Konverge.AI

LARGE LANGUAGE MODELS


ABC
ABC
ABC

LARGE LANGUAGE MODELS

A Bite Sized Guide on LLMs for Managers

Preface

Ever since Chat-GPT was released to the general public late last year, the interest in Large Language Models has skyrocketed. Having seen the relatively high accuracy and ease of use of Chat-GPT, everyone seems convinced about the potential of Large Language Models.

The pace at which the technology has evolved over the last 6-8 months has been breathtaking. A lot of managers we spoke to want to develop an in-depth understanding of this emerging field so that they can make informed decisions.

With the intention to provide managers a simplified, bite-sized guide to Large Language Models, we have created this book.

In this book, you will find our learnings from the field and our experiments on LLM, condensed into a short 40 page book which will help you develop a foundational understanding of this field from different perspectives.

All you need to invest is a couple of hours of your time to read this book and you will have a deeper understanding of LLMs, their capabilities, what is happening in businesses around LLMs and how you can infuse LLMs into your business.

We hope that you find this book useful and enjoy reading it as much as we enjoyed working on it.

TABLE OF CONTENTS


LANGUAGE
EMBEDDINGS
LARGE
CHAT-GPT
VECTORS
SPEECH
ACCENT
EXPRESSION CONVERSATION
NOISE
TEXT
JARGON
LEXICON
TALK
DICTION
PALAVER
INTERCHANGE
SLANG
HYPER PARAMETER
MODEL
ENCODER
DECODER
PHRASEOLOGY
PARLANCE
VOICE
TERMINOLOGY
TUNING
GENERATIVE
CONVERSATION
WORD
WORD
SPEECH
EXPRESSION
TRANSFORMER
TRAINING
UTTERANCE
DICTION
TALK
PHRASEOLOGY
EXPRESSION
ACCENT
SPEECH
SOUND
ARTICULATION

01

Introduction to LLMs

In this section, we provide a collection of foundational articles on Large Language Models (LLMs). We begin with the fundamental concepts and move on to the evolution of LLMs. We then delve into their operational mechanics without getting into too many technical details. The article on fundamentals of LLMs is aimed at helping you understand LLMs sans all the complicated technical concepts. We also take a look at LLM related jargons. We conclude this section with the impact of LLMs on various industries. After reading this section, you will have a solid understanding of LLMs, their functionality, and potential applications.

What are Large Language Models?

Ever since the release of Chat-GPT, everyone is talking (or should we say chatting?) about Chat-GPT. All of us have heard about Chat-GPT, Bard, GPT-4, LaMDA and other "Large Language Models" (LLMs). It seems like something magical that answers all your questions – most of them correctly.

Large Language Models (LLMs) are a type of artificial intelligence model designed to process, understand, and generate human-like text.

In technical terms, Large Language Models are complex neural networks that are trained on massive amount of text data. They use this data to learn patterns in language and generate text that is similar to what a human would write.

LLMs are being used for a wide range of tasks, such as language translation, text summarization, and generating content.

Natural Language Processing (NLP)

There is an entire branch of computer science on Natural Language Processing (NLP) that focuses on human and computer interaction in natural (not coding) language. Language Modelling is a subfield of NLP which deals with generation of words based on the context of the other words in the sentence.

Language Models

There are two kinds of Language Models –

1. Autoregressive models which predict the next word in a sequence of words based on the patterns it has learned.

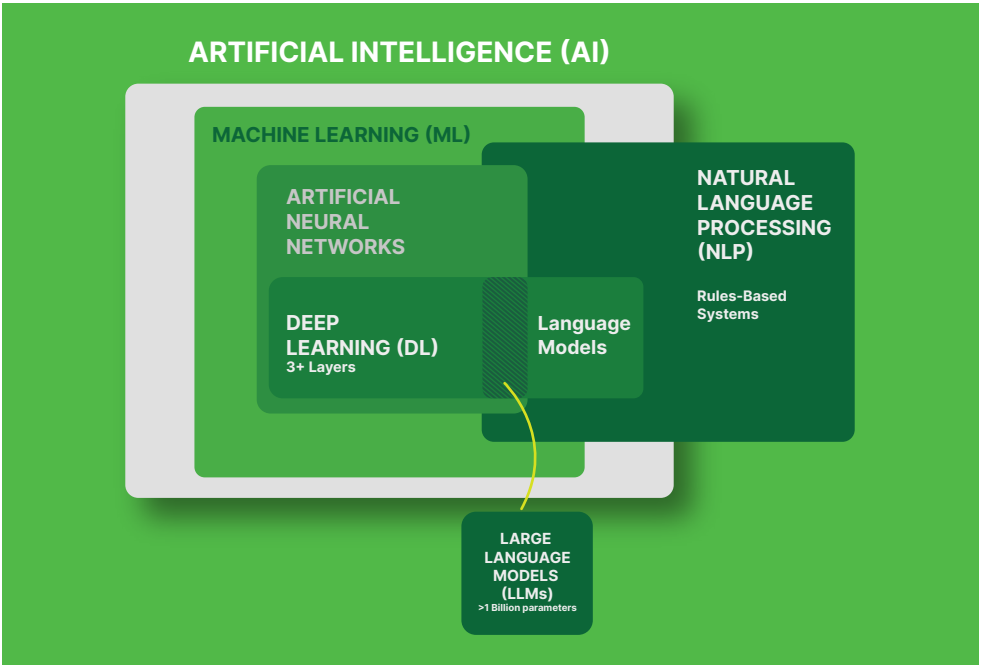
2. Autoencoding models which predict the missing or masked word based on other words in the text.

