development of LLM programming techniques has enabled the creation of novel frameworks to interact with LLMs. LMQL and Low-code LLM [110] are examples of tools that either allow developers to interweave prompts with a control flow (e.g., loops) to increase flexibility and reusability of the prompts, or that incorporate simple low-code visual programming interactions to effectively utilize LLMs for complex tasks. This approach facilitates more controllable and stable LLM responses, making it easier to build applications and automate tasks.

LLM-powered Document Analyzers
LLMs have also been applied to document analysis tasks, leading to the development of assistants such as Arches AI, PDF GPT, and HUMANTA. These tools leverage the power of LLMs to assist with tasks such as document summarization, information extraction, and context-aware analysis. By automating these processes, document analysis assistants can streamline workflows and improve productivity in industries including legal, finance, and research.

LLM-powered Chatbots and Playgrounds
Advances in LLMs have spurred the development of chatbots and playgrounds that facilitate interactive and engaging conversations. OpenAssistant, Vercel.ai playground, Chatbot Arena, h2oGPT, and HuggingChat are notable examples in this domain. These platforms allow users to interact with LLM-powered chatbots, explore creative dialogues, and even develop their own conversational agents.

LLM-powered Domain-Specific Assistants
LLMs have also been customized for specific industries, giving rise to domain-specific assistants. FinChat.io, for instance, caters specifically to the finance industry, providing intelligent support for tasks like financial analysis, investment recommendations, and risk assessment. These assistants leverage domain-specific knowledge and language to deliver tailored solutions to industry-specific challenges.

Model Training, Fine-tuning, and Management Platforms

As the complexity of LLMs increases, there is growing demand for efficient and standardized model training, fine-tuning, and management platforms. It is becoming essential to have robust platforms that facilitate the entire lifecycle of these models. Platforms like H2O LLM Studio, deepspeed Zero++, NVIDIA Nemo Framework, and MosaicML offer unified solutions for model training, fine-tuning, deployment, and monitoring. By providing predefined workflows, tools, and resources, these platforms simplify the process of customizing, fine-tuning, and deploying LLMs, allowing users to leverage the knowledge encoded within the models while being able to tailor models to specific domains or applications and maintain them with minimal difficulty.

Model Compilation and Quantization Frameworks

With larger and more resource-intensive LLMs comes the increasing need for model compilation and quantization frameworks. WebLLM and similar tools provide methods for optimizing and compiling LLMs to reduce their memory footprint and improve efficiency. These frameworks enable deployment of LLMs on resource-constrained devices, running inferences even from a web browser, creating the potential for private, on-device language processing and real-time personal applications.

Other LLM-based Applications

Beyond these categories, a vast array of LLM-based tools and applications is on the horizon. These include marketing tools for sentiment analysis and content generation, knowledge organization platforms for information retrieval and knowledge discovery, text-to-image/video generation models for creative content production, music generation models for composition and harmonization, data analysis frameworks for language-driven insights, voice generation models for natural and expressive speech synthesis, gaming applications for interactive storytelling, and even pharmaceutical applications for molecular compound finding, drug discovery, and protein design.

The diverse range of LLM applications, extensions, and frameworks highlights the versatility and potential of LLMs to address complex challenges across industries. By harnessing the power of language understanding and generation, organizations can unlock new opportunities for automation, innovation, and user experience. As the LLM landscape continues to evolve, it is crucial for businesses to stay informed about the latest advances and consider how these tools could be leveraged to drive their own digital transformation and competitive advantage.

Q In your view, which area of industrial LLM applications shows the most promise for the near future?

A *"In the future, I envision employees seamlessly collaborating with specialized AI assistants to efficiently address daily internal tasks or inquiries by customers. These AI assistants will adeptly access, extract, and integrate relevant knowledge, offering recommended solutions and providing detailed, step-by-step guidance to execute processes effectively."*

- Bernhard Pflugfelder, Head of Use Cases and Applications, appliedAI Initiative GmbH

3.2. Domain-Specific Application of Large Language Models in Industrial Scenarios

3.2.1. Fine-tuning and Adaptation from a Technical Perspective: To What Extent Are They Needed and How Could They Help?

While LLMs are trained on vast amounts of general text data, they can sometimes lack the necessary knowledge for specialized applications. In such cases, fine-tuning models on a smaller, specific dataset can significantly improve performance in that area. Such an approach allows organizations to adapt pre-trained models to their specific needs, and subsequently improve accuracy, relevance, and efficiency [111]. In this section, we will explore some fundamental concepts of fine-tuning and adaptation, before discussing key trends in the development of these techniques that could expand the potential of LLMs even further.

Fundamentals

The terms fine-tuning and adaptation are often used interchangeably to describe closely related techniques used to further train a pre-trained LLM using domain-specific data. The objective is to enable the model to learn specific patterns and nuances unique to the target domain while preserving the core knowledge acquired during pre-training. Fine-tuning is particularly effective when dealing with narrow domains that have limited annotated data available for training. By exposing the model to domain-specific data, it can adapt to the vocabulary, style, and distinctive characteristics of the domain. These approaches result in improved performance and better alignment with specific task requirements, enhancing the applicability of LLMs in domain-specific, specialized industrial scenarios.

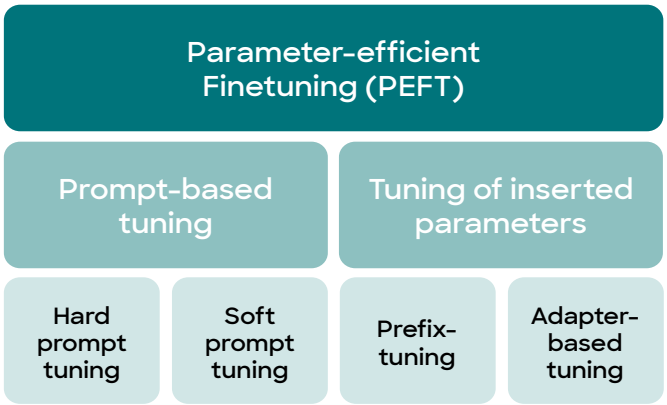


Figure 8. Types of Parameter-efficient Fine-tuning

Full-scale fine-tuning of LLMs poses computational challenges due to the extensive number of parameters involved and requires sufficiently large dedicated hardware resources. Various parameter-efficient fine-tuning (PEFT) approaches have been developed to address this. Such approaches employ techniques such as modification of model input and insertion of trainable parameters into different parts of the model architecture, including hard/soft prompt tuning [112], prefix-tuning [113], and adapter-based tuning (e.g., neural adapters [114], LoRA [115], LLaMA-adapter [116], [117]).

Hard prompt tuning methods modify discrete model input tokens to guide the model's output. Soft prompt tuning optimizes continuous feature vectors derived from the discrete token input layer using gradient-based methods. Techniques involving inserted parameters (prefix-tuning and adapter-based tuning) typically encapsulate them within simple modules that facilitate the language model's adaptation to target domains or tasks. These added modules possess desirable characteristics such as simplicity with a small parameter count, extensibility to the original language models, and flexibility for sequential training on specific domains. By integrating these additional parameters into different parts of the existing LLM architecture, task-specific learning can be achieved, allowing models to be customized for specific tasks or domains. These parameter-efficient fine-tuning approaches aim to strike a balance between model performance and computational resources, such that LLMs can be tailored to meet specific requirements without the need for extensive computational infrastructure (Figure 8).

