LACONIA
CAPITAL
381 Park Ave S,
New York, NY 10016

LARGE LANGUAGE MODELS

Sector Deep Dive

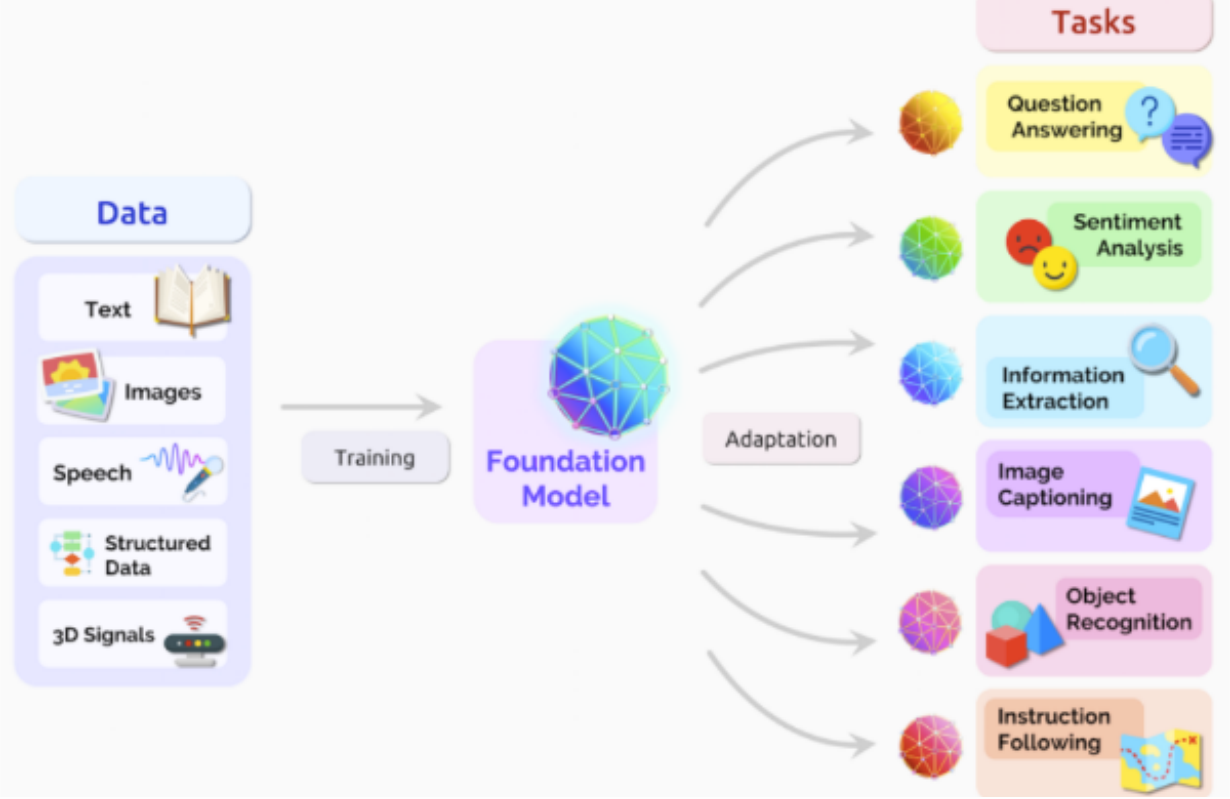
2023

Prepared By :
Abi Aryan

Supervised By :
Reena Jailwala

TABLE OF CONTENTS

LLMs: The Next Frontier in AI



LLMs are a type of AI that can mimic human intelligence by processing and generating text. They are trained on massive datasets of text and code, and they can be used for a variety of tasks, such as:

- Question answering: Answering your questions in an informative way, even if they are open ended, challenging, or strange.
- Text generation: Creating different creative text formats, like poems, code, scripts, musical pieces, email, letters, etc.
- Summarization: Creating a concise and informative summary of a text.
- Translation: Translating text from one language to another.
- Code generation: Writing code in a variety of programming languages.
- Data analysis: Analyzing data and identifying patterns.
- Creative writing: Writing different kinds of creative content, such as poems, stories, and scripts.

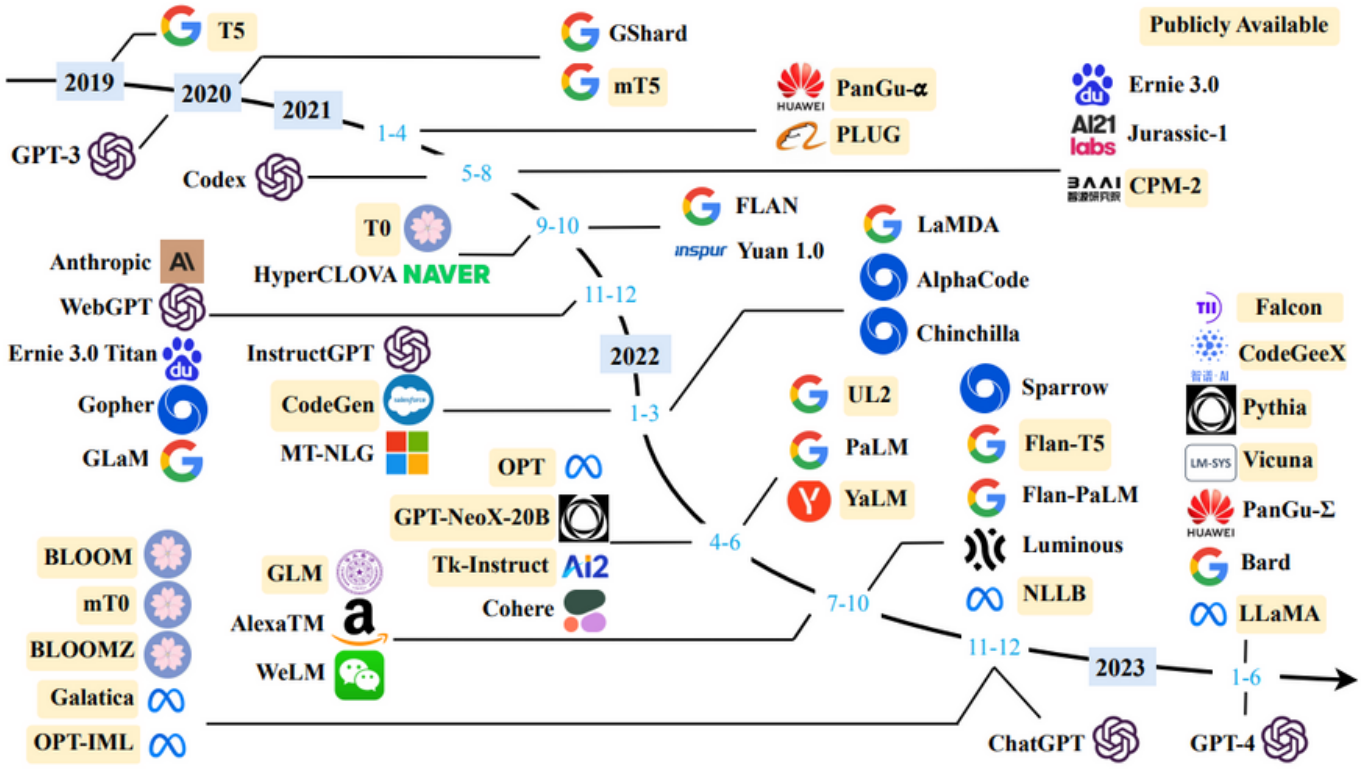
LLMs are still under development, but they have the potential to revolutionize the way we interact with computers. They can be used to create more natural and intuitive user interfaces, and they can also be used to automate tasks that are currently done by humans.

In 2023, we are likely to see even more advances in LLMs. They are expected to be used in a wider range of applications, and they are also expected to become more powerful and accurate. This will have a significant impact on the way we live and work, and it will usher in a new era of AI and ML innovation.

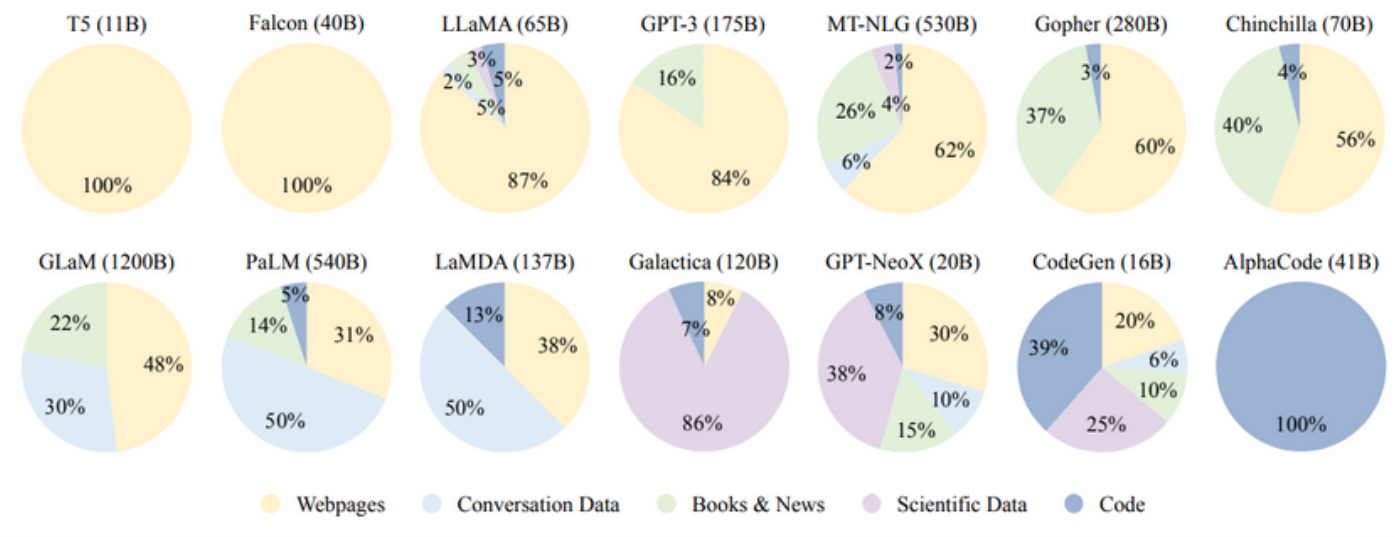
Business Factors in Choosing a LLM

Choosing which large language model (LLM) to use can be a difficult task, as there are many factors to consider. Here are some of the most important factors to keep in mind:

1. **Model Performance and Fit:** Enterprises need to assess how well a particular LLM performs for their specific use case. They should evaluate metrics such as accuracy, language fluency, contextual understanding, and the ability to generate relevant and coherent responses. It's important to choose an LLM that aligns with the specific requirements and quality standards of the intended application.
2. **Customizability and Fine-tuning:** The ability to fine-tune the LLM to adapt it to specific use cases is crucial. Enterprises should consider the flexibility and ease of fine-tuning the model to improve its performance on their particular tasks or domains. Customization options could include adjusting the model's architecture, training on domain-specific data, or incorporating task-specific prompts. The availability and effectiveness of fine-tuning mechanisms should be evaluated.
3. **Ethical and Responsible AI:** Enterprises must consider ethical implications when selecting an LLM. It's important to assess factors like bias, fairness, and inclusivity in the model's training data and outputs. Transparent documentation of the model's behavior, potential risks, and mitigation strategies is also valuable. Adhering to ethical guidelines and ensuring responsible AI practices should be a priority.
4. **Third-Party Verification and Openness:** Consideration should be given to LLMs that have undergone third-party verification or external audits. Independent verification of model performance, fairness, safety, and compliance can provide additional assurance. Openness, including availability of model architecture and training methodologies, fosters transparency and allows external scrutiny.
5. **Observability and Debugging:** Enterprises should assess the observability and debugging capabilities of the LLM. The ability to monitor and understand the model's decision-making process, identify potential issues or biases, and debug any problems that arise is crucial for maintaining control and trust in the system.
6. **Cost and Scalability:** Cost considerations are important, especially for enterprises working with large-scale deployments or resource-intensive applications. The pricing structure, infrastructure requirements, and scalability of the LLM should align with the enterprise's budget and expected usage.
7. **Security and Privacy:** Security measures and data privacy safeguards are essential when dealing with sensitive or confidential information. Enterprises should evaluate the LLM provider's security practices, data handling protocols, and compliance with



A timeline of existing large language models, arXiv:2303.18223



Ratios of various data sources in the pre-training data for existing LLMs, arXiv:2303.18223.

Technical Factors in choosing a LLM

Deciding which provider and foundational model would be a better choice depends on several factors including number of parameters, size of context window, training type, inference speed, cost, fine-tunability as well as data security.

1. N**umber of Parameters:** The number of parameters in a model can influence its capacity and performance. Larger models with more parameters often have the potential to capture more complex patterns and achieve higher accuracy. However, larger models also require more computational resources for training and inference.
2. **Size of Context Window:** The context window refers to the amount of text or context that the model considers when generating responses or predictions. Models with larger context windows have a broader understanding of the context, which can be advantageous for generating more contextually relevant outputs. However, larger context windows can also increase computational requirements and inference time.
3. **Training Type:** Different models can be trained using various methods, such as supervised learning, unsupervised learning, or self-supervised learning. The training type can impact the model's capabilities and generalization across different tasks. Understanding the training methodology and evaluating its suitability for specific use cases is important.
4. **Inference Speed:** Inference speed is a crucial factor, especially in real-time or latency-sensitive applications. Faster inference speeds allow for more responsive systems and better user experiences. Models with smaller architectures or optimized implementations tend to have faster inference times.
5. **Cost:** The cost of using a particular LLM includes considerations such as licensing fees, computational resources required for training and inference, and ongoing maintenance costs. Enterprises should evaluate the cost implications and ensure that the chosen LLM aligns with their budget and cost expectations.
6. **Fine-tunability:** The ability to fine-tune the model to adapt it to specific use cases is valuable. Enterprises should consider the level of fine-tuning flexibility and the availability of resources and documentation for this process. Some models may offer more extensive fine-tuning capabilities than others.
7. **Data Security:** Data security is of paramount importance when working with LLMs. Enterprises should assess the data security practices of the provider, including encryption methods, access controls, and compliance with privacy regulations. Protecting sensitive data and ensuring secure handling and storage is crucial.

While all the factors mentioned pose their own challenges, one of the biggest challenges currently is striking the right balance between model performance and computational resources.