LLMs that generate responses to text queries ("prompts") use Autoregressive Models to predict what comes next in a piece of text, given what they've seen so far.

Language Models use Neural Networks (NN) which are modeled on human brains to get trained to understand the patterns in language. Neural networks are structured in a layered fashion. Neural networks consist of layers of nodes, or "neurons", which are interconnected. Each node takes in input, performs a calculation, and then passes that output on to the next layer. The final layer of nodes provides the model's prediction.

The "learning" part of a neural network happens during training, when it adjusts the weights and biases of the connections between these nodes to minimize the difference between the model's predictions and the actual data. These weights and biases between neurons are called parameters. A Large Language Model has billions of parameters. Autoregressive models predict the next word in a sequence of words based on these parameters.

We can think of it this way – a Large Language Model uses 100 billion equations to predict which words belong at which position in a sentence. So when Chat-GPT generates a text response to your query (prompt), it is just predicting which word will come next in the context of your input prompt. Technically it is not generating anything new, it is just using its knowledge to come up with a string of words that have the best probability of occurring together.



Common LLMs

Chat-GPT:
The renowned language model powering conversations and text generation.

Vicuna 13B:
A powerful LLM designed to handle complex language tasks with its 13 billion parameters.

Bloom:
An innovative LLM that blooms with creative and insightful responses, making language generation a breeze.

Databricks Dolly:
LLM trained on the Databricks machine learning platform. Read more about it on the next page.

Open Source vs Closed Source

Open Source	Closed Source
Available for free can be freely modified.	Requires License or Subscription.
Source code is open and can be modified or built upon. Can be trained with full flexibility.	Source code is not openly available and can not be modified. Limited training flexibility.
Greater transparency in inner workings.	Inner workings and parameters are not transparent.
User owns the IP	LLM provider owns the IP
Examples: • LLaMa from Meta • Alpaca (Stanford) • Vicuna • Databricks Dolly • Koala	Examples: • GPT-3/GPT-4 from Open AI • Bard from Google • Bloomberg GPT • LaMDA from Google • Kosmos – I from Microsoft • Chincilla from Deepmind
Availability is defined by your infrastructure.	Availability is constrained by availability of the LLM infrastructure.
Training costs and operating costs are lower.	Operating costs are higher.

Most open source LLMs have a non-commercial license which restricts the commercial use of these LLMs. However, these LLMs can be used for research and internal projects where the organization does not make any commercial gains. For example, organization's internal knowledge sharing system can be converted into a question-answer system using an open source LLM.

Databricks Dolly

Databricks Dolly 2.0 is the first open source, instruction-following LLM, fine-tuned on a human-generated instruction dataset, licensed for research and commercial use.

Dolly 2.0 is a 12B parameter language model based on the EleutherAI pythia model family and fine-tuned exclusively on a new, high quality human generated instruction following dataset.

Databricks has open sourced the entirety of Dolly 2.0, including the training code, the dataset, and the model weights, all suitable for commercial use. This means that any organization can create, own, and customize powerful LLMs that can talk to people, without paying for API access or sharing data with third parties.

Konverge.AI is a Databricks partner and has developed accelerator based on Dolly.

Neural Language Models

The first neural language model, a feed-forward neural network was proposed in 2001 by Joshua Bengio and team.

Encoding semantic meaning with Word2vec

Tomas Mikolov publishes the word2vec algorithm which uses a neural network model to learn word associations from a large corpus of text and produce word embeddings

Seq2seq + Attention

Seq2Seq model with an attention mechanism consisting of an encoder, decoder, and attention layer is proposed.