Making a Decision to Fine-tune
The decision to pursue fine-tuning of LLMs or use out-of-the-box models depends on factors such as data privacy, information security, budgets, make-or-buy and vendor strategy, and requirements for model diversity. Empirical evaluation results also play a role. Fine-tuning may be preferable when dealing with domain-specific tasks that require a high level of customization and performance optimization or when the out-of-the-box model's performance is unsatisfactory, presumably due to insufficient exposure to domain-relevant data during pre-training. Out-of-the-box LLMs are more suitable for general-purpose applications or when the task aligns well with the pre-existing knowledge encoded in the models. These models can offer convenient and efficient solutions without the need for extensive fine-tuning.

Benefits of Fine-tuning and Adaptation from a Technical Perspective
The benefits of fine-tuning and adaptation are evident when it comes to addressing the specific challenges and requirements of domain-specific scenarios. By applying these approaches, organizations can achieve the following:

1) **Improved Performance.** Fine-tuning an LLM with domain-specific data can significantly improve its performance and accuracy on specific tasks. The model becomes more adept at understanding the intricacies of the target domain, leading to more reliable and precise results.

2) **Enhanced Relevance.** Adaptation allows LLMs to understand and generate content in different languages or professional jargon. This is particularly valuable in industrial settings where models need to process and generate text using appropriate terms to cater to a specialized user base. Adapting the model to company-internal language can help ensure that the generated output is technically appropriate for that context.

3) **Personalization and Tailored Outputs:** Customization enables companies to create models that align closely with their specific business needs. By incorporating domain-specific data or organization-specific criteria, models can generate outputs that are highly relevant, personalized, and aligned with the organization's objectives. This level of customization enhances user experience and enables more effective communication with customers or users.

Challenges of Fine-tuning from a Technical Perspective
Despite these benefits, there are limitations and challenges associated with these approaches. Fine-tuning and adaptation require carefully curated datasets that accurately represent the target domain, language, or business context. Obtaining high-quality and representative data can be a challenge, especially in niche or specialized domains where labeled data may be scarce. Additionally, fine-tuning, adaptation, and customization require expertise in machine learning and NLP techniques, as well as sufficient computational resources to train and deploy models effectively.

Fine-tuning, adaptation, and customization offer advantages when it comes to leveraging the potential of LLMs in domain-specific scenarios. These approaches enable organizations to tailor models to their specific needs, resulting in improved performance, relevance, and personalization. A caveat is that careful consideration must be given to the availability of high-quality data, expertise, and computational resources required for implementation. By understanding and utilizing these techniques, organizations can unlock the full potential of LLMs and drive innovation in their respective industries.

As with modeling techniques, there are notable trends emerging in fine-tuning, adaptation, and customization methods. These trends go beyond foundational concepts and provide fresh perspectives on the ever-evolving landscape of LLMs. They present exciting opportunities for further exploration and innovation, pushing the boundaries of what can be achieved.

Low-cost and Efficient Fine-tuning
Researchers are investigating methods to fine-tune LLMs more efficiently [118][119], with several innovative approaches showing promise. QLoRA, for example, implements LoRA on quantized LLMs and has reached 99.3% of the performance level of ChatGPT while requiring just 24 hours of fine-tuning on a single GPU [120]. Another noteworthy recent technique, memory-efficient zeroth-order optimizer (MeZO), addresses the issue of memory consumption during fine-tuning [121]. By reducing memory requirements to the level of inference, MeZO enables efficient training of 30-billion parameter models using a single A100 80GB GPU. These improvements in fine-tuning efficiency not only accelerate the adaptation of LLMs to specific domains or tasks but also resolve problems with resource constraints and time-intensive processes. Optimization of the fine-tuning process can further help organizations exploit the full potential of LLMs while reducing the time and computational resources required.

Finetuning-free Approaches
Another recent trend involves methods that achieve comparable performance without fine-tuning. Researchers have, for example, introduced a mechanism called "distilling step-by-step" that trains smaller models using LLM rationales as additional supervision within a multi-task training framework [46]. These smaller models achieve better performance with fewer labeled/unlabeled training examples and substantially smaller model sizes, while still outperforming LLMs on benchmark tasks. Such trends highlight ongoing efforts to push the boundaries of LLM capabilities and overcome challenges in practical applications.

3.2.2. Towards Domain-specific Dynamic Benchmarking Approaches

Benchmarking is a crucial process for evaluating LLM performance [122]. It involves measuring and comparing various metrics to assess models' capabilities and limitations. With continued advances in the field, there is an increasing need for effective benchmarking. This section focuses on techniques used to benchmark LLMs and emphasizes the significance of dynamic benchmarking.

Traditional Approaches
Benchmarking of LLMs encompasses a range of methods and metrics. One common approach is to evaluate models on standard NLP tasks, such as text classification, sentiment analysis, machine translation, and question-answering. These tasks serve as benchmarks to gauge model performance and provide a basis for comparison across different models. Additional metrics such as accuracy, precision, recall, and F1 score have also been widely used to quantify model performance.

Another approach involves using datasets specifically designed to evaluate performance. These datasets may include diverse linguistic phenomena such as syntactic structures, semantic relationships, and pragmatic understanding. By evaluating performance on these datasets, researchers can better understand a model's capacity to handle complex linguistic tasks.

The Need for Dynamic Benchmarking
Though traditional benchmarking approaches remain useful, they are not without limitations. LLMs are renowned for their capacity to adapt and enhance performance over time through continual learning. Similarly, the content and style

of model input, such as the inclusion of latest news reports or research findings, is likely to evolve as time progresses. These considerations highlight the inadequacy of static benchmarks in assessing a model's potential. This is where dynamic benchmarking comes into play[123][124]. Dynamic benchmarking involves continuously evaluating and updating benchmarks as the model evolves. By periodically assessing the model's performance on new tasks and datasets, researchers can track progress and identify areas that require improvement.

Advantages of Dynamic Benchmarking

Dynamic benchmarking offers several advantages over static benchmarking. First, it enables researchers to evaluate the performance of LLMs in real-world scenarios that evolve over time, taking into account factors such as 'domain shift' or 'concept drift' where data changes. As language evolves, new linguistic phenomena emerge and models must be able to adapt to these changes. Dynamic benchmarking allows researchers to assess a model's ability to handle novel and evolving linguistic changes.

Second, dynamic benchmarking promotes innovation and drives further research and development in the field. By continuously evaluating model performance, researchers can identify weaknesses and focus on addressing limitations. This iterative process encourages development of more advanced techniques and architectures to improve LLM performance.

Third, dynamic benchmarking provides a more comprehensive and up-to-date understanding of a model's strengths and weaknesses. As benchmarks evolve, researchers learn about a model's performance across domains, languages, and tasks. This information is invaluable for strategic managers and NLP practitioners who need to assess the suitability of LLMs for specific applications.