Key Considerations

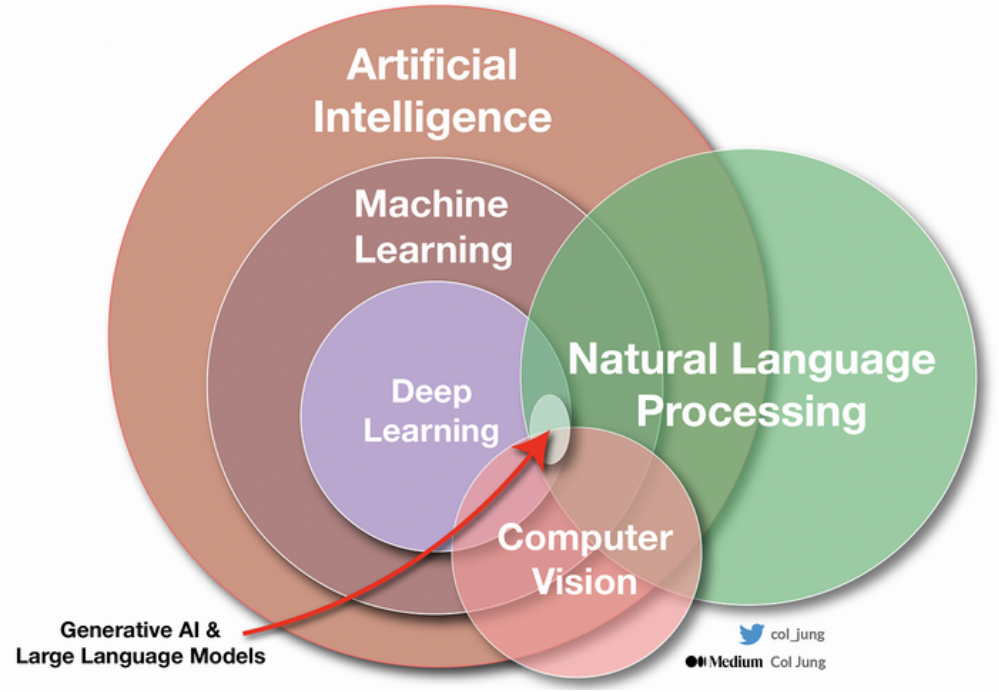
	Model	Release Time	Size (B)	Base Model	Adaptation IT	Adaptation RLHF	Pre-train Data Scale	Latest Data Timestamp	Hardware (GPUs / TPUs)	Training Time	Evaluation ICL	Evaluation CoT
Publicly Available	T5 [73]	Oct-2019	11	-	-	-	1T tokens	Apr-2019	1024 TPU v3	-	✓	-
	mT5 [74]	Oct-2020	13	-	-	-	1T tokens	-	-	-	✓	-
	PanGu-α [75]	Apr-2021	13*	-	-	-	1.1TB	-	2048 Ascend 910	-	✓	-
	CPM-2 [76]	Jun-2021	198	-	-	-	2.6TB	-	-	-	-	-
	T0 [28]	Oct-2021	11	T5	✓	-	-	-	512 TPU v3	27 h	✓	-
	CodeGen [77]	Mar-2022	16	-	-	-	577B tokens	-	-	-	✓	-
	GPT-NeoX-20B [78]	Apr-2022	20	-	-	-	825GB	-	96 40G A100	-	✓	-
	Tk-Instruct [79]	Apr-2022	11	T5	✓	-	-	-	256 TPU v3	4 h	✓	-
	UL2 [80]	May-2022	20	-	-	-	1T tokens	Apr-2019	512 TPU v4	-	✓	✓
	OPT [81]	May-2022	175	-	-	-	180B tokens	-	992 80G A100	-	✓	-
	NLLB [82]	Jul-2022	54.5	-	-	-	-	-	-	-	✓	-
	GLM [83]	Oct-2022	130	-	-	-	400B tokens	-	768 40G A100	60 d	✓	-
	Flan-T5 [64]	Oct-2022	11	T5	✓	-	-	-	-	-	✓	✓
	BLOOM [69]	Nov-2022	176	-	-	-	366B tokens	-	384 80G A100	105 d	✓	-
	mT0 [84]	Nov-2022	13	mT5	✓	-	-	-	-	-	✓	-
	Galactica [35]	Nov-2022	120	-	-	-	106B tokens	-	-	-	✓	✓
	BLOOMZ [84]	Nov-2022	176	BLOOM	✓	-	-	-	-	-	✓	-
	OPT-IML [85]	Dec-2022	175	OPT	✓	-	-	-	128 40G A100	-	✓	✓
	LLaMA [57]	Feb-2023	65	-	-	-	1.4T tokens	-	2048 80G A100	21 d	✓	-
	CodeGeeX [86]	Sep-2022	13	-	-	-	850B tokens	-	1536 Ascend 910	60 d	✓	-
	Pythia [87]	Apr-2023	12	-	-	-	300B tokens	-	256 40G A100	-	✓	-
Closed Source	GPT-3 [55]	May-2020	175	-	-	-	300B tokens	-	-	-	✓	-
	GShard [88]	Jun-2020	600	-	-	-	1T tokens	-	2048 TPU v3	4 d	-	-
	Codex [89]	Jul-2021	12	GPT-3	-	-	100B tokens	May-2020	-	-	✓	-
	ERNIE 3.0 [90]	Jul-2021	10	-	-	-	375B tokens	-	384 V100	-	✓	-
	Jurassic-1 [91]	Aug-2021	178	-	-	-	300B tokens	-	800 GPU	-	✓	-
	HyperCLOVA [92]	Sep-2021	82	-	-	-	300B tokens	-	1024 A100	13.4 d	✓	-
	FLAN [62]	Sep-2021	137	LaMDA-PT	✓	-	-	-	128 TPU v3	60 h	✓	-
	Yuan 1.0 [93]	Oct-2021	245	-	-	-	180B tokens	-	2128 GPU	-	✓	-
	Anthropic [94]	Dec-2021	52	-	-	-	400B tokens	-	-	-	✓	-
	WebGPT [72]	Dec-2021	175	GPT-3	-	✓	-	-	-	-	✓	-
	Gopher [59]	Dec-2021	280	-	-	-	300B tokens	-	4096 TPU v3	920 h	✓	-
	ERNIE 3.0 Titan [95]	Dec-2021	260	-	-	-	-	-	-	-	✓	-
	GLaM [96]	Dec-2021	1200	-	-	-	280B tokens	-	1024 TPU v4	574 h	✓	-
	LaMDA [63]	Jan-2022	137	-	-	-	768B tokens	-	1024 TPU v3	57.7 d	-	-
	MT-NLG [97]	Jan-2022	530	-	-	-	270B tokens	-	4480 80G A100	-	✓	-
	AlphaCode [98]	Feb-2022	41	-	-	-	967B tokens	Jul-2021	-	-	-	-
	InstructGPT [61]	Mar-2022	175	GPT-3	✓	✓	-	-	-	-	✓	-
	Chinchilla [34]	Mar-2022	70	-	-	-	1.4T tokens	-	-	-	✓	-
	PaLM [56]	Apr-2022	540	-	-	-	780B tokens	-	6144 TPU v4	-	✓	✓
	AlexaTM [99]	Aug-2022	20	-	-	-	1.3T tokens	-	128 A100	120 d	✓	✓
	Sparrow [100]	Sep-2022	70	-	-	✓	-	-	64 TPU v3	-	✓	-
	WeLM [101]	Sep-2022	10	-	-	-	300B tokens	-	128 A100 40G	24 d	✓	-
	U-PaLM [102]	Oct-2022	540	PaLM	-	-	-	-	512 TPU v4	5 d	✓	✓
	Flan-PaLM [64]	Oct-2022	540	PaLM	✓	-	-	-	512 TPU v4	37 h	✓	✓
	Flan-U-PaLM [64]	Oct-2022	540	U-PaLM	✓	-	-	-	-	-	✓	✓
	GPT-4 [46]	Mar-2023	-	-	✓	✓	-	-	-	-	✓	✓
	PanGu-Σ [103]	Mar-2023	1085	PanGu-α	-	-	329B tokens	-	512 Ascend 910	100 d	✓	-

Statistics of large language models, arXiv:2303.18223

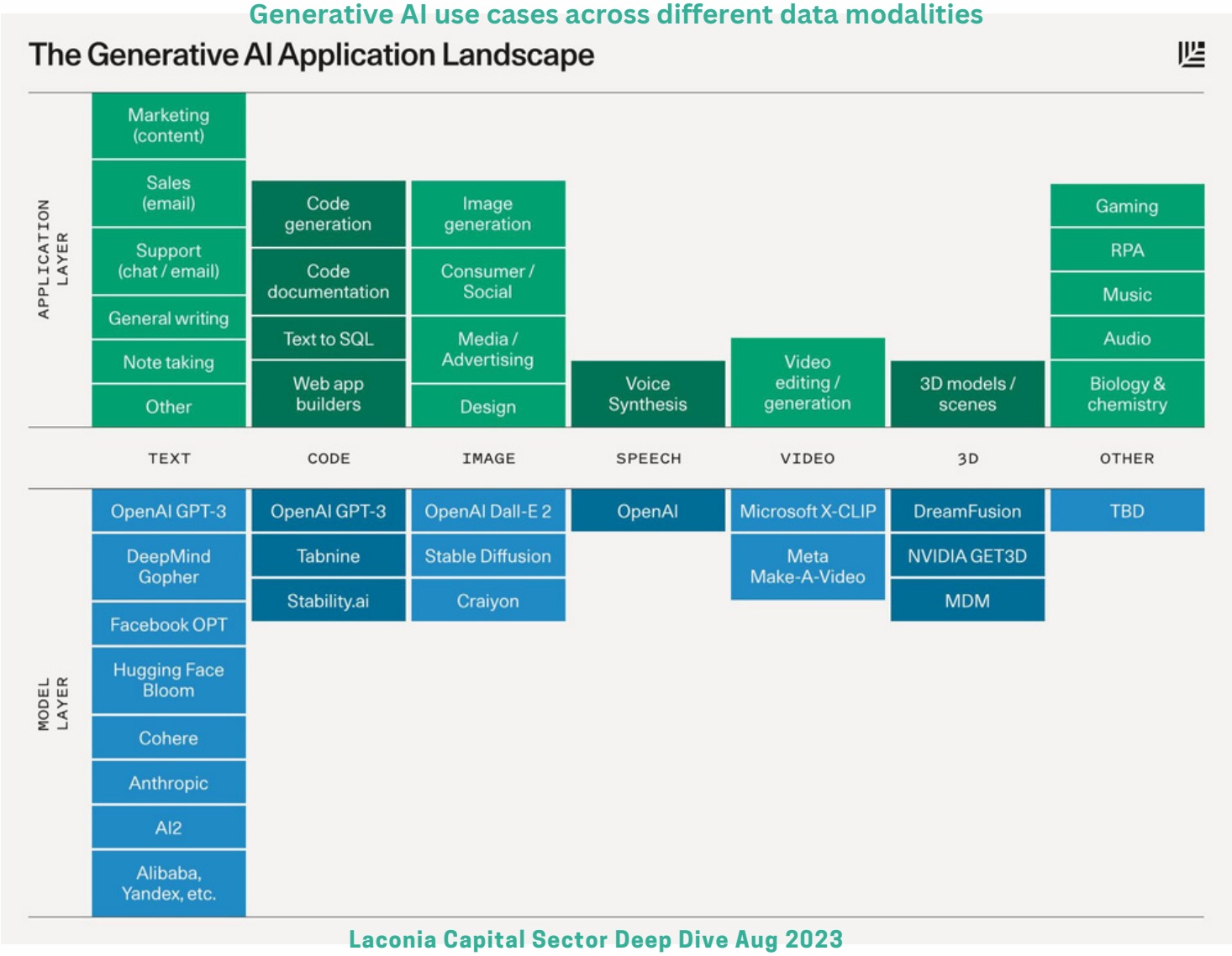
In Context-Learning (ICL) - In-context learning (ICL) is the ability of a language model to learn a new task from a few examples in the context of a prompt. It is an important evaluation metric for language models because it measures their ability to generalize to new tasks without being explicitly trained on them.

Chain of Thought (CoT) - chain of thought can be used as a way to evaluate the performance of a language model. For example, a language model could be given a prompt that requires it to follow a logical sequence of ideas. The model's output could then be scored based on how well it follows the prompt.

Placing Generative AI and LLMs in the bigger picture



Generative AI use cases across different data modalities



Discriminative vs Generative NLP Models: Different Use-Cases

Discriminative and generative machine learning language models have different strengths and weaknesses, so they are used for different tasks.

Discriminative language models are better at tasks that require understanding the relationship between words and their meaning. For example, they can be used for:

- **Text classification: Categorizing text into different classes, such as news articles, product reviews, or spam.**
- **Named entity recognition: Identifying named entities in text, such as people, places, and organizations.**
- **Sentiment analysis: Identifying the sentiment of text, such as whether it is positive, negative, or neutral.**

Generative language models are better at tasks that require creating new text. For example, they can be used for:

- **Text summarization: Generating a shorter version of a text that captures the main points.**
- **Machine translation: Translating text from one language to another.**
- **Text generation: Generating new text, such as poems, code, or scripts.**

Here are some specific examples of discriminative and generative language models:

- **Discriminative language models:** Logistic regression, support vector machines, conditional random fields etc
- **Generative language models:** Naive Bayes, Bayesian networks, hidden Markov models, etc.

In recent years, there has been a growing interest in using deep learning techniques to train generative language models. **Large language models (LLMs) are generative models.** They are trained on a massive dataset of text and code, and they learn to generate new text that is similar to the text they were trained on. LLMs can be used for a variety of tasks, such as text summarization, machine translation, and text generation.