Present Transformers + Large Language Models

Ashish Vaswani and team publish a pathbreaking article "Attention is all you need" introducing the transformer architecture

BERT Google

Google launches BERT (Bidirectional Encoder Representations from Transformers) which showed an unmatched ability to understand the context from both directions.

GPT-2 Open AI

Open AI's GPT-2 with 10B tokens intrigues the AI community with its ability to create coherent, context-relevant text

GPT-3 Open AI

GPT-3 launched with its colossal 175 billion parameters. GPT-3 outshone its predecessors, demonstrating an ability to generate human-like text and perform diverse tasks.

Megatron-Turning NLG Microsoft & NVIDIA

The 530 B parameter Model demonstrates unmatched accuracy in a broad set of natural language tasks such as, Completion prediction, Reading comprehension, Natural language inferences, Word sense disambiguation etc.

Chat-GPT

Chat-GPT takes the world by storm, Google launches LaMDA Language Model for Dialogue Applications".

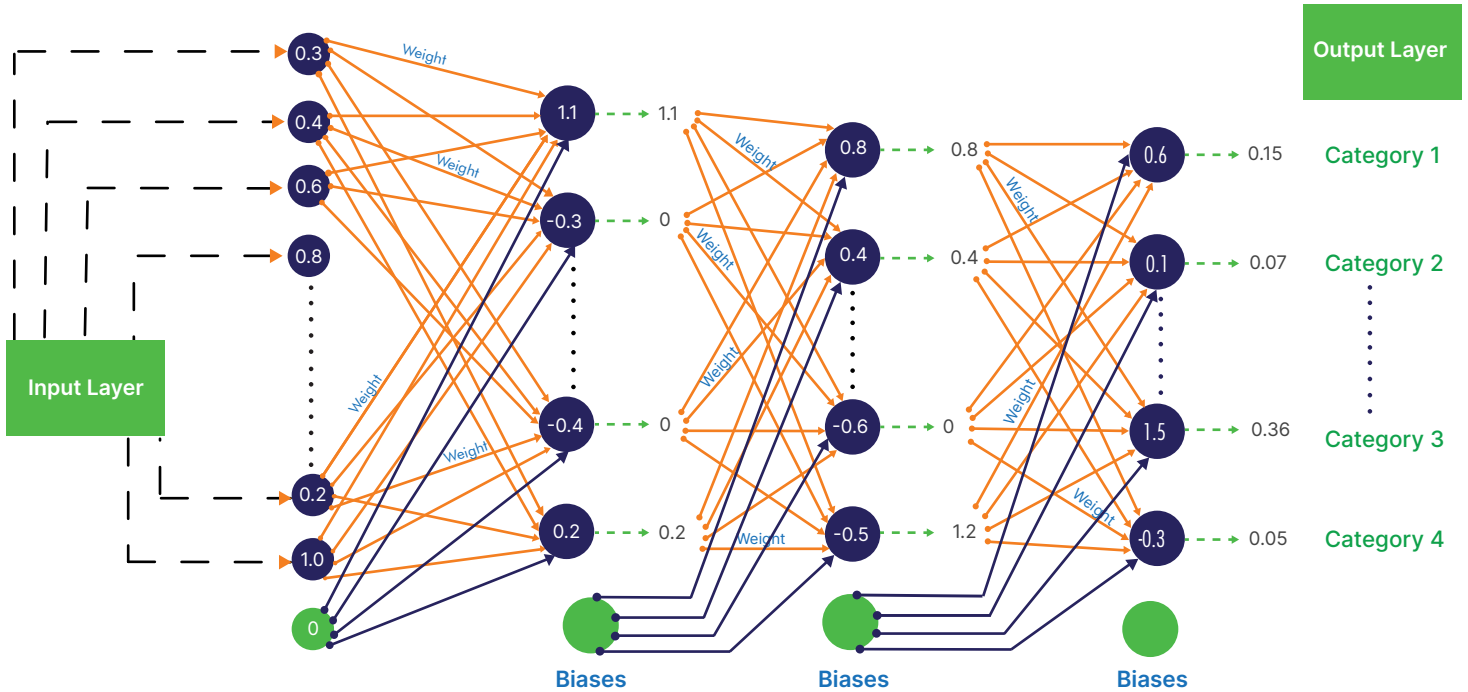
LLaMA Meta, Open AI GPT-4, Bloomberg BloombergGPT

Following on heels of Chat-GPT a multitude of open source and domain specific LLMs are launched.

???