Implementation

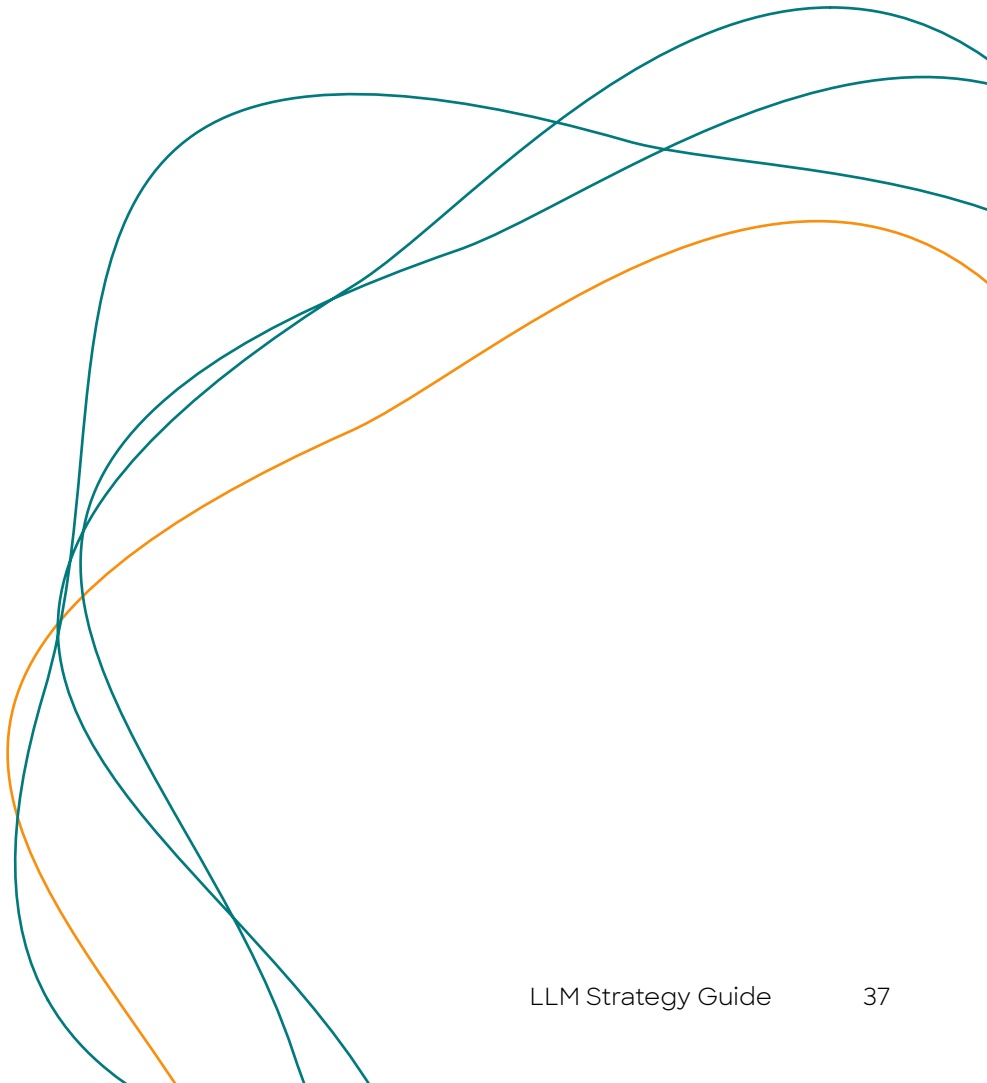
To implement dynamic benchmarking, researchers require access to diverse and evolving datasets that reflect real-world language use. These should include domain-specific data, multilingual data, and data that captures the nuances and complexities of natural language. Collaborations with industry partners, academia, and the wider NLP community can help gather and curate datasets to support dynamic benchmarking.

Q Looking into the crystal ball - to what extent will LLMs be integrated into everyday human and corporate activities in 2030?

A *"They will be everywhere. But we talk about 2025 and not 2030."*

- Dr. Andreas Liebl, Managing Director and Founder, appliedAI Initiative GmbH

Benchmarking plays a crucial role in evaluating LLM performance. While traditional static benchmarks provide valuable insights, dynamic benchmarking offers a more comprehensive and timely understanding of a model's capabilities. It allows researchers to track a model's progress, identify areas for improvement, and ensure its suitability for evolving real-world language challenges. Dynamic benchmarking promotes innovation, drives research, and enables strategic managers and NLP practitioners to make informed decisions about deploying LLMs in their respective domains.



References

[1] W. X. Zhao et al., “A Survey of Large Language Models.” arXiv, Jun. 29, 2023. Accessed: Jul. 13, 2023. [Online]. Available: http://arxiv.org/abs/2303.18223

[2] A. Vaswani et al., “Attention Is All You Need.” arXiv, Dec. 05, 2017. Accessed: Feb. 15, 2023. [Online]. Available: http://arxiv.org/abs/1706.03762

[3] L. Ouyang et al., “Training language models to follow instructions with human feedback.” arXiv, Mar. 04, 2022. Accessed: Feb. 13, 2023. [Online]. Available: http://arxiv.org/abs/2203.02155

[4] P. Christiano, J. Leike, T. B. Brown, M. Martic, S. Legg, and D. Amodei, “Deep reinforcement learning from human preferences.” arXiv, Feb. 17, 2023. Accessed: Jul. 10, 2023. [Online]. Available: http://arxiv.org/abs/1706.03741

[5] R. Taylor et al., “Galactica: A Large Language Model for Science.” arXiv, Nov. 16, 2022. doi: 10.48550/arXiv.2211.09085.

[6] D. Dasgupta, D. Venugopal, and K. D. Gupta, “A Review of Generative AI from Historical Perspectives.” TechRxiv, Feb. 17, 2023. doi: 10.36227/techrxiv.22097942.v1.

[7] R. Huang et al., “AudioGPT: Understanding and Generating Speech, Music, Sound, and Talking Head.” arXiv, Apr. 25, 2023. Accessed: Apr. 26, 2023. [Online]. Available: http://arxiv.org/abs/2304.12995

[8] X. Yang, W. Cheng, L. Petzold, W. Y. Wang, and H. Chen, “DNA-GPT: Divergent N-Gram Analysis for Training-Free Detection of GPT-Generated Text.” arXiv, May 26, 2023. Accessed: May 30, 2023. [Online]. Available: http://arxiv.org/abs/2305.17359

[9] Q. Jin, Y. Yang, Q. Chen, and Z. Lu, “GeneGPT: Augmenting Large Language Models with Domain Tools for Improved Access to Biomedical Information,” ArXiv, p. arXiv:2304.09667v2, Apr. 2023.

[10] A. Yüksel, E. Ulusoy, A. Ünlü, G. Deniz, and T. Doğan, “SELFormer: Molecular Representation Learning via SELFIES Language Models.” arXiv, Apr. 10, 2023. Accessed: Apr. 25, 2023. [Online]. Available: http://arxiv.org/abs/2304.04662

[11] T. S. Frisby and C. J. Langmead, “Identifying Promising Sequences For Protein Engineering Using A Deep Transformer Protein Language Model.” bioRxiv, p. 2023.02.15.528697, Feb. 16, 2023. doi: 10.1101/2023.02.15.528697.