Here are some examples of how LLMs can be used:

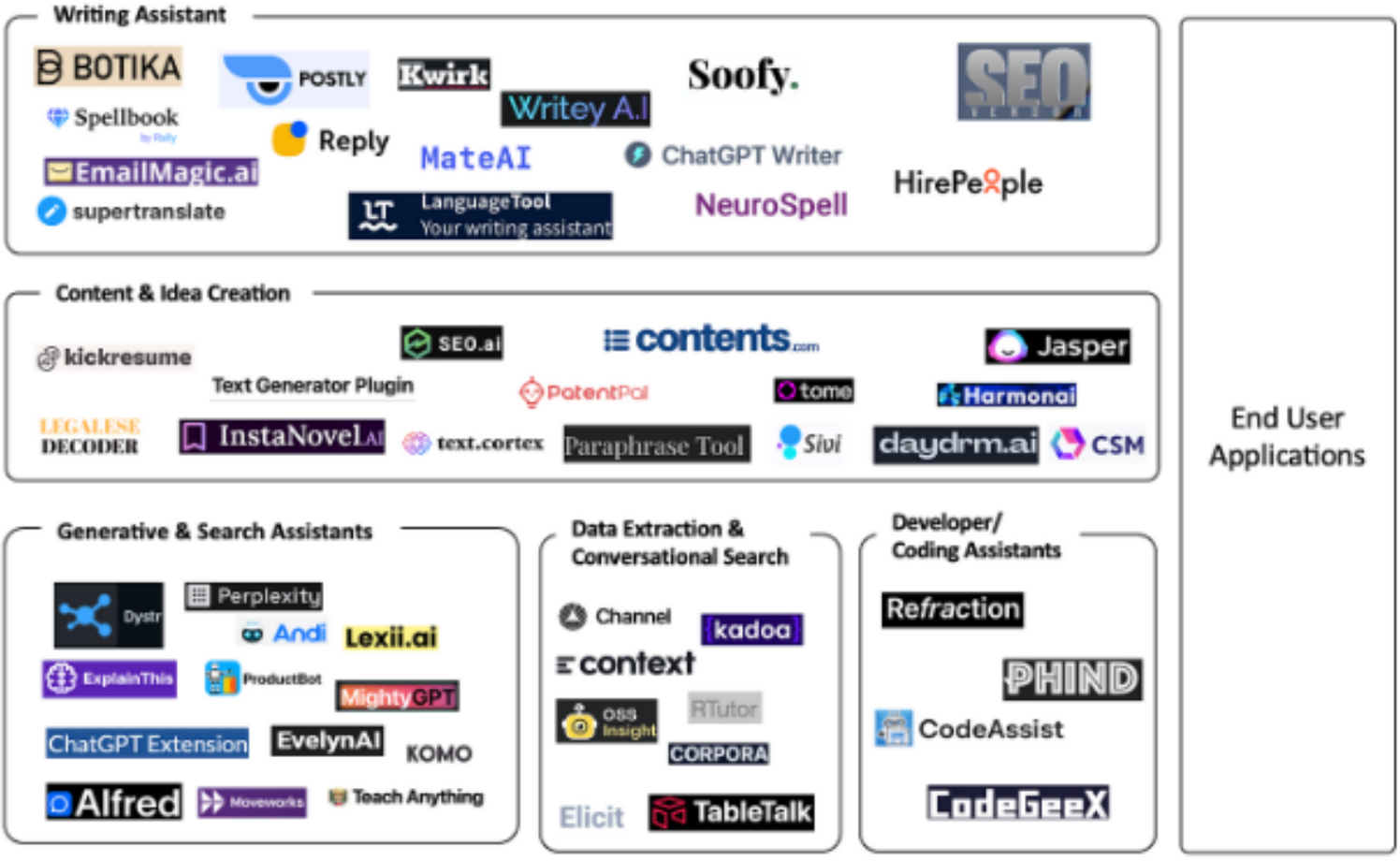
- **Text summarization: An LLM can be used to generate a shorter version of a text that captures the main points. For example, an LLM could be used to summarize a news article or a research paper.**
- **Machine translation: An LLM can be used to translate text from one language to another. For example, an LLM could be used to translate a website from English to Spanish.**
- **Text generation: An LLM can be used to generate new text, such as poems, code, or scripts. For example, an LLM could be used to generate a new poem or to write a new computer program.**

LLMs are still under development, but they have the potential to revolutionize the way we interact with computers. In the future, we can expect to see LLMs being used for a wider range of tasks, such as creating realistic chatbots and generating creative content.

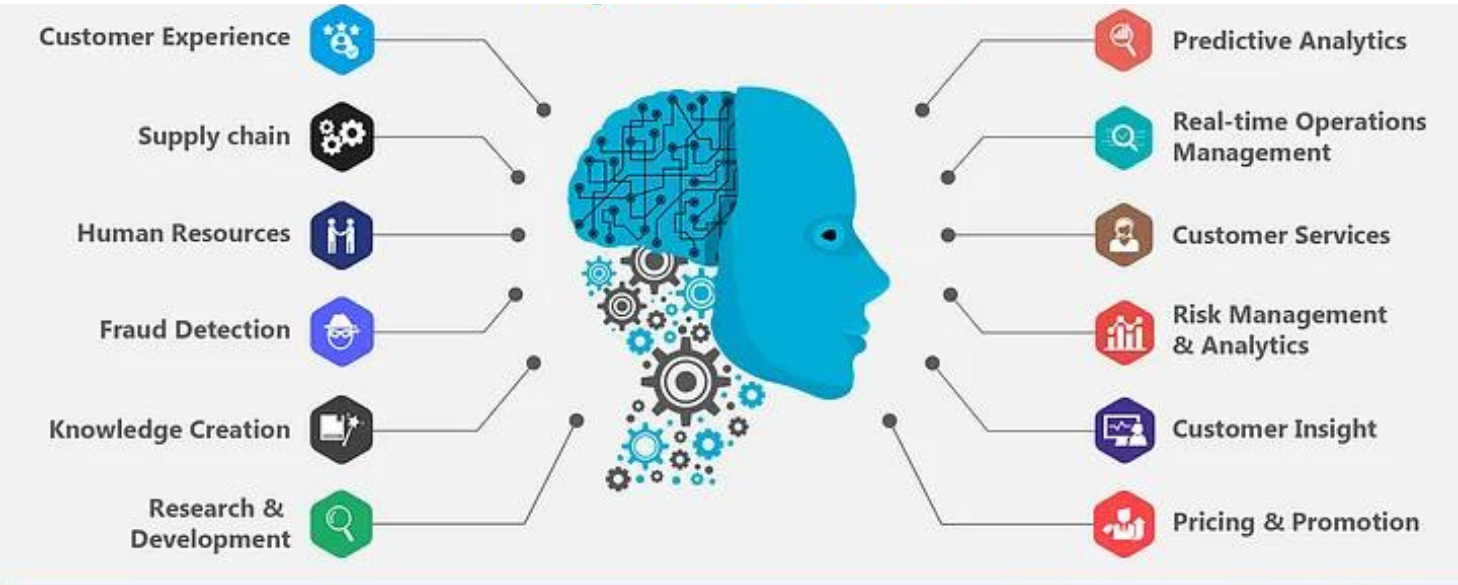
Use Cases of LLMs across industries

Large Language Models (LLMs) are a rapidly expanding field of research and development. They are being used in a variety of industries, including healthcare, retail, tech, and more. Here are some common use cases of LLMs:

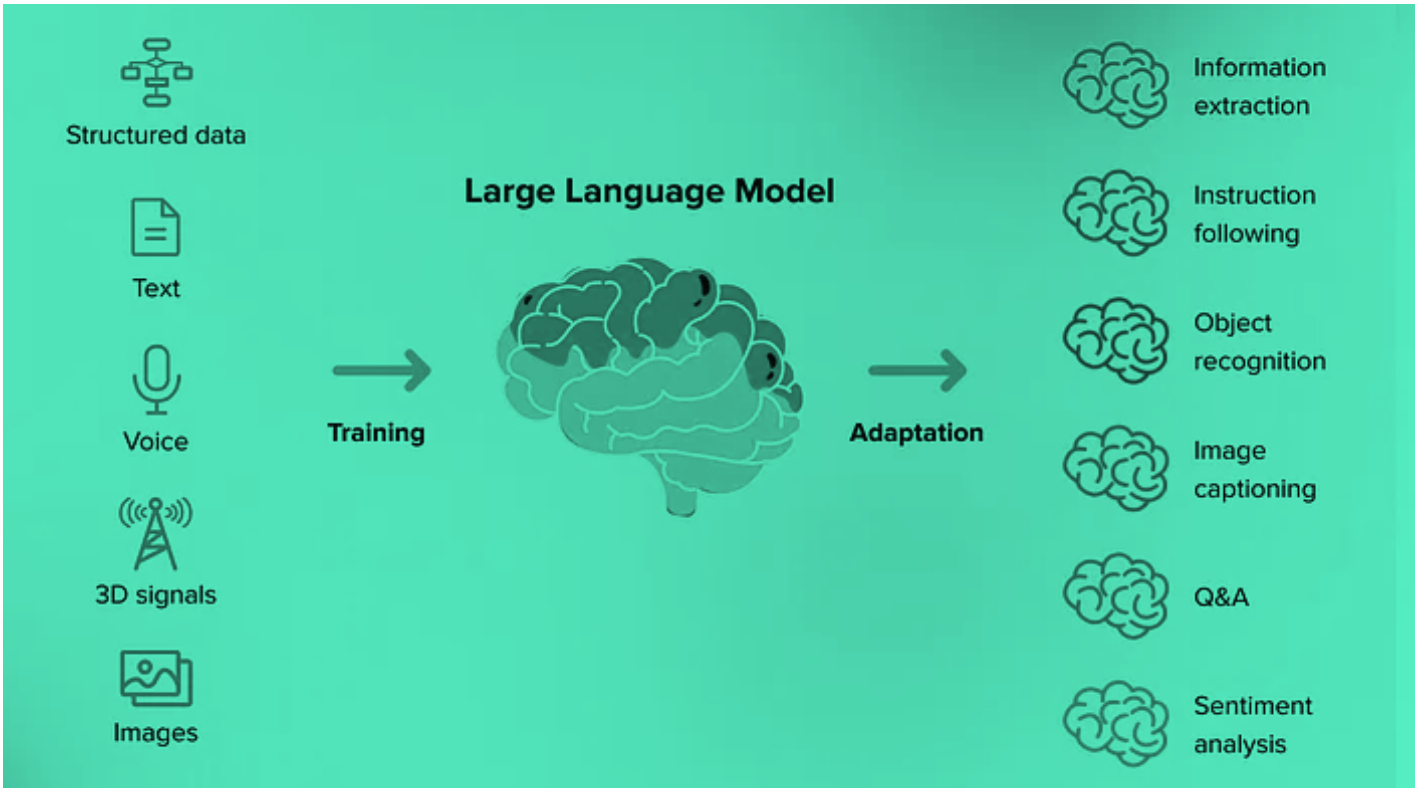
1. **Text summarization:** LLMs can summarize blocks of text or multiple documents into a shorter version while retaining the most important information
2. **Text generation:** LLMs can generate text on any topic they are trained on. This can be used to create chatbots, virtual assistants, and conversational AI.
3. **Sentiment analysis:** LLMs can analyze text to determine the sentiment behind it. This can be used to gauge customer satisfaction or public opinion.
4. **Content creation:** LLMs can be used to create content such as articles, stories, and even poetry.
5. **Question answering:** LLMs can answer questions based on the context provided to them. This can be used in chatbots or virtual assistants to provide quick answers to common questions
6. **Clustering:** LLMs can group similar data points together based on their similarity in meaning or context.
7. **Classification:** LLMs can classify data points into different categories based on their content or meaning.