[12] S. Islam et al., “A Comprehensive Survey on Applications of Transformers for Deep Learning Tasks.” arXiv, Jun. 11, 2023. Accessed: Jun. 21, 2023. [Online]. Available: http://arxiv.org/abs/2306.07303

[13] L. Bariah, Q. Zhao, H. Zou, Y. Tian, F. Bader, and M. Debbah, “Large Language Models for Telecom: The Next Big Thing?” arXiv, Jun. 16, 2023. Accessed: Jun. 26, 2023. [Online]. Available: http://arxiv.org/abs/2306.10249

[14] S. Yin et al., “A Survey on Multimodal Large Language Models.” arXiv, Jun. 23, 2023. Accessed: Jun. 27, 2023. [Online]. Available: http://arxiv.org/abs/2306.13549

[15] G. Chen et al., “VideoLLM: Modeling Video Sequence with Large Language Models.” arXiv, May 23, 2023. Accessed: May 25, 2023. [Online]. Available: http://arxiv.org/abs/2305.13292

[16] M. Maaz, H. Rasheed, S. Khan, and F. S. Khan, “Video-ChatGPT: Towards Detailed Video Understanding via Large Vision and Language Models.” arXiv, Jun. 08, 2023. Accessed: Jun. 14, 2023. [Online]. Available: http://arxiv.org/abs/2306.05424

[17] T. Muhammad et al., “Transformer-Based Deep Learning Model for Stock Price Prediction: A Case Study on Bangladesh Stock Market,” Int. J. Comp. Intel. Appl., p. 2350013, Apr. 2023, doi: 10.1142/S146902682350013X.

[18] Z. Bi et al., “Relphormer: Relational Graph Transformer for Knowledge Graph Representations.” arXiv, Mar. 14, 2023. Accessed: Jul. 06, 2023. [Online]. Available: http://arxiv.org/abs/2205.10852

[19] Y. Gai, L. Zhou, K. Qin, D. Song, and A. Gervais, “Blockchain Large Language Models.” arXiv, Apr. 29, 2023. Accessed: May 02, 2023. [Online]. Available: http://arxiv.org/abs/2304.12749

[20] M. Horton, S. Mehta, A. Farhadi, and M. Rastegari, “Bytes Are All You Need: Transformers Operating Directly On File Bytes.” arXiv, May 31, 2023. Accessed: Jun. 05, 2023. [Online]. Available: http://arxiv.org/abs/2306.00238

[21] R. Bommasani et al., “On the Opportunities and Risks of Foundation Models.” arXiv, Jul. 12, 2022. Accessed: Feb. 28, 2023. [Online]. Available: http://arxiv.org/abs/2108.07258

[22] C. Zhou et al., “A Comprehensive Survey on Pretrained Foundation Models: A History from BERT to ChatGPT.” arXiv, Feb. 18, 2023. Accessed: Feb. 23, 2023. [Online]. Available: http://arxiv.org/abs/2302.09419

[23] H. Jiang, “A Latent Space Theory for Emergent Abilities in Large Language Models.” arXiv, Apr. 24, 2023. Accessed: Apr. 25, 2023. [Online]. Available: http://arxiv.org/abs/2304.09960

[24] R. Schaeffer, B. Miranda, and S. Koyejo, “Are Emergent Abilities of Large Language Models a Mirage?” arXiv, Apr. 28, 2023. Accessed: May 02, 2023. [Online]. Available: http://arxiv.org/abs/2304.15004

[25] K. Ahuja and D. Lopez-Paz, “A Closer Look at In-Context Learning under Distribution Shifts.” arXiv, May 26, 2023. Accessed: May 30, 2023. [Online]. Available: http://arxiv.org/abs/2305.16704

[26] X. Han, D. Simig, T. Mihaylov, Y. Tsvetkov, A. Celikyilmaz, and T. Wang, “Understanding In-Context Learning via Supportive Pretraining Data.” arXiv, Jun. 26, 2023. Accessed: Jun. 28, 2023. [Online]. Available: http://arxiv.org/abs/2306.15091

[27] Y. Hou et al., “Large Language Models are Zero-Shot Rankers for Recommender Systems.” arXiv, May 15, 2023. Accessed: May 19, 2023. [Online]. Available: http://arxiv.org/abs/2305.08845

[28] T. Kojima, S. S. Gu, M. Reid, Y. Matsuo, and Y. Iwasawa, “Large Language Models are Zero-Shot Reasoners.” arXiv, Jan. 29, 2023. Accessed: Feb. 13, 2023. [Online]. Available: http://arxiv.org/abs/2205.11916

[29] T. Shen, G. Long, X. Geng, C. Tao, T. Zhou, and D. Jiang, “Large Language Models are Strong Zero-Shot Retriever.” arXiv, Apr. 27, 2023. Accessed: May 02, 2023. [Online]. Available: http://arxiv.org/abs/2304.14233

[30] Y. Li, Y. Wu, J. Li, and S. Liu, “Prompting Large Language Models for Zero-Shot Domain Adaptation in Speech Recognition.” arXiv, Jun. 28, 2023. Accessed: Jul. 03, 2023. [Online]. Available: http://arxiv.org/abs/2306.16007

[31] S. Albanie, L. Momeni, and J. F. Henriques, “Large Language Models are Few-shot Publication Scoopers.” arXiv, Apr. 02, 2023. Accessed: Apr. 06, 2023. [Online]. Available: http://arxiv.org/abs/2304.00521

[32] Z. Chen, M. M. Balan, and K. Brown, “Language Models are Few-shot Learners for Prognostic Prediction.” arXiv, Feb. 26, 2023. Accessed: Mar. 02, 2023. [Online]. Available: http://arxiv.org/abs/2302.12692

[33] T. B. Brown et al., “Language Models are Few-Shot Learners.” arXiv, Jul. 22, 2020. doi: 10.48550/arXiv.2005.14165.

[34] S. Diao, P. Wang, Y. Lin, and T. Zhang, “Active Prompting with Chain-of-Thought for Large Language Models.” arXiv, Feb. 26, 2023. Accessed: Feb. 28, 2023. [Online]. Available: http://arxiv.org/abs/2302.12246

[35] Y. Fu, L. Ou, M. Chen, Y. Wan, H. Peng, and T. Khot, “Chain-of-Thought Hub: A Continuous Effort to Measure Large Language Models’ Reasoning Performance.” arXiv, May 26, 2023. Accessed: May 30, 2023. [Online]. Available: http://arxiv.org/abs/2305.17306

[36] Z. Zhang, A. Zhang, M. Li, H. Zhao, G. Karypis, and A. Smola, “Multimodal Chain-of-Thought Reasoning in Language Models.” arXiv, Feb. 08, 2023. Accessed: Feb. 16, 2023. [Online]. Available: http://arxiv.org/abs/2302.00923

[37] B. Peng et al., “RWKV: Reinventing RNNs for the Transformer Era.” arXiv, May 22, 2023. Accessed: May 24, 2023. [Online]. Available: http://arxiv.org/abs/2305.13048

[38] J. Ding et al., “LongNet: Scaling Transformers to 1,000,000,000 Tokens.” arXiv, Jul. 05, 2023. Accessed: Jul. 07, 2023. [Online]. Available: http://arxiv.org/abs/2307.02486

[39] A. Bertsch, U. Alon, G. Neubig, and M. R. Gormley, “Unlimiformer: Long-Range Transformers with Unlimited Length Input.” arXiv, May 02, 2023. Accessed: May 03, 2023. [Online]. Available: http://arxiv.org/abs/2305.01625

[40] D. Uthus, S. Ontañón, J. Ainslie, and M. Guo, “mLongT5: A Multilingual and Efficient Text-To-Text Transformer for Longer Sequences.” arXiv, May 18, 2023. Accessed: May 22, 2023. [Online]. Available: http://arxiv.org/abs/2305.11129

[41] C. Xu et al., “WizardLM: Empowering Large Language Models to Follow Complex Instructions.” arXiv, Apr. 24, 2023. Accessed: Apr. 28, 2023. [Online]. Available: http://arxiv.org/abs/2304.12244

[42] J. Kaddour, “The MiniPile Challenge for Data-Efficient Language Models.” arXiv, Apr. 17, 2023.

Accessed: Apr. 20, 2023. [Online]. Available: http://arxiv.org/abs/2304.08442

[43] H. S. V. N. S. K. Renduchintala et al., “INGENIOUS: Using Informative Data Subsets for Efficient Pre-Training of Large Language Models.” arXiv, May 11, 2023. Accessed: May 16, 2023. [Online]. Available: http://arxiv.org/abs/2305.06677

[44] S. M. Xie et al., “DoReMi: Optimizing Data Mixtures Speeds Up Language Model Pretraining.” arXiv, May 17, 2023. Accessed: May 19, 2023. [Online]. Available: http://arxiv.org/abs/2305.10429

[45] I. R. McKenzie et al., “Inverse Scaling: When Bigger Isn’t Better.” arXiv, Jun. 15, 2023. Accessed: Jun. 21, 2023. [Online]. Available: http://arxiv.org/abs/2306.09479

[46] C.-Y. Hsieh et al., “Distilling Step-by-Step! Outperforming Larger Language Models with Less Training Data and Smaller Model Sizes.” arXiv, May 03, 2023. doi: 10.48550/arXiv.2305.02301.

[47] R. Eldan and Y. Li, “TinyStories: How Small Can Language Models Be and Still Speak Coherent English?” arXiv, May 12, 2023. Accessed: May 16, 2023. [Online]. Available: http://arxiv.org/abs/2305.07759

[48] C. Xu, Y. Xu, S. Wang, Y. Liu, C. Zhu, and J. McAuley, “Small Models are Valuable Plug-ins for Large Language Models.” arXiv, May 15, 2023. Accessed: May 16, 2023. [Online]. Available: http://arxiv.org/abs/2305.08848

[49] F. Brahman et al., “PlaSma: Making Small Language Models Better Procedural Knowledge Models for (Counterfactual) Planning.” arXiv, May 30, 2023. Accessed: Jun. 01, 2023. [Online]. Available: http://arxiv.org/abs/2305.19472

[50] Z. Guo, P. Wang, Y. Wang, and S. Yu, “Dr. LLaMA: Improving Small Language Models in Domain-Specific QA via Generative Data Augmentation.” arXiv, May 12, 2023. Accessed: May 16, 2023. [Online]. Available: http://arxiv.org/abs/2305.07804

[51] Y. Zhao, R. Joshi, T. Liu, M. Khalman, M. Saleh, and P. J. Liu, “SLiC-HF: Sequence Likelihood Calibration with Human Feedback.” arXiv, May 17, 2023. Accessed: May 19, 2023. [Online]. Available: http://arxiv.org/abs/2305.10425

[52] Z. Yuan, H. Yuan, C. Tan, W. Wang, S. Huang, and F. Huang, “RRHF: Rank Responses to Align Language Models with Human Feedback without tears.” arXiv, Apr. 11, 2023. Accessed: Apr. 25, 2023. [Online]. Available: http://arxiv.org/abs/2304.05302

[53] C. Zhou et al., “LIMA: Less Is More for Alignment.” arXiv, May 18, 2023. Accessed: May 22, 2023. [Online]. Available: http://arxiv.org/abs/2305.11206

[54] A. Bietti, V. Cabannes, D. Bouchacourt, H. Jegou, and L. Bottou, “Birth of a Transformer: A Memory Viewpoint.” arXiv, Jun. 01, 2023. Accessed: Jun. 05, 2023. [Online]. Available: http://arxiv.org/abs/2306.00802

[55] W. Wang et al., “Augmenting Language Models with Long-Term Memory.” arXiv, Jun. 12, 2023. Accessed: Jun. 14, 2023. [Online]. Available: http://arxiv.org/abs/2306.07174

[56] X. Liang et al., “Unleashing Infinite-Length Input Capacity for Large-scale Language Models with Self-Controlled Memory System.” arXiv, Apr. 26, 2023. Accessed: Apr. 28, 2023. [Online]. Available: http://arxiv.org/abs/2304.13343

[57] J. Kang, R. Laroche, X. Yuan, A. Trischler, X. Liu, and J. Fu, “Think Before You Act: Decision Transformers with Internal Working Memory.” arXiv, May 23, 2023. Accessed: May 30, 2023. [Online]. Available: http://arxiv.org/abs/2305.16338

[58] W. Zhong, L. Guo, Q. Gao, and Y. Wang, “MemoryBank: Enhancing Large Language Models with Long-Term Memory.” arXiv, May 18, 2023. Accessed: May 22, 2023. [Online]. Available: http://arxiv.org/abs/2305.10250

[59] Y. Zeng et al., “What Matters in Training a GPT4-Style Language Model with Multimodal Inputs?” arXiv, Jul. 05, 2023. Accessed: Jul. 07, 2023. [Online]. Available: http://arxiv.org/abs/2307.02469

[60] W. Berrios, G. Mittal, T. Thrush, D. Kiela, and A. Singh, “Towards Language Models That Can See: Computer Vision Through the LENS of Natural Language.” arXiv, Jun. 28, 2023. Accessed: Jun. 29, 2023. [Online]. Available: http://arxiv.org/abs/2306.16410

[61] C. Lyu et al., “Macaw-LLM: Multi-Modal Language Modeling with Image, Audio, Video, and Text Integration.” arXiv, Jun. 15, 2023. Accessed: Jun. 21, 2023. [Online]. Available: http://arxiv.org/abs/2306.09093

[62] R. Girdhar et al., “ImageBind: One Embedding Space To Bind Them All.” arXiv, May 31, 2023. doi:

10.48550/arXiv.2305.05665.

[63] L. Xue et al., “ULIP-2: Towards Scalable Multimodal Pre-training For 3D Understanding.” arXiv, May 14, 2023. Accessed: May 16, 2023. [Online]. Available: http://arxiv.org/abs/2305.08275

[64] C. Yeh, Y. Chen, A. Wu, C. Chen, F. Viégas, and M. Wattenberg, “AttentionViz: A Global View of Transformer Attention.” arXiv, May 04, 2023. Accessed: May 08, 2023. [Online]. Available: http://arxiv.org/abs/2305.03210

[65] J. Copet et al., “Simple and Controllable Music Generation.” arXiv, Jun. 08, 2023. Accessed: Jun. 14, 2023. [Online]. Available: http://arxiv.org/abs/2306.05284

[66] M. S. Ozdayi et al., “Controlling the Extraction of Memorized Data from Large Language Models via Prompt-Tuning.” arXiv, May 19, 2023. Accessed: May 22, 2023. [Online]. Available: http://arxiv.org/abs/2305.11759

[67] H. Zhang, M. Dang, N. Peng, and G. V. den Broeck, “Tractable Control for Autoregressive Language Generation.” arXiv, Apr. 17, 2023. Accessed: Apr. 20, 2023. [Online]. Available: http://arxiv.org/abs/2304.07438