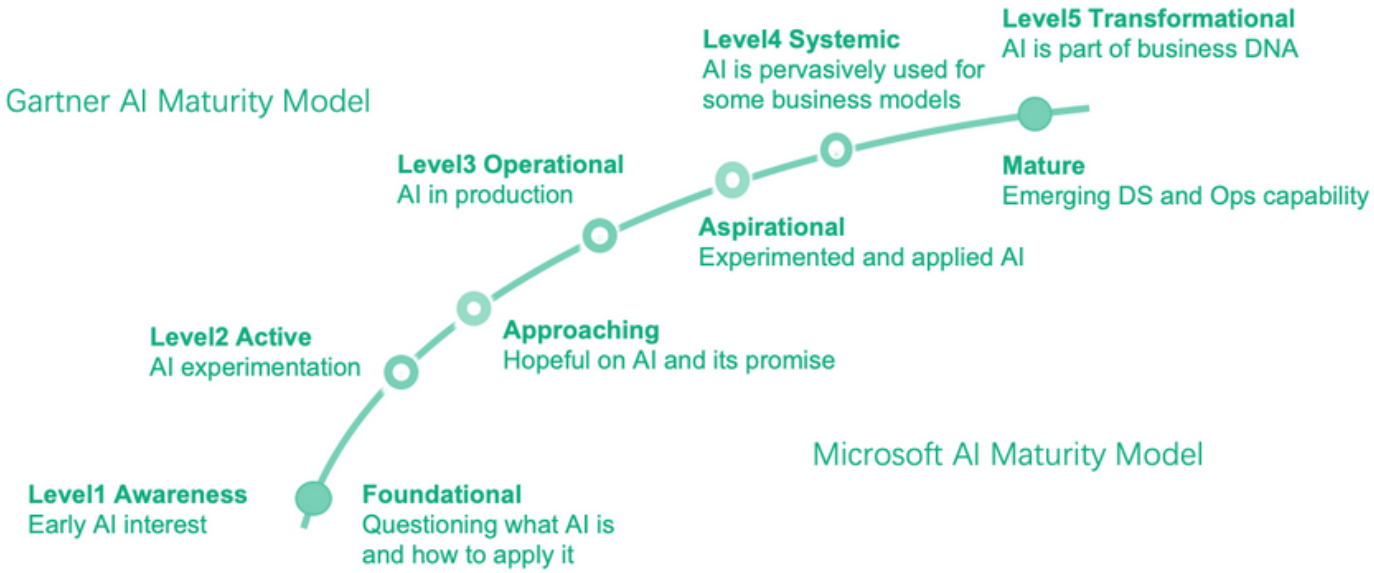


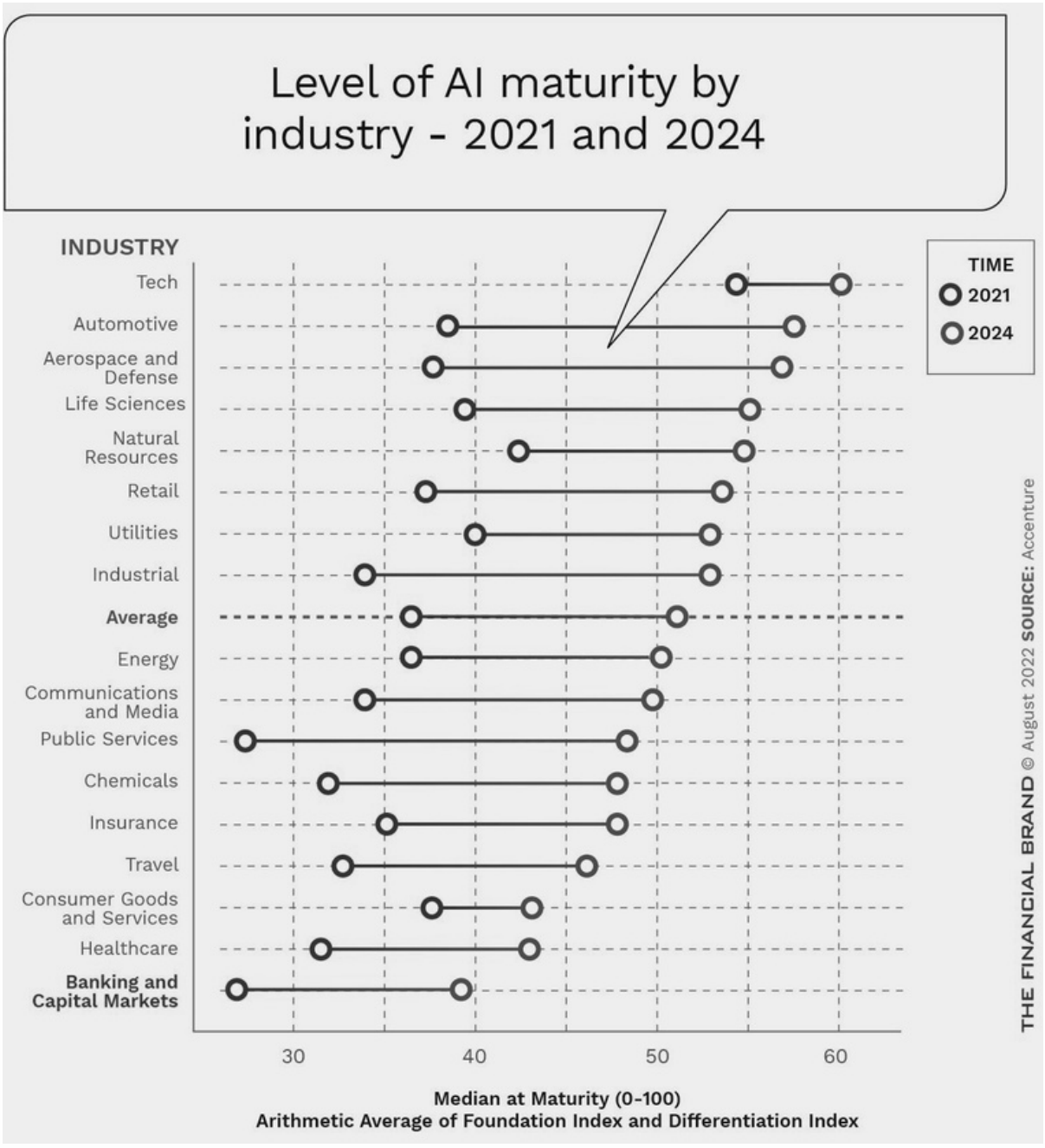
AI/ML Use-Cases



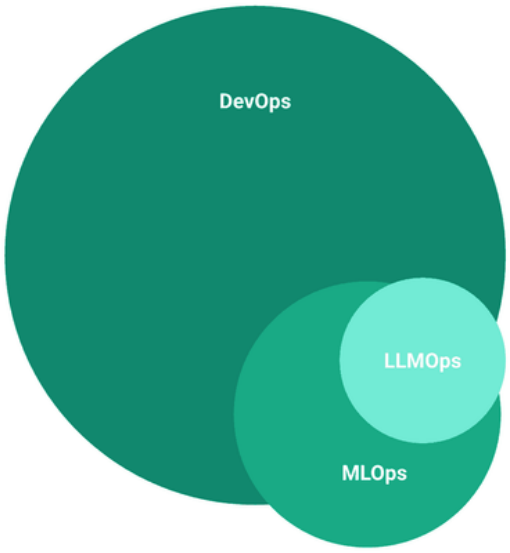
Adapting LLMs in enterprise




Gartner AI Maturity Model
Level1 Awareness
Early AI interest
Level2 Active
AI experimentation
Level3 Operational
AI in production
Level4 Systemic
AI is pervasively used for some business models
Level5 Transformational
AI is part of business DNA
Foundational
Questioning what AI is and how to apply it
Approaching
Hopeful on AI and its promise
Aspirational
Experimented and applied AI
Mature
Emerging DS and Ops capability
Microsoft AI Maturity Model


Level of AI maturity by industry - 2021 and 2024
INDUSTRY
Tech
Automotive
Aerospace and Defense
Life Sciences
Natural Resources
Retail
Utilities
Industrial
Average
Energy
Communications and Media
Public Services
Chemicals
Insurance
Travel
Consumer Goods and Services
Healthcare
Banking and Capital Markets
TIME
2021
2024
30
40
50
60
Median at Maturity (0-100)
Arithmetic Average of Foundation Index and Differentiation Index
THE FINANCIAL BRAND © August 2022 SOURCE: Accenture

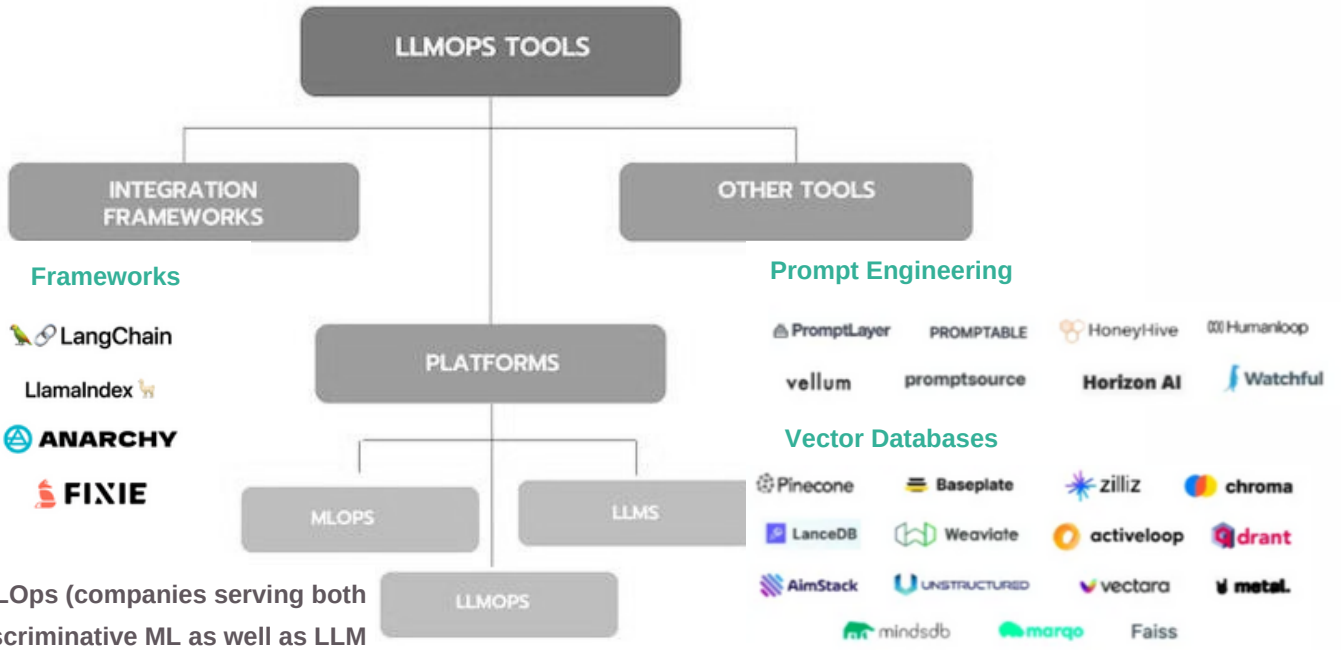
From MLOps to LLMOps



API based generative models come with own unique challenges, thus necessitating a new framework called LLMOps

LLMOps is an emerging field that draws from MLOps and DevOps, and is focused on managing the entire lifecycle of large language models, for robus, reliable and efficient integration of large langauge models in enterprise.

LLMOps Landscape



Frameworks

LangChain
LlamaIndex
ANARCHY
FIXIE

Prompt Engineering

PromptLayer PROMPTABLE HoneyHive Humanloop
vellum promptsource Horizon AI Watchful

Vector Databases

Pinecone Baseplate zilliz chroma
LanceDB Weaviate activeloop qdrant
AimStack UNSTRUCTURED vectara metal.
mindsdb marqo Faiss

LLMs

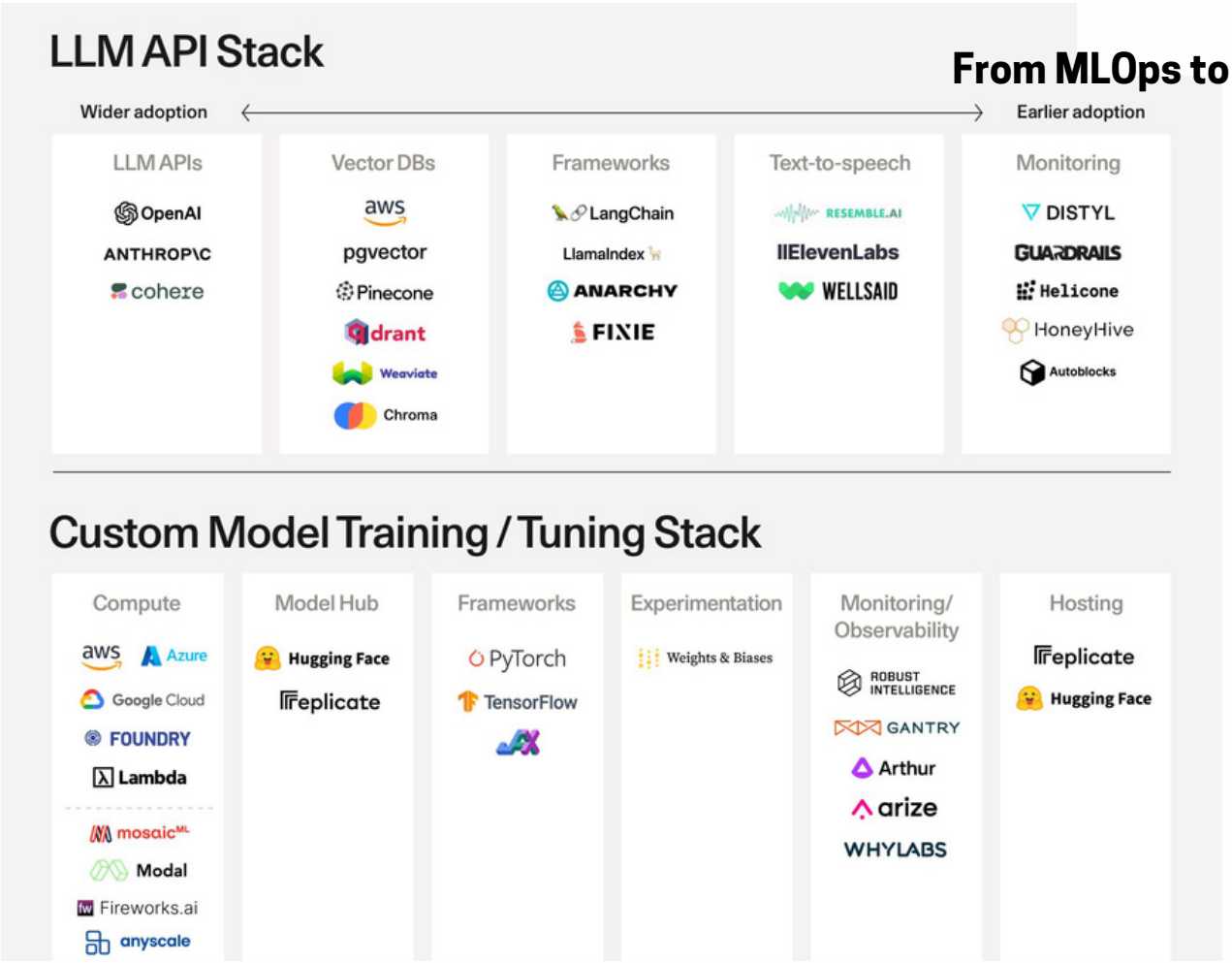
OpenAI AI21labs PaLM 2 cerebras
ANTHROPIC cohere BLOOM FLAN-T5
LLaMA amazon TITAN stability.ai EleutherAI
mosaicML Mistral AI

MLOps (companies serving both Discriminative ML as well as LLM applications)

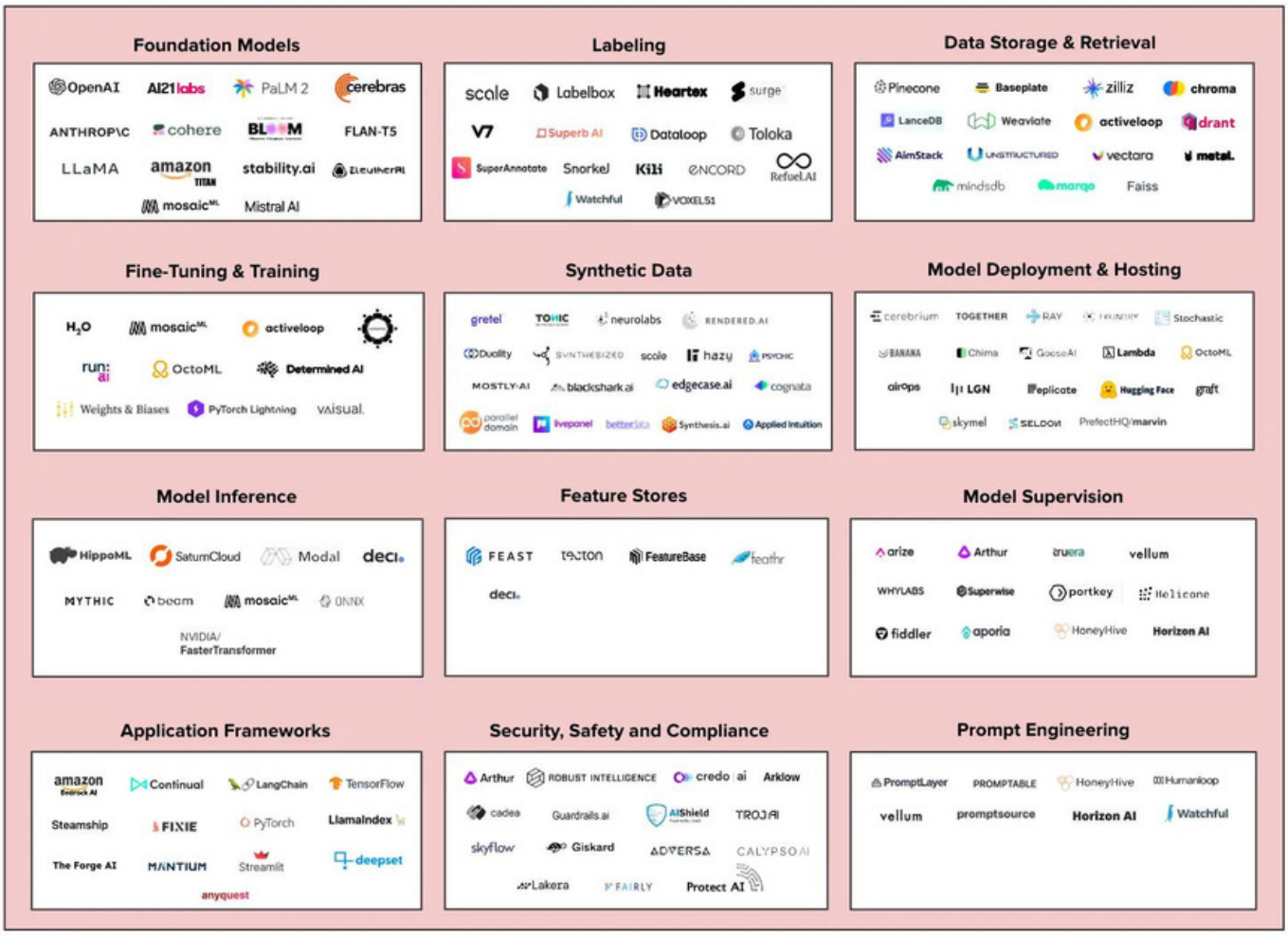
- Application Frameworks
- Labeling
- Data Storage & Retrival
- Fine-Tuning and Training
- Synthetic Data
- Model Deployment & Hosting
- Model Inference
- Feature Stores
- Model Supervision
- Security, Safety & Compliance

LLMOps (companies serving only LLM applications)

- Fine-Tuning and Training
- Model Deployment & Hosting
- Model Inference
- Model Supervision
- Security, Safety & Compliance