[68] T. Zhang, Y. Zhang, V. Vineet, N. Joshi, and X. Wang, “Controllable Text-to-Image Generation with GPT-4.” arXiv, May 29, 2023. Accessed: May 31, 2023. [Online]. Available: http://arxiv.org/abs/2305.18583

[69] J. Hewitt, J. Thickstun, C. D. Manning, and P. Liang, “Backpack Language Models.” arXiv, May 26, 2023. Accessed: May 30, 2023. [Online]. Available: http://arxiv.org/abs/2305.16765

[70] H. Zhang, M. Dang, N. Peng, and G. V. den Broeck, “Tractable Control for Autoregressive Language Generation.” arXiv, Apr. 17, 2023. Accessed: Apr. 20, 2023. [Online]. Available: http://arxiv.org/abs/2304.07438

[71] S. Chen, S. Gao, and J. He, “Evaluating Factual Consistency of Summaries with Large Language Models.” arXiv, May 23, 2023. Accessed: May 27, 2023. [Online]. Available: http://arxiv.org/abs/2305.14069

[72] P. Laban et al., “LLMs as Factual Reasoners: Insights from Existing Benchmarks and Beyond.” arXiv, May 23, 2023. Accessed: May 25, 2023. [Online]. Available: http://arxiv.org/abs/2305.14540

[73] S. Zheng, J. Huang, and K. C.-C. Chang, “Why Does ChatGPT Fall Short in Answering Questions Faithfully?” arXiv, Apr. 20, 2023. Accessed: Apr. 25, 2023. [Online]. Available: http://arxiv.org/abs/2304.10513

[74] A. Borji, “A Categorical Archive of ChatGPT Failures.” arXiv, Feb. 18, 2023. doi: 10.48550/arXiv.2302.03494.

[75] Z. Gekhman, J. Herzig, R. Aharoni, C. Elkind, and I. Szpektor, “TrueTeacher: Learning Factual Consistency Evaluation with Large Language Models.” arXiv, May 18, 2023. Accessed: May 22, 2023. [Online]. Available: http://arxiv.org/abs/2305.11171

[76] J. Kirchenbauer et al., “On the Reliability of Watermarks for Large Language Models.” arXiv, Jun. 09, 2023. Accessed: Jun. 13, 2023. [Online]. Available: http://arxiv.org/abs/2306.04634

[77] E. Mavroudi, T. Afouras, and L. Torresani, “Learning to Ground Instructional Articles in Videos through Narrations.” arXiv, Jun. 06, 2023. Accessed: Jun. 07, 2023. [Online]. Available: http://arxiv.org/abs/2306.03802

[78] Z. Lin, S. Trivedi, and J. Sun, “Generating with Confidence: Uncertainty Quantification for Black-box Large Language Models.” arXiv, May 30, 2023. Accessed: Jun. 05, 2023. [Online]. Available: http://arxiv.org/abs/2305.19187

[79] J. Yu et al., “KoLA: Carefully Benchmarking World Knowledge of Large Language Models.” arXiv, Jun. 15, 2023. Accessed: Jun. 21, 2023. [Online]. Available: http://arxiv.org/abs/2306.09296

[80] R. Cohen, M. Hamri, M. Geva, and A. Globerson, “LM vs LM: Detecting Factual Errors via Cross Examination.” arXiv, May 22, 2023. Accessed: May 24, 2023. [Online]. Available: http://arxiv.org/abs/2305.13281

[81] T. Zhang et al., “Interpretable Unified Language Checking.” arXiv, Apr. 07, 2023. Accessed: Apr. 12, 2023. [Online]. Available: http://arxiv.org/abs/2304.03728

[82] P. Manakul, A. Liusie, and M. J. F. Gales, “SelfCheckGPT: Zero-Resource Black-Box Hallucination Detection for Generative Large Language Models.” arXiv, Mar. 15, 2023. Accessed: Mar. 22, 2023. [Online]. Available: http://arxiv.org/abs/2303.08896

[83] B. Peng et al., “Check Your Facts and Try Again: Improving Large Language Models with External Knowledge and Automated Feedback.” arXiv, Mar. 01, 2023. Accessed: Mar. 02, 2023. [Online]. Available: http://arxiv.org/abs/2302.12813

[84] Z. Luo et al., “Augmented Large Language Models with Parametric Knowledge Guiding.” arXiv, May 08, 2023. Accessed: May 12, 2023. [Online]. Available: http://arxiv.org/abs/2305.04757

[85] Y. Xi et al., “Towards Open-World Recommendation with Knowledge Augmentation from Large Language Models.” arXiv, Jun. 19, 2023. Accessed: Jun. 26, 2023. [Online]. Available: http://arxiv.org/abs/2306.10933

[86] L. Yang, H. Chen, Z. Li, X. Ding, and X. Wu, “ChatGPT is not Enough: Enhancing Large Language Models with Knowledge Graphs for Fact-aware Language Modeling.” arXiv, Jun. 20, 2023. Accessed: Jun. 26, 2023. [Online]. Available: http://arxiv.org/abs/2306.11489

[87] N. Mündler, J. He, S. Jenko, and M. Vechev, “Self-contradictory Hallucinations of Large Language Models: Evaluation, Detection and Mitigation.” arXiv, May 25, 2023. Accessed: May 30, 2023. [Online]. Available: http://arxiv.org/abs/2305.15852

[88] T. Gao, H. Yen, J. Yu, and D. Chen, “Enabling Large Language Models to Generate Text with Citations.” arXiv, May 23, 2023. Accessed: May 27, 2023. [Online]. Available: http://arxiv.org/abs/2305.14627

[89] X. Li et al., “Chain of Knowledge: A Framework for Grounding Large Language Models with Structured Knowledge Bases.” arXiv, May 22, 2023. Accessed: May 25, 2023. [Online]. Available: http://arxiv.org/abs/2305.13269

[90] K. Bhatia, A. Narayan, C. De Sa, and C. Ré, “TART: A plug-and-play Transformer module for task-agnostic reasoning.” arXiv, Jun. 13, 2023. Accessed: Jun. 21, 2023. [Online]. Available: http://arxiv.org/abs/2306.07536

[91] M. Kwon, H. Hu, V. Myers, S. Karamcheti, A. Dragan, and D. Sadigh, “Toward Grounded Social Reasoning.” arXiv, Jun. 14, 2023. Accessed: Jun. 21, 2023. [Online]. Available: http://arxiv.org/abs/2306.08651

[92] A. Piktus, “Online tools help large language models to solve problems through reasoning,” Nature, May 2023, doi: 10.1038/d41586-023-01411-4.