The New Generative AI Infra Stack, Cowboy Ventures

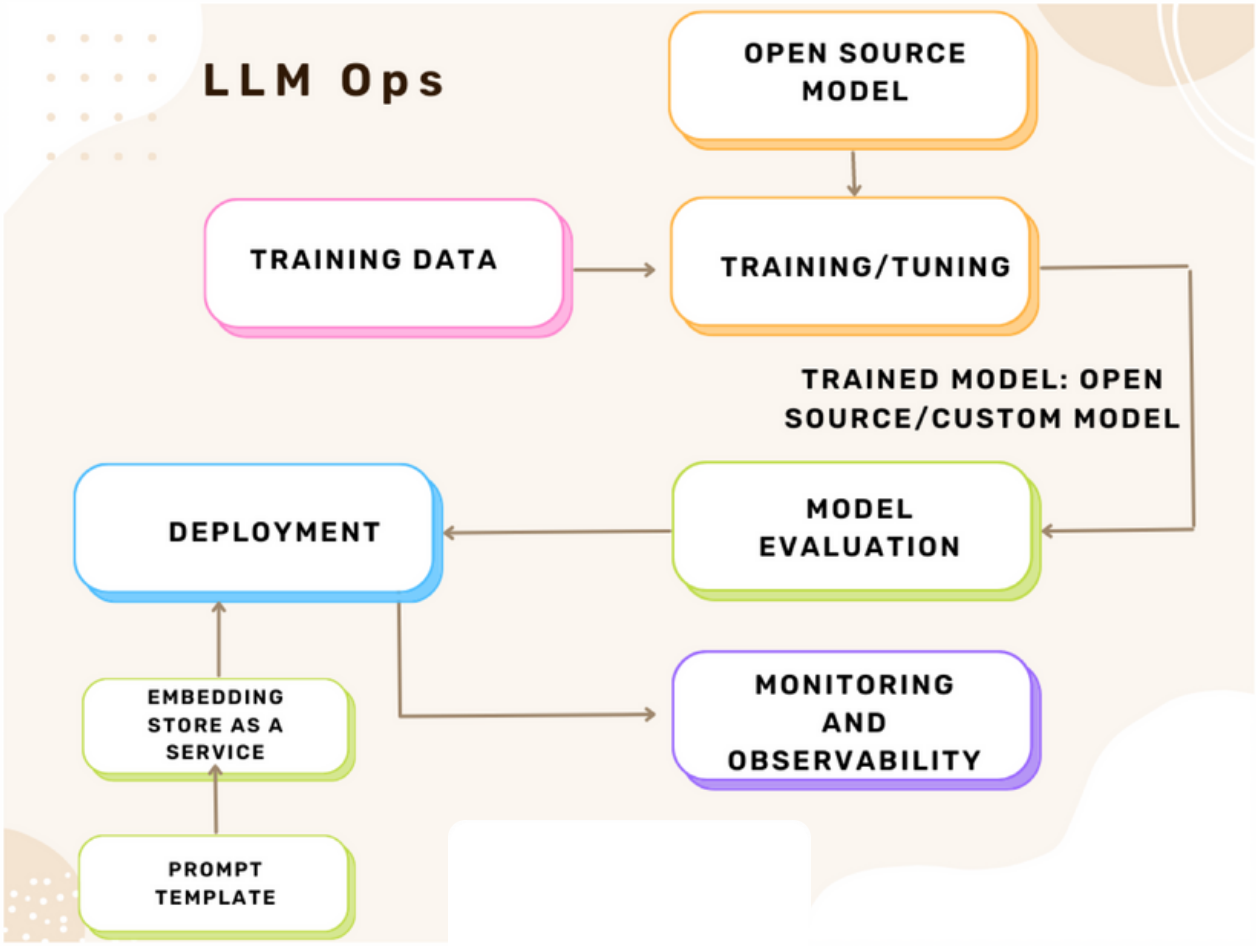


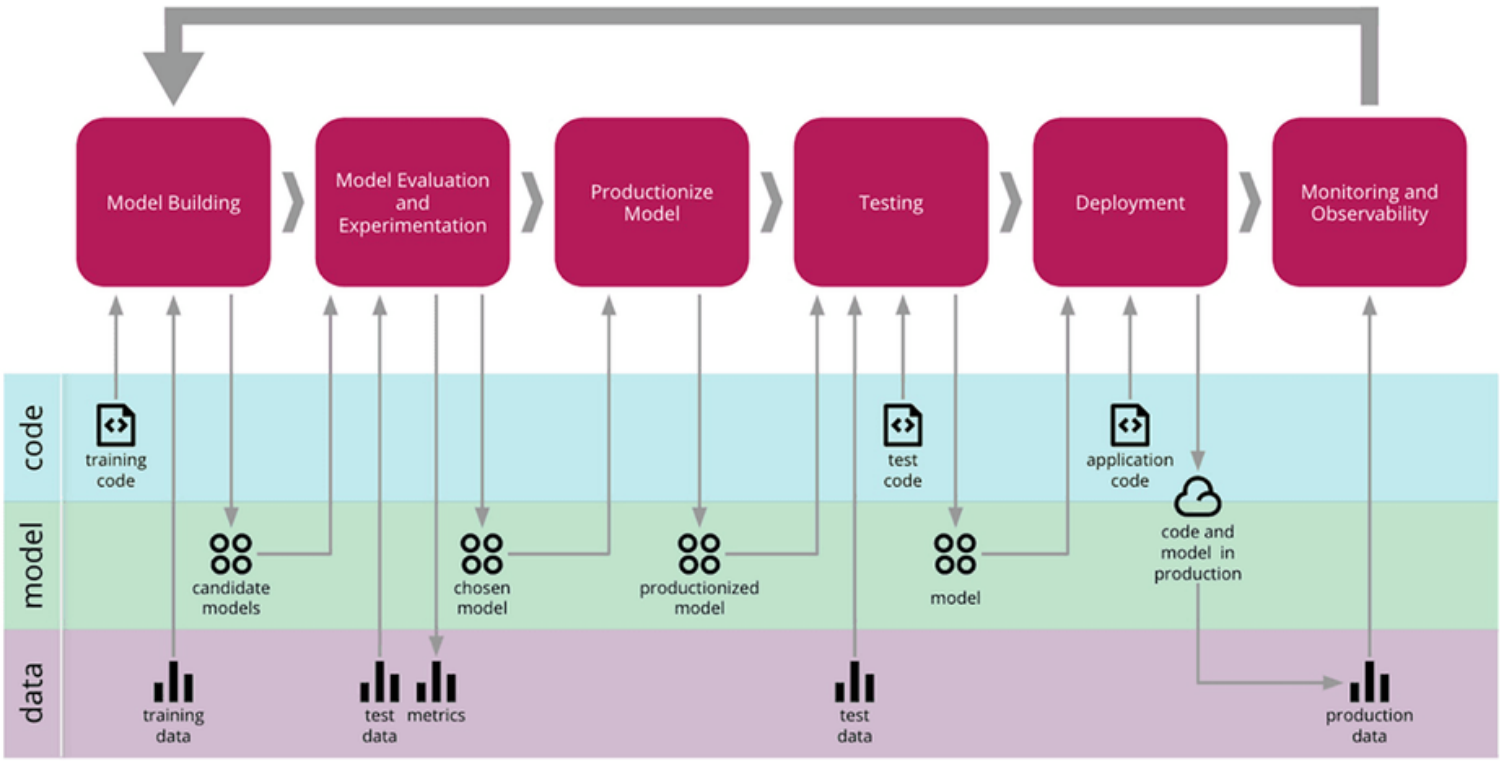
Laconia Capital Sector Deep Dive Aug 2023

What Changed between MLOps and LLMOps?

Pipeline Component	MLOps	LLMOps
Data Collection and Labeling	Sourcing new data, wrangling data, cleaning data, and data labeling (outsourced or in-house).	Requires larger scale data collection and emphasizes data diversity and representativeness. May need automated or semi-automated labeling techniques, such as pre-trained models for data annotation, active learning, or weak supervision methods.
Feature Engineering and Model Experimentation	Improving ML performance through data-driven or model-driven experiments, such as creating new features or changing model architecture.	Feature engineering becomes less relevant due to LLMs' ability to learn effective feature representations from raw data, for the near future shifting towards prompt design and fine-tuning. Model experimentation continues to play a crucial role but will return to the earlier days of the data science evolution in the short-term, centered around getting consistently performing models for a specific use case, requiring quick iteration speeds to create value. Long-term, unclear where we are heading in this space due to the rapid advancements & shifts in LLM capabilities.
Model Evaluation and Deployment	Computing metrics (e.g., accuracy) over a validation dataset. Deployment includes staging, A/B testing, and keeping records for rollbacks.	Evaluation and deployment are more nuanced, requiring a broader set of metrics and techniques to assess fairness, robustness, and interpretability, not just accuracy. These could include "golden test sets", which are human-validated feedback on narrow questions/tasks. Deployment needs robust tools for managing the training data, training process, versioning of models, and possibly switching between different models depending on the use case. Drift detection systems and measures to handle adversarial attacks or misaligned inputs are also crucial. The complexity of these tasks may necessitate new roles or specialized skills.
ML Pipeline Monitoring and Response	Tracking live metrics, investigating prediction quality, patching the model with non-ML heuristics, and adding failures to the evaluation set.	Involves tracking model performance across multiple tasks, languages, and domains using tools like "watcher models", which are other LLMs or ML models that automatically evaluate output for real-time monitoring. Monitoring for potential biases, ethical issues, or unintended consequences. Responding to issues may involve adjusting the prompt, fine-tuning the model on new examples or edge cases, or even retraining the model

LLMOps System Architecture



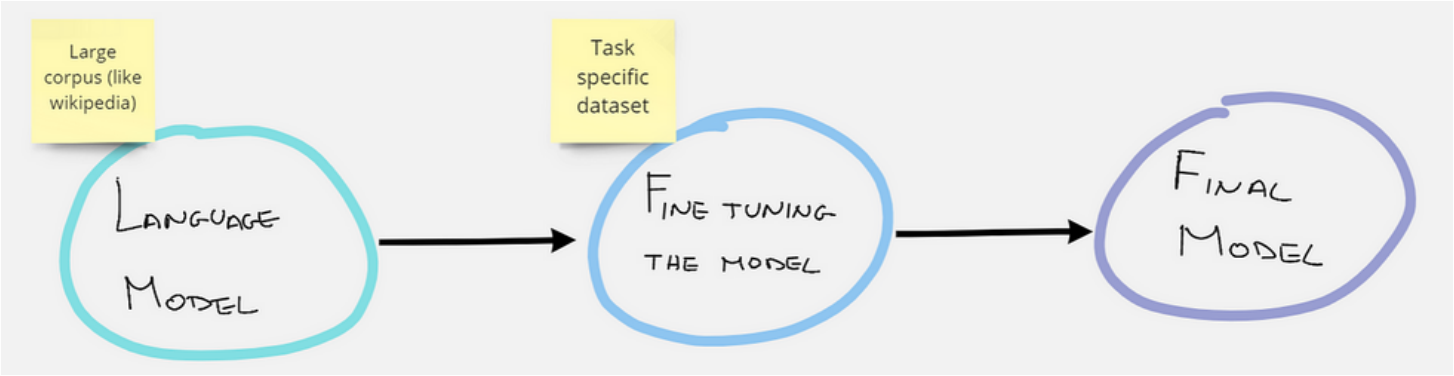


Key Challenges in LLMOps

In the context of refining MLOps practices, a comprehensive assessment of the adjustments required for machine learning (ML) workflows and prerequisites when incorporating Large Language Models (LLMs) is paramount. This analysis reveals several critical considerations:

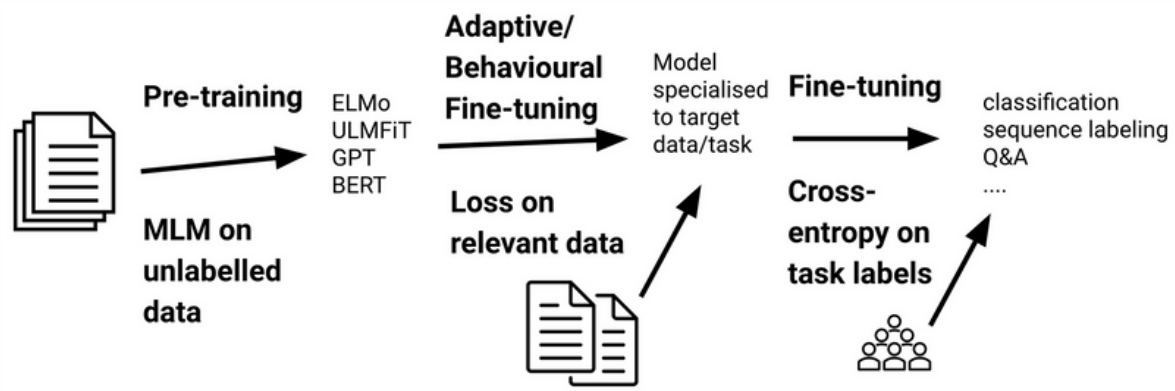
1. **Computational Resources:** The integration of extensive language models for training and optimization entails a significant upscaling of computational demands, involving intricate calculations on substantial datasets. This process is expedited through the utilization of specialized hardware such as Graphics Processing Units (GPUs), enabling rapid data-parallel operations. Access to these dedicated computational resources is pivotal for both the training phase and the subsequent deployment of LLMs. Moreover, the optimization of inference costs underscores the importance of techniques like model compression and distillation.
2. **Transfer Learning:** Contrary to conventional ML models developed from scratch, a characteristic feature of many LLMs is their foundation model, which undergoes fine-tuning using new data to enhance performance in specific domains. Fine-tuning contributes to achieving cutting-edge functionality for targeted applications while utilizing reduced data and computational resources.
3. **Human Feedback:** A notable enhancement in training LLMs has been achieved through reinforcement learning grounded in human feedback (RLHF). Given the often open-ended nature of LLM tasks, insights from end-users become instrumental in evaluating LLM efficacy. Integrating this feedback loop into LLMOps pipelines not only streamlines evaluation but also furnishes valuable data for potential future fine-tuning endeavors.
4. **Hyperparameter Tuning:** While conventional ML emphasizes hyperparameter tuning to enhance accuracy or equivalent metrics, the scenario for LLMs encompasses broader considerations including cost-efficiency and reduced computational requirements. Elements like batch sizes and learning rates significantly impact training speed and costs, necessitating purposeful optimization strategies distinct from those used in traditional ML contexts.
5. **Performance Metrics:** The well-defined performance metrics governing traditional ML models, such as accuracy, AUC, and F1 score, diverge substantially from the metrics applicable to LLMs. Metrics like bilingual evaluation understudy (BLEU) and Recall-Oriented Understudy for Gisting Evaluation (ROUGE) play a pivotal role in LLM evaluation, requiring a meticulous approach to their implementation.
6. **Prompt Engineering:** Precision and reliability in instruction-following LLMs are hinged on expertly designed prompt templates. Diligent prompt engineering mitigates the risk of content deviations generated by the model, including inadvertent content generation, prompt intrusion, unintended disclosure of sensitive data, and potential security breaches.
7. **Constructing LLM Chains or Pipelines:** The construction of LLM pipelines, facilitated by tools like LangChain or LlamaIndex, empowers the seamless connection of multiple LLM calls and interactions with external systems such as vector databases or web searches. These pipelines facilitate complex tasks, including knowledge base Q&A and responding to user queries based on a corpus of documents. Notably, the emphasis in LLM application development often shifts toward pipeline creation rather than novel LLM generation.

Fine-Tuning Large Language Models



Fine-tuning large language models like GPT-3 serves to adapt the model to specific tasks or domains, enhance its performance, and make it more usable for practical applications. There are several reasons why fine-tuning is necessary:

1. **Task-specific Adaptation:** While pre-trained models like GPT-3 possess a general understanding of language, they might not be directly suitable for specific tasks or domains. Fine-tuning allows the model to specialize in a particular task, making it more accurate and efficient.
2. **Domain Expertise:** Fine-tuning helps the model gain domain-specific knowledge that might not be present in the general pre-training data. For instance, a model intended for medical text might need to understand medical jargon and concepts, which it can learn through fine-tuning on medical text data.
3. **Data Scarcity:** In some cases, there might not be enough task-specific data to train a large model from scratch. Fine-tuning can leverage the knowledge encoded in the pre-trained model while adapting to the specific task using a smaller amount of task-specific data.
4. **Better Resource Utilization:** Fine-tuning is computationally less intensive compared to training a model from scratch. Utilizing pre-trained models and fine-tuning them is more resource-efficient and faster than training a model from the ground up.
5. **Faster Deployment:** Pre-trained models have already learned grammar, syntax, and some level of semantics from a massive amount of text. Fine-tuning allows for quicker deployment of models that can understand and generate human-like text, saving time and resources.
6. **Mitigating Biases:** Pre-trained models can inadvertently contain biases present in their training data. Fine-tuning provides an opportunity to address and mitigate these biases by focusing on fairness and inclusivity during the fine-tuning process.
7. **Control and Customization:** Fine-tuning allows developers to have more control over the model's behavior and output. By fine-tuning, you can shape the model's responses to align with specific guidelines, tones, or preferences.
8. **Improved Performance:** Fine-tuning can lead to significant improvements in performance for specific tasks. The model can learn to generate more contextually relevant and accurate responses.



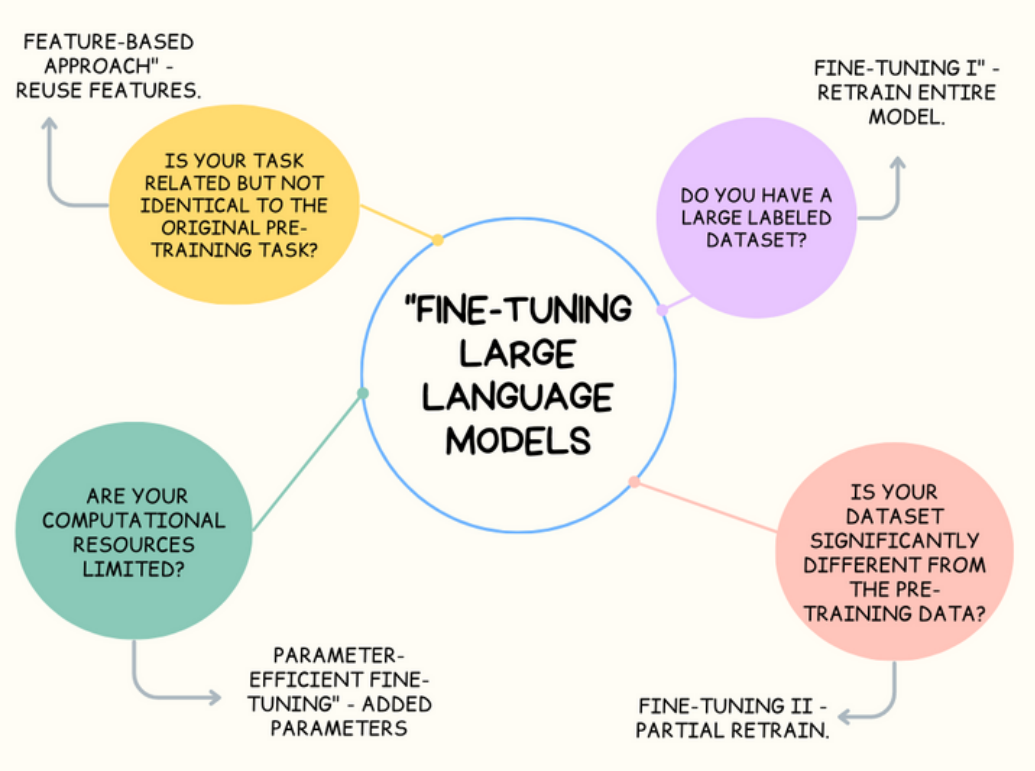
There are many tools available for fine-tuning large language models. Some of the most popular ones include:

- **Hugging Face Transformers:** This is a popular open-source library that provides easy access to pre-trained language models and utilities for fine-tuning. It supports a wide range of tasks, including text classification, question answering, and summarization.
- **PyTorch Lightning:** This is a framework for training deep learning models that is designed to be easy to use and efficient. It includes a number of features that are useful for fine-tuning large language models, such as distributed training and hyperparameter optimization.
- **DeepSpeed:** This is a library that can accelerate the training and inference of large language models by using a number of techniques, such as mixed precision and distributed training.
- **Google AI Platform:** This is a cloud platform that provides a number of tools for training and deploying machine learning models. It includes a managed instance of TPUs, which are specialized hardware accelerators for machine learning.
- **Amazon SageMaker:** This is another cloud platform that provides tools for training and deploying machine learning models. It also includes a managed instance of TPUs.
- **LMFlow:** An Extensible Toolkit for Finetuning and Inference of Large Foundation Models.

The best tool for fine-tuning a large language model will depend on the specific task and the resources that are available.

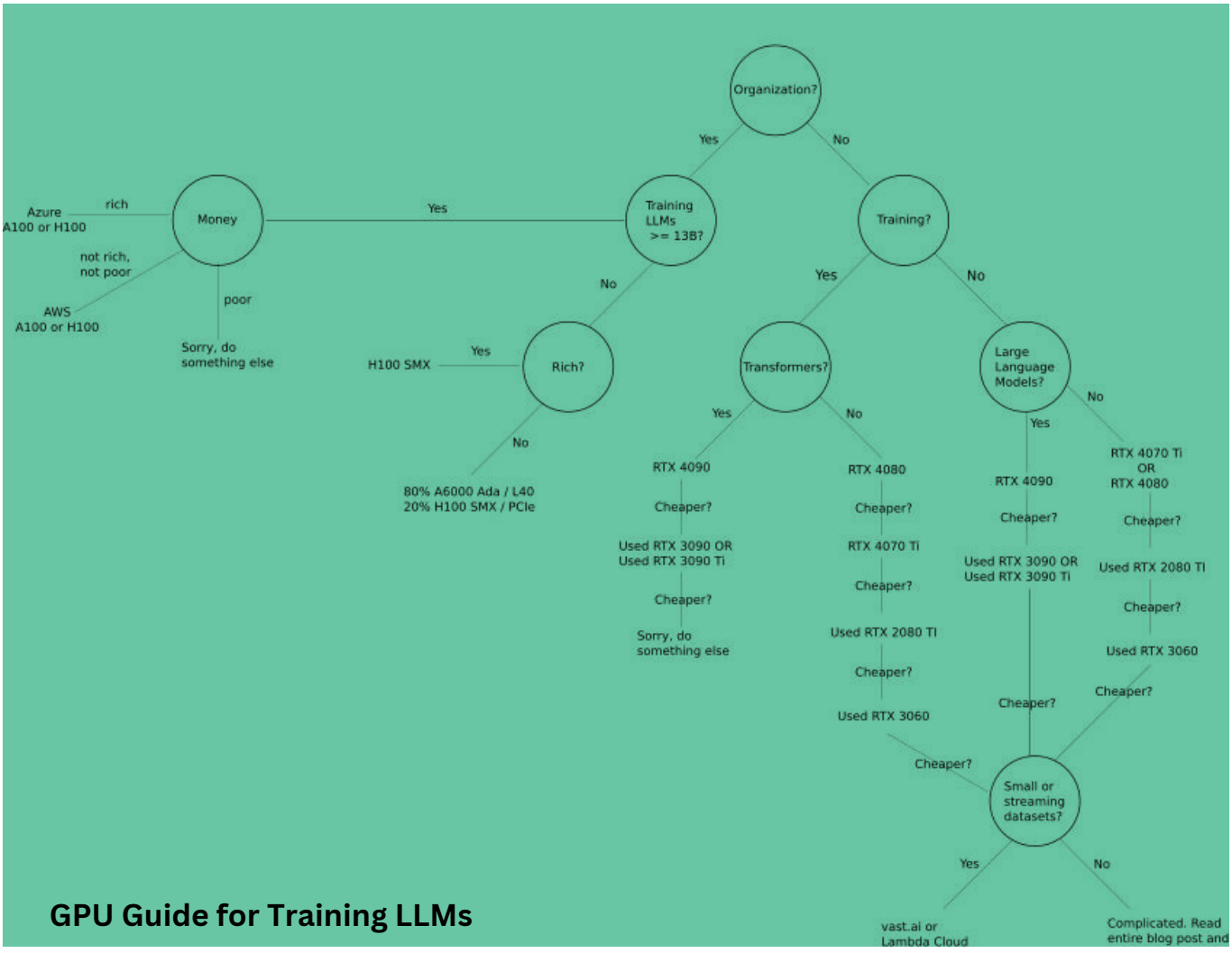
Trends Driving Growth in the Language Model Fine-Tuning Market:

1. **Growing Demand for LLM-Powered Applications**: The surge in demand for applications powered by Large Language Models (LLMs) is a pivotal trend propelling the growth of this category. Industries spanning customer service, marketing, education, and more are harnessing the capabilities of LLMs to enhance their operations. The adaptability of these models to diverse tasks is a driving force, compelling companies to fine-tune LLMs for specific functions, thereby maximizing their utility and efficiency.
2. **Advancements in Fine-Tuning Techniques**: Researchers are actively developing innovative fine-tuning techniques tailored to the nuances of LLMs. These techniques cater to the intricacies of tasks, leading to heightened performance levels. Such advancements are streamlining the fine-tuning process, rendering it quicker and more efficient. Consequently, businesses are increasingly seeking fine-tuning services to leverage the latest methodologies and unlock the full potential of LLMs.



Challenges Faced by the Market Segment:

1. **Cost of Labeled Data:** The considerable need for labeled data poses a significant challenge in the fine-tuning of large language models. Acquiring sufficient labeled data to train these models can be an expensive endeavor. The costs associated with data collection, annotation, and curation can be prohibitive for certain businesses, hindering their ability to engage in effective fine-tuning.
2. **Time-Intensive Fine-Tuning Process:** The process of fine-tuning large language models, despite access to robust computational resources, can be time-consuming. This temporal aspect presents challenges to businesses seeking to swiftly adapt their models to evolving market conditions. The prolonged fine-tuning process can potentially limit a company's agility in responding to dynamic demands.
3. **Risk of Overfitting:** One pertinent challenge in fine-tuning is the risk of overfitting. As the model becomes excessively tailored to the training data, its capacity to generalize to new, unseen data might be compromised. This overfitting phenomenon, if not managed effectively, can lead to subpar performance when the model encounters data beyond its training scope.
4. **Lack of Expertise:** The intricate nature of fine-tuning large language models necessitates specialized expertise in machine learning and natural language processing. Securing professionals with the requisite skills can be a challenging endeavor, both in terms of identification and affordability. This scarcity of expertise can impact a company's ability to fine-tune models optimally.



GPU Guide for Training LLMs

Market Size:

The current landscape of the market indicates that it is relatively smaller compared to previous periods but it is likely to grow exponentially as the adoption and compliance catch-up. The reason for this is the substantial expense associated with training customized language models from scratch. The cost and resources required for such endeavors have led to a preference for leveraging existing models and fine-tuning them to specific needs.

Approaches of Key Players:

Prominent players such as OpenAI, Google, Microsoft, Nvidia, Databricks, Meta, Cohere, Anthropic, and Hugging Face are shaping the market with their distinctive strategies. These players are amalgamating various models into unified services, capitalizing on their extensive databases garnered from their other products and services. This integration approach enables them to provide comprehensive solutions that address a wide spectrum of applications.

Winning Approaches in the Industry:

Determining winning approaches in the language model fine-tuning industry hinges on a few key factors. Models that successfully strike a balance between generalization and specificity, while also demonstrating adaptability to diverse tasks, are likely to excel. Additionally, solutions that are well-versed in mitigating biases, ensuring ethical behavior, and delivering reliable, contextually relevant outputs will stand out.

Key Diligence Questions for Evaluating AI Model Quality:

When evaluating the quality of an AI model, several diligence questions come to the forefront. Some pertinent queries include:

- How diverse and representative is the training data?
- Has the model been tested on a range of scenarios to gauge its robustness?
- What measures have been taken to address biases in the model's responses?
- How does the model handle out-of-distribution inputs?
- Is the model's decision-making process transparent and interpretable?

Data Access Strategy:

Acquiring access to data is a pivotal aspect of AI model development. Key considerations include:

- Partnerships and collaborations with data providers.
- Data augmentation techniques to diversify the training dataset.
- Exploring publicly available datasets that align with the model's objectives.
- Ensuring compliance with data privacy regulations and ethical considerations.

Evaluation: The Missing Piece

The development and deployment of ML solutions requires new types of testing and evaluation not present in traditional software development.

In particular, machine learning models introduce stochastic and statistical behaviors that need to be tested in aggregate across datasets, as well as on specific atomic pieces of data that can help validate base functionality. This process is referred to as 'evaluation'.

Why is evaluation hard for language models?
There are a number of reasons why evaluation is hard for generative language models. Some of the most common reasons include:

1. Human language is complex: Human language is very complex and can be difficult to quantify. This makes it difficult to develop evaluation metrics that accurately measure the quality of language model outputs.
2. Language models are trained on large datasets: Language models are typically trained on large datasets of text. This makes it difficult to find a representative sample of text to use for evaluation.
3. Language models can be biased: Language models can exhibit bias in line with the datasets they are trained on. This can lead to language models generating text that is not deemed acceptable against some social, ethical or legal norms.
4. Language models can be difficult to interpret: Language models are very complex and it can be difficult to interpret why they generate particular outputs. This can lead to challenges around reproducibility and consistent experimental design.

In the realm of evaluation benchmarks, two distinct categories emerge: model-focused and business-focused. These paradigms cater to different aspects of assessment, with model-focused benchmarks concentrating on the intrinsic capabilities of the language model itself, while business-focused benchmarks prioritize alignment with specific practical applications and desired outcomes. Each category offers unique insights into the performance and suitability of language models for various contexts, contributing to a well-rounded understanding of their effectiveness.