[93] E. Kıcıman, R. Ness, A. Sharma, and C. Tan, “Causal Reasoning and Large Language Models: Opening a New Frontier for Causality.” arXiv, Apr. 28, 2023. Accessed: May 04, 2023. [Online]. Available: http://arxiv.org/abs/2305.00050

[94] G. Poesia, K. Gandhi, E. Zelikman, and N. D. Goodman, “Certified Reasoning with Language Models.” arXiv, Jun. 06, 2023. Accessed: Jun. 14, 2023. [Online]. Available: http://arxiv.org/abs/2306.04031

[95] O. Yoran, T. Wolfson, B. Bogin, U. Katz, D. Deutch, and J. Berant, “Answering Questions by Meta-Reasoning over Multiple Chains of Thought.” arXiv, Apr. 25, 2023. Accessed: Apr. 26, 2023. [Online]. Available: http://arxiv.org/abs/2304.13007

[96] P. Lu et al., “Chameleon: Plug-and-Play Compositional Reasoning with Large Language Models.” arXiv, Apr. 19, 2023. Accessed: Apr. 24, 2023. [Online]. Available: http://arxiv.org/abs/2304.09842

[97] S. Bubeck et al., “Sparks of Artificial General Intelligence: Early experiments with GPT-4.” arXiv, Apr. 13, 2023. Accessed: May 03, 2023. [Online]. Available: http://arxiv.org/abs/2303.12712

[98] S. Yao et al., “Tree of Thoughts: Deliberate Problem Solving with Large Language Models.” arXiv, May 17, 2023. Accessed: May 22, 2023. [Online]. Available: http://arxiv.org/abs/2305.10601

[99] J. Lanchantin, S. Toshniwal, J. Weston, A. Szlam, and S. Sukhbaatar, “Learning to Reason and Memorize with Self-Notes.” arXiv, May 01, 2023. doi: 10.48550/arXiv.2305.00833.

[100] Y. Xie et al., “OlaGPT: Empowering LLMs With Human-like Problem-Solving Abilities.” arXiv, May 23, 2023. Accessed: May 30, 2023. [Online]. Available: http://arxiv.org/abs/2305.16334

[101] H. Zhang et al., “Building Cooperative Embodied Agents Modularly with Large Language Models.” arXiv, Jul. 05, 2023. Accessed: Jul. 07, 2023. [Online]. Available: http://arxiv.org/abs/2307.02485

[102] D. Gao et al., “AssistGPT: A General Multi-modal Assistant that can Plan, Execute, Inspect, and Learn.” arXiv, Jun. 14, 2023. Accessed: Jun. 21, 2023. [Online]. Available: http://arxiv.org/abs/2306.08640

[103] Z. Zhang, X. Zhang, W. Xie, and Y. Lu, “Responsible Task Automation: Empowering Large Language Models as Responsible Task Automators.” arXiv, Jun. 01, 2023. Accessed: Jun. 05, 2023. [Online]. Available: http://arxiv.org/abs/2306.01242

[104] G. Wang et al., “Voyager: An Open-Ended Embodied Agent with Large Language Models.” arXiv, May

25, 2023. doi: 10.48550/arXiv.2305.16291.

[105] T. Cai, X. Wang, T. Ma, X. Chen, and D. Zhou, “Large Language Models as Tool Makers.” arXiv, May 26, 2023. Accessed: May 30, 2023. [Online]. Available: http://arxiv.org/abs/2305.17126

[106] C. Colas, L. Teodorescu, P.-Y. Oudeyer, X. Yuan, and M.-A. Côté, “Augmenting Autotelic Agents with Large Language Models.” arXiv, May 21, 2023. Accessed: May 23, 2023. [Online]. Available: http://arxiv.org/abs/2305.12487

[107] L. Chen, M. Zaharia, and J. Zou, “FrugalGPT: How to Use Large Language Models While Reducing Cost and Improving Performance.” arXiv, May 09, 2023. Accessed: May 10, 2023. [Online]. Available: http://arxiv.org/abs/2305.05176

[108] E. Nijkamp, H. Hayashi, C. Xiong, S. Savarese, and Y. Zhou, “CodeGen2: Lessons for Training LLMs on Programming and Natural Languages.” arXiv, May 03, 2023. Accessed: May 09, 2023. [Online]. Available: http://arxiv.org/abs/2305.02309

[109] Y. Wang, H. Le, A. D. Gotmare, N. D. Q. Bui, J. Li, and S. C. H. Hoi, “CodeT5+: Open Code Large Language Models for Code Understanding and Generation.” arXiv, May 13, 2023. Accessed: May 16, 2023. [Online]. Available: http://arxiv.org/abs/2305.07922

[110] Y. Cai et al., “Low-code LLM: Visual Programming over LLMs.” arXiv, Apr. 17, 2023. doi: 10.48550/arXiv.2304.08103.

[111] J. Yang et al., “Harnessing the Power of LLMs in Practice: A Survey on ChatGPT and Beyond.” arXiv, Apr. 26, 2023. Accessed: Apr. 27, 2023. [Online]. Available: http://arxiv.org/abs/2304.13712

[112] B. Lester, R. Al-Rfou, and N. Constant, “The Power of Scale for Parameter-Efficient Prompt Tuning,” in Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, Online and Punta Cana, Dominican Republic: Association for Computational Linguistics, Nov. 2021, pp. 3045–3059. doi: 10.18653/v1/2021.emnlp-main.243.

[113] X. L. Li and P. Liang, “Prefix-Tuning: Optimizing Continuous Prompts for Generation.” arXiv, Jan. 01, 2021. doi: 10.48550/arXiv.2101.00190.

[114] Z. Hu et al., “LLM-Adapters: An Adapter Family for Parameter-Efficient Fine-Tuning of Large Language Models.” arXiv, Apr. 04, 2023. Accessed: Apr. 11, 2023. [Online]. Available: http://arxiv.org/abs/2304.01933

[115] E. J. Hu et al., “LoRA: Low-Rank Adaptation of Large Language Models.” arXiv, Oct. 16, 2021. doi: 10.48550/arXiv.2106.09685.

[116] P. Gao et al., “LLaMA-Adapter V2: Parameter-Efficient Visual Instruction Model.” arXiv, Apr. 28, 2023. Accessed: May 02, 2023. [Online]. Available: http://arxiv.org/abs/2304.15010

[117] R. Zhang et al., “LLaMA-Adapter: Efficient Fine-tuning of Language Models with Zero-init Attention.” arXiv, Mar. 28, 2023. Accessed: Apr. 03, 2023. [Online]. Available: http://arxiv.org/abs/2303.16199

[118] Y. Chai, J. Gkountouras, G. G. Ko, D. Brooks, and G.-Y. Wei, “INT2.1: Towards Fine-Tunable Quantized Large Language Models with Error Correction through Low-Rank Adaptation.” arXiv, Jun. 13, 2023. doi: 10.48550/arXiv.2306.08162.

[119] L. Chen, J. Chen, T. Goldstein, H. Huang, and T. Zhou, “InstructZero: Efficient Instruction Optimization for Black-Box Large Language Models.” arXiv, Jun. 05, 2023. Accessed: Jun. 06, 2023. [Online]. Available: http://arxiv.org/abs/2306.03082

[120] T. Dettmers, A. Pagnoni, A. Holtzman, and L. Zettlemoyer, “QLoRA: Efficient Finetuning of Quantized LLMs.” arXiv, May 23, 2023. Accessed: May 25, 2023. [Online]. Available: http://arxiv.org/abs/2305.14314

[121] S. Malladi et al., “Fine-Tuning Language Models with Just Forward Passes.” arXiv, May 26, 2023. Accessed: May 30, 2023. [Online]. Available: http://arxiv.org/abs/2305.17333

[122] Y. Chang et al., “A Survey on Evaluation of Large Language Models.” arXiv, Jul. 06, 2023. Accessed: Jul. 07, 2023. [Online]. Available: http://arxiv.org/abs/2307.03109

[123] D. Kiela et al., “Dynabench: Rethinking Benchmarking in NLP.” arXiv, Apr. 07, 2021. Accessed: Apr. 13, 2023. [Online]. Available: http://arxiv.org/abs/2104.14337

[124] Z. Ma et al., “Dynaboard: An Evaluation-As-A-Service Platform for Holistic Next-Generation Benchmarking.” arXiv, May 20, 2021. doi: 10.48550/arXiv.2106.06052.

Authors

Dr. Paul Yu-Chun Chang
AI Expert: Foundation Models - Large Language Models,
appliedAI Initiative GmbH
p.chang@appliedai.de

Paul Yu-Chun Chang works as an AI Expert specializing in Large Language Models at appliedAI Initiative GmbH. He has 10 years of interdisciplinary research experience in computational linguistics, cognitive neuroscience, and AI, and 5 years of industrial experience in developing AI algorithms in language modeling and image analytics. Paul holds a PhD from LMU Munich, where he integrated NLP and machine learning methods to study brain language cognition.

Bernhard Pflugfelder
Head of Use Cases and Applications,
appliedAI Initiative GmbH
b.pflugfelder@appliedai.de

Bernhard Pflugfelder works as Head of Use Cases and Applications at the appliedAI Initiative GmbH. Bernhard has 15 years of experience in the fields of Data Science, Natural Language Processing (NLP), as well as data and AI across different companies such as BMW Group or Volkswagen Group. He is renowned for his expertise especially in the field of AI in general, NLP and Generative AI in particular.

Contributors

Dr. Philipp Hartmann
Director of AI Strategy,
appliedAI Initiative GmbH
p.hartmann@appliedai.de

Philipp Hartmann serves appliedAI as Director of AI Strategy at the appliedAI Initiative GmbH. Prior to joining appliedAI, he spent four years at McKinsey&Company as a strategy consultant. Philipp holds a PhD from Technical University of Munich where he investigated factors of competitive advantage in Artificial Intelligence.

Mingyang Ma
Senior AI Strategist,
appliedAI Initiative GmbH
m.ma@appliedai.de

Mingyang Ma works as Senior AI Strategist at the appliedAI Initiative GmbH, supporting all partner companies' decision making and technical solution identification of various AI use cases, with a particular focus on leveraging LLMs. With over 6 years of expertise in NLP, Mingyang has excelled in the realm of Conversational AI, demonstrating her proficiency in application DevOps and platform development across various processes during her tenure at BMW Group in both Germany and the USA.

Simon-Pierre Genot
Senior Manager AI Strategy,
Infineon Technologies
simon.genot@infineon.com

Simon works as a Senior Manager AI Strategy at Infineon, where he is responsible for AI strategy and use case development. Previously, Simon worked in machine learning research for IBM Research in the USA before transitioning to the strategic side by launching the first AI initiative at BayWa.

Dr. Mark Buckley
Research scientist,
Siemens AG

Mark holds a PhD in computational linguistics from Saarland University, where he worked on machine learning methods for dialogue systems. He joined Siemens Technology as a research scientist for industrial NLP in 2015, working on low-resource NLP, domain adaptation and the interface of structured and unstructured data.

About appliedAI Initiative GmbH

appliedAI is Europe's largest initiative for the application of trusted AI technology. The initiative was established in 2017 by Dr. Andreas Liebl as a division of UnternehmerTUM Munich and transferred to a joint venture with Innovation Park Artificial Intelligence (IPAI) Heilbronn in 2022.

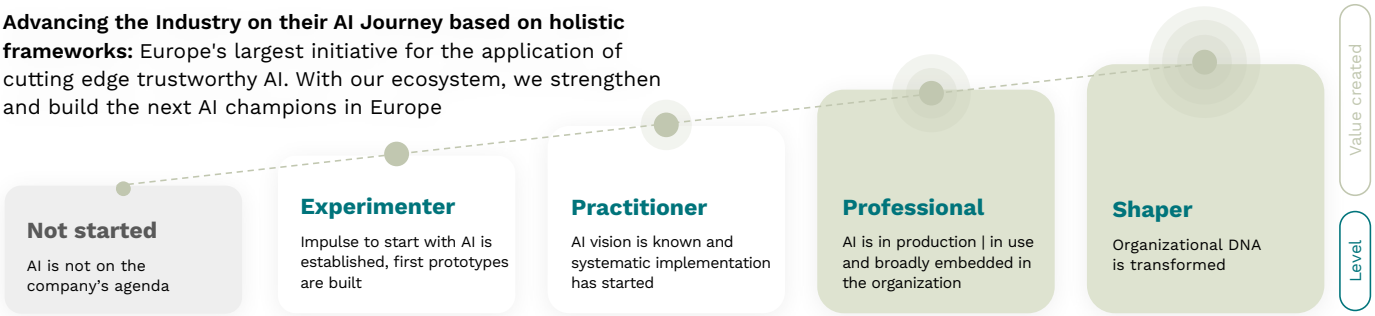
At the Munich and Heilbronn offices, more than 100 employees pursue the goal of making the European industry a shaper in the AI era in order to maintain Europe's competitiveness and actively shape the future.

appliedAI holistically supports international corporations, including BMW and Siemens, as well as medium-sized companies in their AI transformation. This is accomplished through partnership-based exchange and joint knowledge building, comprehensive accelerator programs, and specific solutions and services.

For more information, please visit
https://www.appliedai.de/en/

Advancing Europe's industry to compete in the age of AI, shaping a future that we desire to live in

Advancing the Industry on their AI Journey based on holistic frameworks: Europe's largest initiative for the application of cutting edge trustworthy AI. With our ecosystem, we strengthen and build the next AI champions in Europe



Partnership + Solutions & Services + Programmes + Ventures

100+	300+	200+	50+	25+	2.5K+	45K+
Experts	Partners in ecosystem	Companies supported	AI applications	AI strategies	Trained in live sessions	Trained in online courses

Embedded in **global** AI networks

Represented in **OECD & GPAI**

Acknowledgement

The content presented in Chapter 2 *"Make or To Buy: Leveraging Large Language Models in Business"* is based upon the invaluable insights and research findings derived from the publication titled "AI Insights - Making business decisions in the realm of Large Language Models," authored by Dr. Philip Hutchinson and Bernhard Pflugfelder and published by the appliedAI Institute for Europe gGmbH. We extend our sincere appreciation to the authors for their work that has served as a fundamental reference and inspiration. Their expertise and dedication have played a crucial role in shaping the ideas and understanding presented herein.

The content presented in this white paper has been influenced and inspired by discussions and exchanges within the appliedAI Working Group *"Large Language Models"* including appliedAI industry partners such as BMW Group, Giesecke+Devrient GmbH, EnBW Energie Baden-Württemberg AG, Infineon Technologies AG, Miele & Cie. KG, Munich Re Group, Rohde & Schwarz GmbH & Co. KG, Siemens AG. The collective expertise, exchange and dedication to advancing the knowledge in Generative AI was a great inspiration throughout the process of creating this white paper.

initiative for applied artificial intelligence

A Guide for Large Language Model Make-or-Buy Strategies: Business and Technical Insights

appliedAI Initiative GmbH
Freddie Mercury Street 5
80797 Munich
Germany
www.appliedai.de