Benchmark	Focus	Domain	Evaluation Criteria
SOCKET (Choi et al., 2023)	Social knowledge	Specific downstream task	Social language understanding
MME (Fu et al., 2023a)	Multimodal LLMs	General language task	Ability of perception and cognition
Xiezhi (Gu et al., 2023)	Comprehensive domain knowledge	General language task	Overall performance across multiple benchmarks
CUAD (Hendrycks et al., 2021b)	Legal contract review	Specific downstream task	Legal contract understanding
TRUSTGPT (Huang et al., 2023c)	Ethic	Specific downstream task	Toxicity, bias, and value-alignment
MMLU (Hendrycks et al., 2020b)	Text models	General language task	Multitask accuracy
MATH (Hendrycks et al., 2021c)	Mathematical problem	Specific downstream task	Mathematical ability
APPS (Hendrycks et al., 2021a)	Coding challenge competence	Specific downstream task	Code generation ability
C-Eval (Huang et al., 2023b)	Chinese evaluation	General language task	52 Exams in a Chinese context
OpenLLM (HuggingFace, 2023)	Chatbots	General language task	Leaderboard rankings
DynaBench (Kiela et al., 2021)	Dynamic evaluation	General language task	NLI, QA, sentiment, and hate speech
Chatbot Arena (LMSYS, 2023)	Chat assistants	General language task	Crowdsourcing and Elo rating system
AlpacaEval (Li et al., 2023d)	Automated evaluation	General language task	Metrics, robustness, and diversity
HELM (Liang et al., 2022)	Transparency of language models	General language task	Multi-metric
API-Bank (Li et al., 2023a)	Tool utilization	Specific downstream task	API call, retrieval, and planning
M3KE (Liu et al., 2023a)	Multi-task	General language task	Multi-task accuracy
ARB (Sawada et al., 2023)	Advanced reasoning ability	Specific downstream task	Multidomain advanced reasoning ability
Big-Bench (Srivastava et al., 2022)	Capabilities and limitations of LMs	General language task	Model performance and calibration
MultiMedQA (Singhal et al., 2022)	Medical QA	Specific downstream task	Model performance, medical knowledge, and reasoning ability
CVALUES (Xu et al., 2023b)	Safety and responsibility	Specific downstream task	Alignment ability of LLMs
ToolBench (ToolBench, 2023)	Software tools	Specific downstream task	Execution success rate
PandaLM (Wang et al., 2023h)	Instruction tuning	General language task	Winrate judged by PandaLM
GLUE-X (Yang et al., 2022)	OOD robustness for NLU tasks	General language task	OOD robustness
KoLA (Yu et al., 2023)	Knowledge-oriented evaluation	General language task	Self-contrast metrics
AGIEval (Zhong et al., 2023)	Human-centered foundational models	General language task	General
PromptBench (Zhu et al., 2023)	Adversarial prompt resilience	General language task	Adversarial robustness
MT-Bench (Zheng et al., 2023)	Multi-turn conversation	General language task	Winrate judged by GPT-4
M3Exam (Zhang et al., 2023c)	Human exams	Specific downstream task	Task-specific metrics
GAOKAO-Bench (Zhang et al., 2023e)	Chinese Gaokao examination	Specific downstream task	Accuracy and scoring rate

Model Focused Benchmarks, arXiv:2307.03109

Evaluating LLMs

When assessing the performance of a Language Model (LM) for a business, a systematic evaluation strategy becomes crucial. There exists a range of methodologies for evaluating LM performance, each offering distinct advantages and drawbacks.

Business Focused Benchmarks

GLUE Benchmark
The General Language Understanding Evaluation (GLUE) benchmark comprises nine diverse natural language understanding tasks, serving as a comprehensive yardstick for assessing various LM models. The GLUE benchmark proves effective in evaluating LMs designed for versatile applications.

Task-Specific Downstream Evaluation
Alternatively, it might be more pertinent to appraise an LM's efficacy based on its designated task. For instance, an LM tailored for text classification can be assessed through conventional classification metrics like precision, recall, and F1 score.

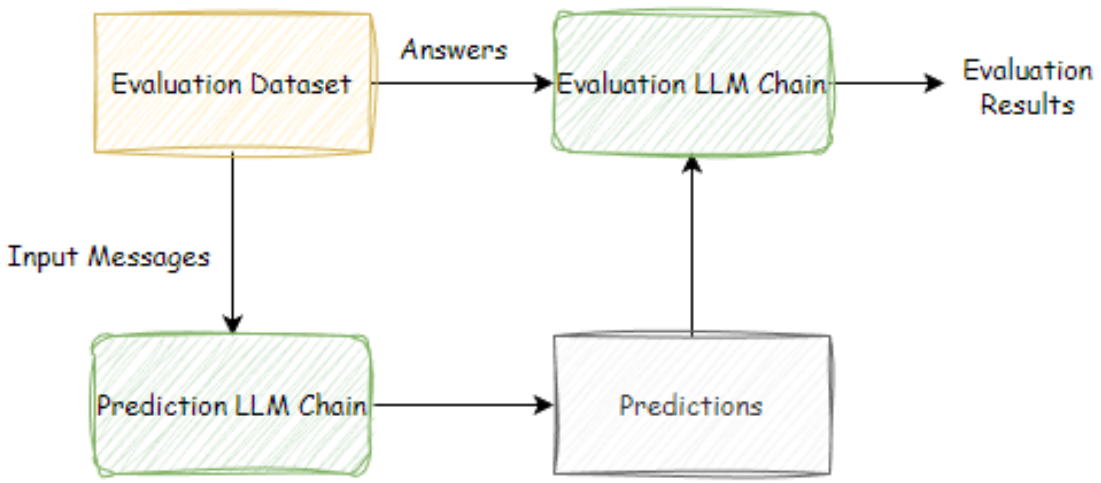
Perplexity
Perplexity serves as a statistical gauge of an LM's text prediction confidence. Lower perplexity values reflect adept test set prediction, while higher values denote inadequate prediction. This metric is pertinent for evaluating LMs designed for text generation and machine translation tasks.

BLEU Score
The BLEU (bilingual evaluation understudy) score, ranging from 0 to 1, gauges the quality of machine translation compared to a reference translation. Higher BLEU scores indicate greater similarity to the reference text. This metric holds significance in evaluating LMs geared towards machine translation tasks.

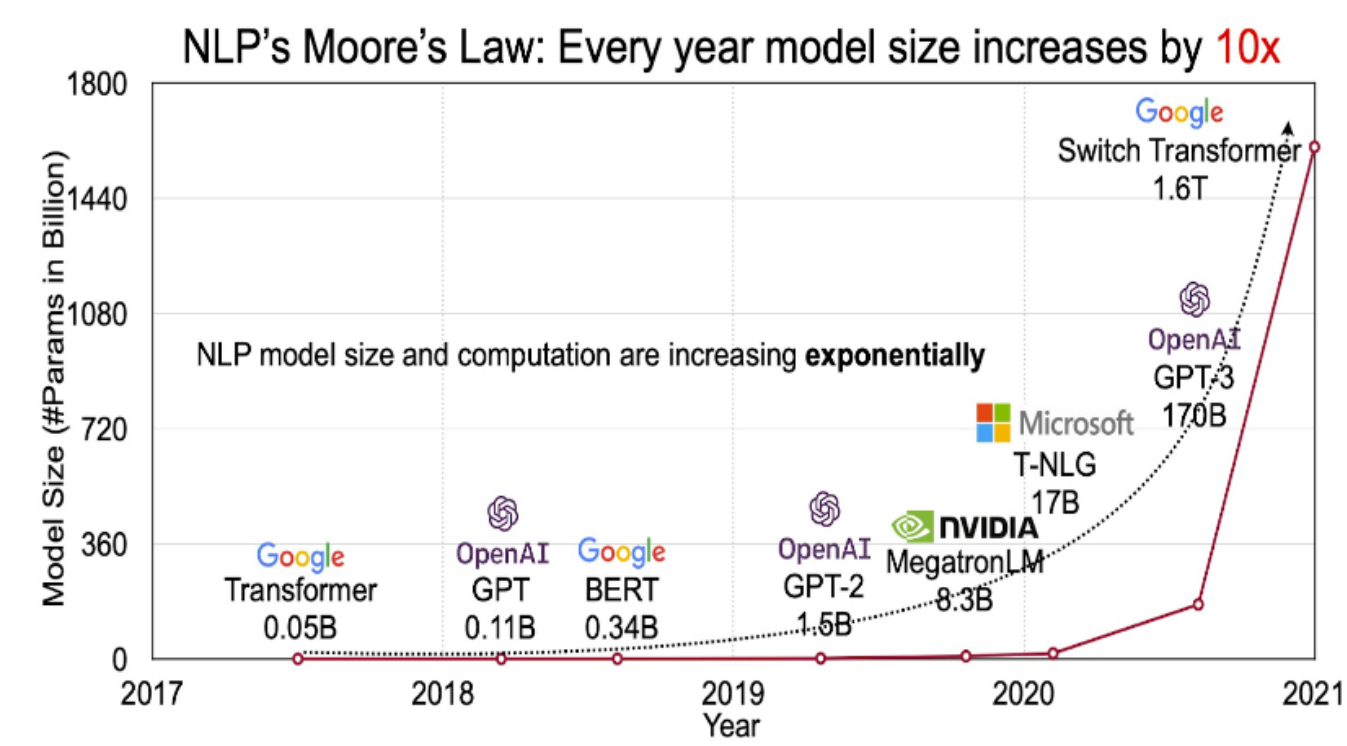
Human Evaluation

In conjunction with statistical and automated evaluation methodologies, human evaluators play a pivotal role in assessing LMs' attributes such as creativity, humor, and toxicity. Their insights offer valuable feedback on the quality of LM-generated content.

The selection of the most suitable evaluation approach for an LM hinges on its designated application. However, the aforementioned methodologies present a strong foundational framework for LM evaluation.



Not just hype: LLMs are here to stay



Challenges associated with LLMs

- LLMs are trained on massive datasets of text, which can contain biases and harmful content. It is important to be aware of these biases and take steps to mitigate them.
- LLMs can be used to generate harmful content, such as hate speech or propaganda. It is important to use LLMs responsibly and to be aware of the potential risks.
- LLMs are still under development, and they can be prone to errors. It is important to test LLMs carefully before using them in production.

Opportunities associated with LLMs

- LLMs have the potential to revolutionize the way we interact with computers. They can be used to create more natural and intuitive user interfaces, and they can also be used to automate tasks that are currently done by humans.
- LLMs can be used to improve our understanding of the world. They can be used to analyze data and identify patterns, and they can also be used to generate creative content.
- LLMs can be used to solve real-world problems. They can be used to develop new medical treatments, to create more effective educational tools, and to improve the efficiency of businesses.

Overall, LLMs are a powerful new technology with the potential to change the world. However, it is important to be aware of the challenges and opportunities associated with LLMs before using them.

Technical Challenges for LLM applications in enterprise

Large language models are not magic solutions that can seamlessly address all natural language processing (NLP) tasks. They do come with their own set of challenges. Here are some key challenges associated with large language models:

1. **Data Bias and Fairness:** Large language models are trained on vast amounts of data, which can introduce biases present in the training data. These biases can manifest in the generated text or predictions, potentially perpetuating societal biases and inequalities. Ensuring fairness and mitigating biases in large language models is a significant challenge that requires careful data curation, evaluation, and ongoing monitoring.
2. **Ethical Considerations:** Large language models raise ethical concerns related to the generation of potentially harmful or misleading content. There have been instances of models producing offensive, biased, or false information. Ensuring responsible and ethical use of large language models is crucial, which involves implementing safeguards, transparency measures, and guidelines to prevent misuse.
3. **Privacy and Security:** The key reason LLMs leak data is memorization. However, this still remains an unsolved problem and was the key reason behind Samsung's ban on all generative applications in April 2023.
4. **Computational Resources and Efficiency:** Training and deploying large language models require substantial computational resources, both in terms of processing power and memory. This can be a challenge for organizations with limited resources or when deploying models in resource-constrained environments. Optimizing the computational efficiency of these models is an ongoing area of research to make them more accessible and practical. There are two kind of costs involved in training LLMs - the obvious costs and the hidden costs. These costs have been covered extensively in this survey paper. On GPU Selection Tim Dettmers has put together this guide to GPU selection for training LLMs.
5. **Interpretability and Explainability:** Large language models are often referred to as "black boxes" because they lack interpretability. Understanding how these models arrive at their predictions or generate specific outputs can be challenging. This lack of interpretability can be problematic, especially in critical domains where explainability is required for accountability and decision-making.
6. **Fine-tuning and Transfer Learning:** While fine-tuning is a powerful technique to adapt large language models to specific tasks or domains, it requires access to task-specific data and careful parameter tuning. Obtaining high-quality labeled data for fine-tuning can be expensive and time-consuming. Additionally, effectively transferring knowledge from pre-trained models to new tasks remains a challenge, particularly for tasks with limited labeled data.

Acknowledging and addressing these challenges is essential for responsible and effective utilization of large language models. Researchers, developers, and policymakers are actively working towards developing solutions and guidelines to mitigate these challenges and foster the responsible and beneficial use of large language models in real-world applications.

Conclusion

As LLMs continue to develop, we can expect to see even more innovative and creative applications of this technology.

#1 LLMs: The Next Frontier in Conversational Technology

Some of the specific benefits of using LLMs in audio speech synthesis and voice technology:

- **Improved realism and naturalness:** LLMs can generate speech that is more realistic and natural-sounding than traditional speech synthesis methods. This is because LLMs are able to learn the statistical relationships between words and sounds, which allows them to produce speech that is more accurate and consistent.
- **Increased flexibility:** LLMs can be used to create custom voices that are tailored to specific applications. This means that businesses can create voices that are both informative and engaging, and that are designed to meet the specific needs of their customers.
- **Reduced development time and cost:** LLMs can be used to automate many of the tasks involved in audio speech synthesis, which can save businesses time and money.

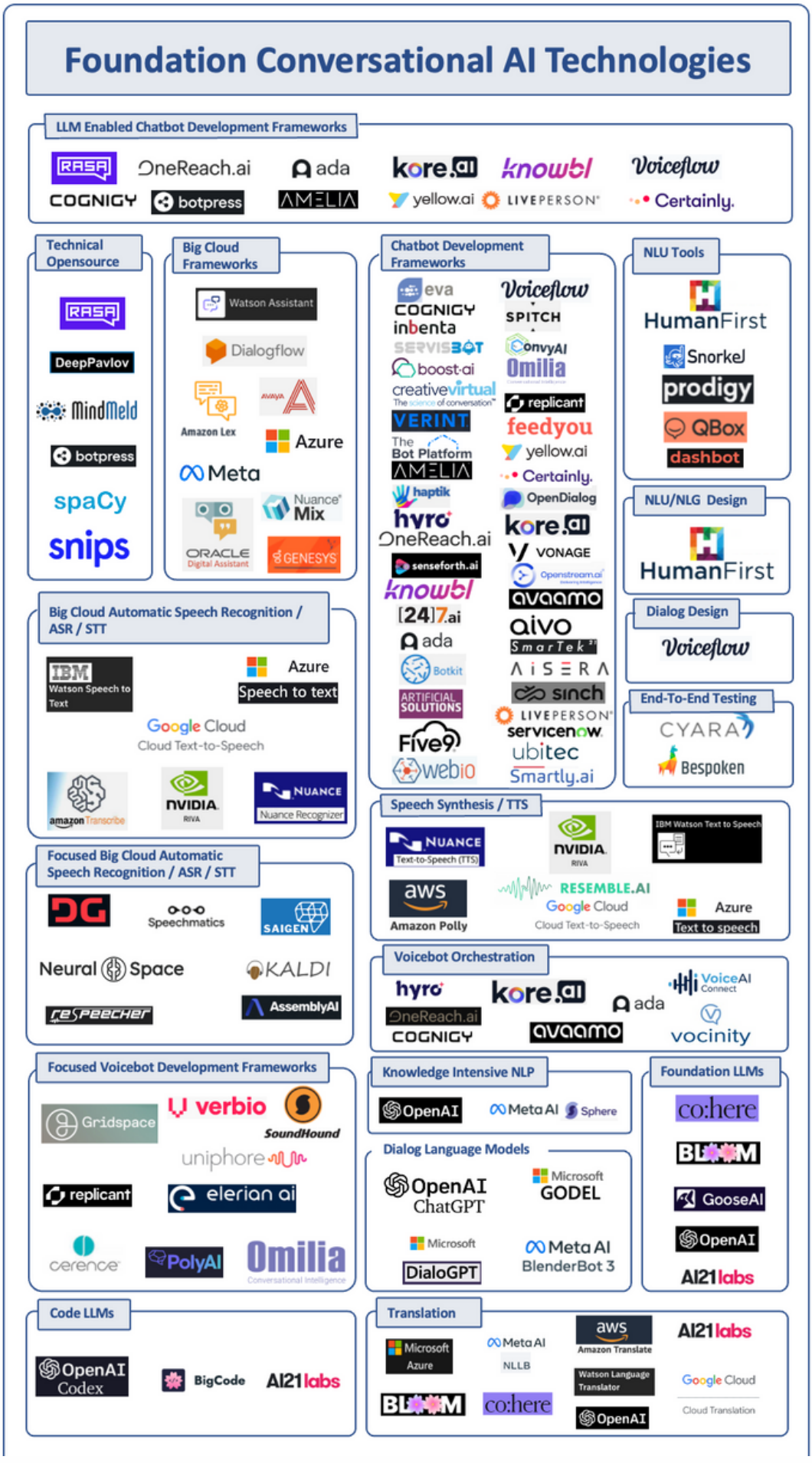
Key Industries taking advantage of LLMs in conversational technology

- **Virtual assistants:** **LLMs are being used to create virtual assistants that can provide information and complete tasks for users. For example, Amazon's Alexa and Apple's Siri are both powered by LLMs.**
- **Call centers:** **LLMs are being used to create chatbots that can answer customer questions and resolve issues. This can help businesses to reduce the cost of providing customer support.**
- **Education:** **LLMs are being used to create educational tools that can help students learn new languages and concepts. For example, Google's AIY Project includes a voice-activated language learning app that uses an LLM to generate realistic and natural-sounding speech.**

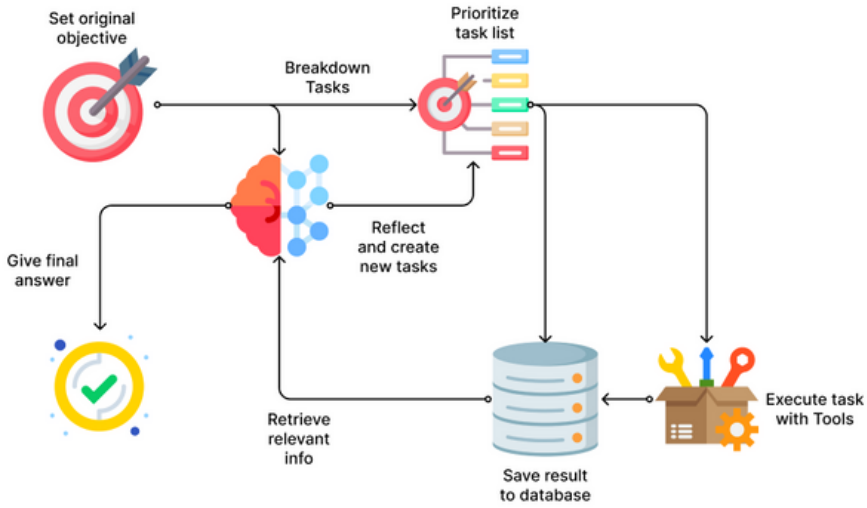
The big challenges in voice technology beyond just language models:

- Background noise: This is one of the biggest challenges in voice recognition technology. Even the best language models can struggle to understand speech when there is a lot of background noise present. This can be a problem in noisy environments, such as busy streets or crowded rooms.
- Multi-speaker environments: Voice recognition systems also need to be able to handle multi-speaker environments. This is when multiple people are speaking at the same time, and the system needs to be able to distinguish between different voices and understand what each person is saying.
- Accents and dialects: Another challenge is dealing with accents and dialects. Language models are typically trained on a specific set of accents and dialects, and they may not be able to understand speech that is spoken with a different accent or dialect.
- Grammar and syntax: Voice recognition systems also need to be able to understand grammar and syntax. This is important for understanding the meaning of a sentence, and it can be a challenge for language models.
- Emotion: Voice recognition systems are also starting to be used to recognize emotions. This is a challenging task, as emotions can be expressed in different ways through speech.

These are just some of the challenges that need to be addressed in order to improve voice recognition technology. As the technology continues to develop, we can expect to see these challenges overcome and voice recognition become even more accurate and reliable.


Foundation Conversational AI Technologies
LLM Enabled Chatbot Development Frameworks
RASA
OneReach.ai
ada
kore.ai
knowbl
Voiceflow
COGNIGY
botpress
AMELIA
yellow.ai
LIVEPERSON
Certainly.
Technical Opensource
RASA
DeepPavlov
MindMeld
botpress
spaCy
snips
Big Cloud Frameworks
Watson Assistant
Dialogflow
Amazon Lex
Azure
Meta
Nuance Mix
ORACLE Digital Assistant
GENESYS
Chatbot Development Frameworks
eva
COGNIGY
inbenta
SERVISBOT
boost·ai
creativevirtual
VERINT
The Bot Platform
AMELIA
haptik
hyro
OneReach.ai
senseforth.ai
knowbl
[24]7.ai
ada
Botkit
ARTIFICIAL SOLUTIONS
Five9
webio
Voiceflow
SPITCH
onvyAI
Omilia
replicant
feedyou
yellow.ai
Certainly.
OpenDialog
kore.ai
VONAGE
Openstream.ai
avaamo
aivo
SmarTek
AISERA
sinch
LIVEPERSON
servicenow
ubitec
Smartly.ai
NLU Tools
HumanFirst
Snorkel
prodigy
QBox
dashbot
NLU/NLG Design
HumanFirst
Dialog Design
Voiceflow
End-To-End Testing
CYARA
Bespoken
Big Cloud Automatic Speech Recognition / ASR / STT
IBM Watson Speech to Text
Azure Speech to text
Google Cloud Cloud Text-to-Speech
amazon Transcribe
NVIDIA RIVA
NUANCE Nuance Recognizer
Speech Synthesis / TTS
NUANCE Text-to-Speech (TTS)
NVIDIA RIVA
IBM Watson Text to Speech
aws Amazon Polly
RESEMBLE.AI
Google Cloud Cloud Text-to-Speech
Azure Text to speech
Focused Big Cloud Automatic Speech Recognition / ASR / STT
DG
Speechmatics
SAIGEN
Neural Space
KALDI
reSpeecher
AssemblyAI
Voicebot Orchestration
hyro
OneReach.ai
COGNIGY
kore.ai
avaamo
ada
VoiceAI Connect
vocinity
Focused Voicebot Development Frameworks
Gridspace
verbio
SoundHound
uniphore
replicant
elerian ai
cerence
PolyAI
Omilia Conversational Intelligence
Knowledge Intensive NLP
OpenAI
Meta AI
Sphere
Foundation LLMs
co:here
BLOOM
GooseAI
OpenAI
AI21labs
Dialog Language Models
OpenAI ChatGPT
Microsoft GODEL
Microsoft DialoGPT
Meta AI BlenderBot 3
Code LLMs
OpenAI Codex
BigCode
AI21labs
Translation
Microsoft Azure
Meta AI NLLB
aws Amazon Translate
AI21labs
Watson Language Translator
Google Cloud Cloud Translation
BLOOM
co:here
OpenAI

2 LLMs: The Missing Piece for Autonomous Agents



LLMs (Large Language Models) are now also seeing adoption for autonomous agents because they have the following capabilities:

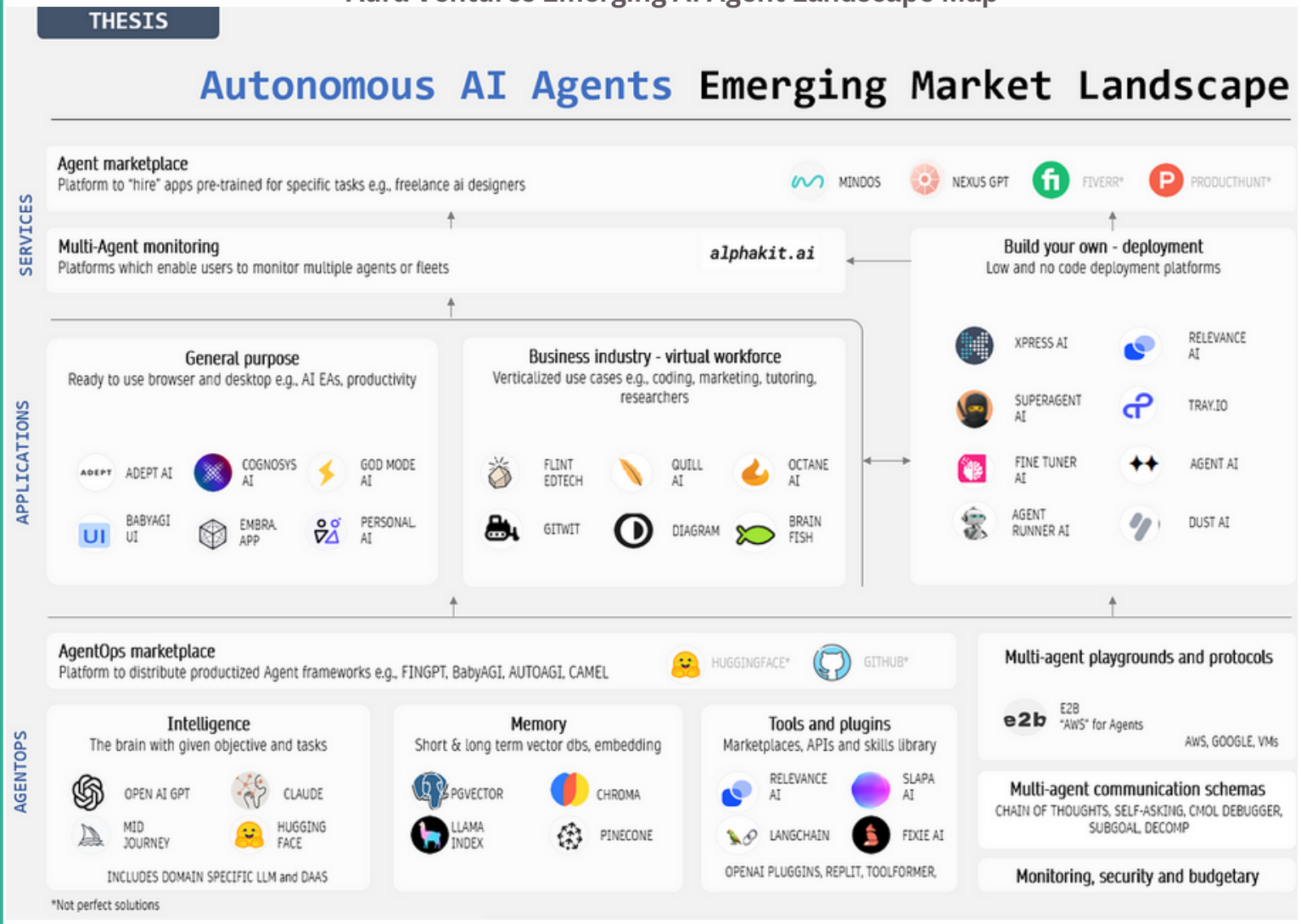
- **Natural language understanding:** LLMs can understand and process natural language, which is essential for interacting with humans and the environment.
- **Planning and decision-making:** LLMs can plan and make decisions based on the information they have gathered. This allows them to act autonomously and complete tasks without human intervention.
- **Learning and adaptation:** LLMs can learn and adapt to new information and situations. This allows them to improve their performance over time.
- **Communication:** LLMs can communicate with humans and other agents using natural language. This allows them to coordinate their actions and work together to achieve common goals.

Here are some specific examples of how LLMs are being used for autonomous agents:

- **Self-driving cars:** LLMs are being used to help self-driving cars navigate the road, avoid obstacles, and make decisions about how to drive safely.
- **Virtual assistants:** LLMs are being used to power virtual assistants like Amazon Alexa and Google Assistant. These assistants can understand natural language commands and complete tasks for users, such as setting alarms, playing music, and making reservations.
- **Chatbots:** LLMs are being used to power chatbots that can interact with customers in a natural and engaging way. These chatbots can answer questions, provide support, and even sell products.
- **Robotic assistants:** LLMs are being used to develop robotic assistants that can help people with tasks around the house or in the workplace. These assistants can understand natural language commands and interact with the environment in a safe and efficient way.

As LLMs continue to develop, they will become even more powerful and capable. This will open up new possibilities for using them in autonomous agents to solve a wide range of problems.

Aura Ventures Emerging AI Agent Landscape Map



Wrapping Up,

LLMs are a powerful new technology with the potential to revolutionize many industries. They are already being used for a variety of tasks, such as translation, writing, and question answering. As LLMs continue to develop, they will become even more powerful and capable. This will open up new possibilities for their use in a wide range of applications.

However, there are also some challenges and limitations associated with LLMs. One challenge is that they can be biased, reflecting the biases of the data they are trained on. Another challenge is that they can be fooled by adversarial examples, which are intentionally crafted inputs that can cause them to make mistakes.

Despite these challenges, LLMs are a promising technology with the potential to make a significant impact on the world. **As they continue to develop, their impact and penetration for AI-powered applications will be unmatched.**

2023

LARGE LANGUAGE MODELS

Prepared By :
Abi Aryan

Supervised By :
Reena Jailwala

LACONIA CAPITAL
381 Park Ave S,
New York, NY 10016