Autogenerated LLM powered e-book ??? Who knows? We might still be required to write another e-book.😅

Neural Network with Weights and Balances



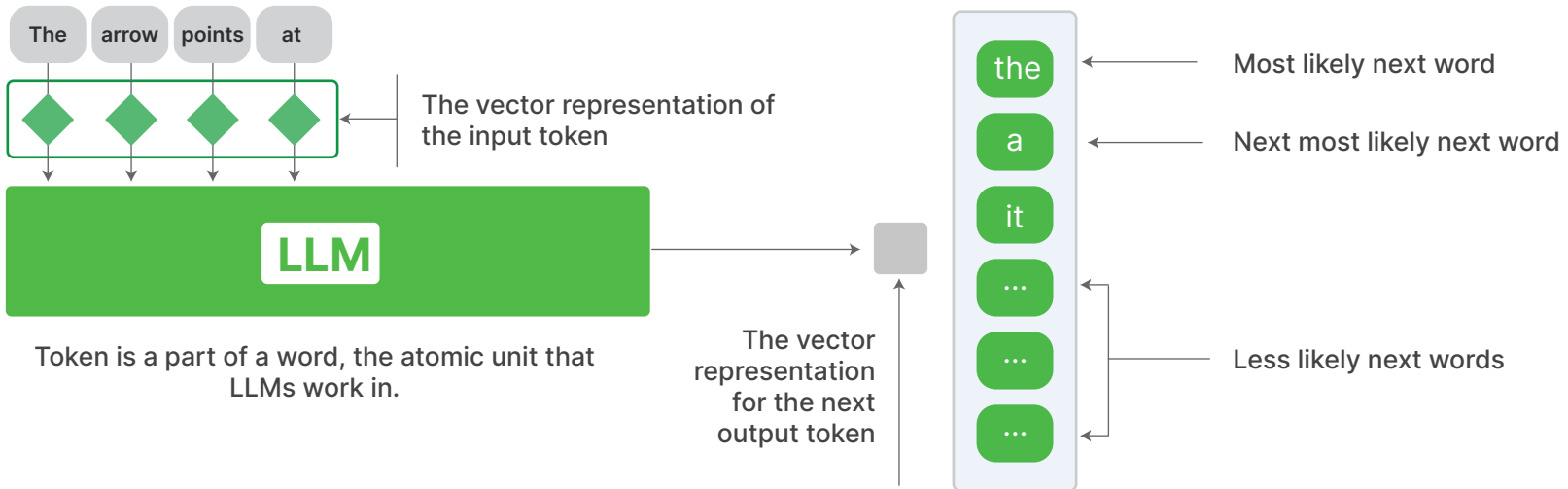
A basic neural network has interconnected artificial neurons in three layers:

Input Layer
Information enters the artificial neural network from the input layer. Input nodes process the data and pass it on to the next layer.

Hidden Layer
Hidden layers take their input from the input layer or other hidden layers. Each hidden layer analyzes the output from the previous layer, processes it further, and passes it on to the next layer.

Output Layer
The output layer gives the final result of all the data processing by the artificial neural network.

Generic language model - A next word predictor



An LLM is trained by giving it text and asking it to predict the next word.

The model compares its prediction with the actual word from a book to calculate how wrong it is. This is called the "loss". It then adjusts its weights to reduce the loss. This is called "learning".

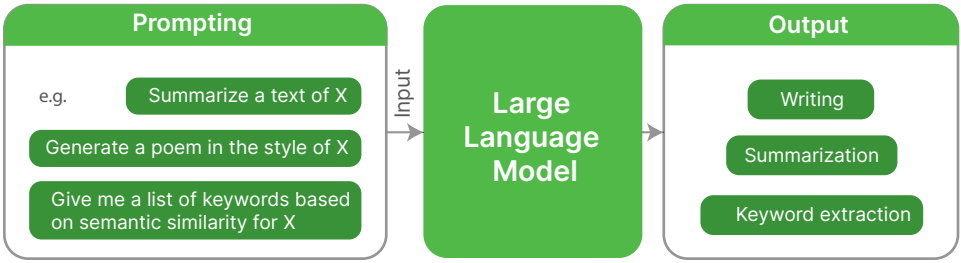
A trained model learns i.e. adjusts the weights and biases to predict the most likely next word.

Glossary of LLM terms

Keywords	Definition	Example
Large Language Model (LLM)	A computer program that learns to understand and create human language.	A model that generates realistic dialogue for virtual assistants.
Natural Language Processing (NLP)	Teaching computers to understand, interpret and create human language.	Analyzing customer reviews to determine sentiment towards a product.
Transformer	A Neural network architecture used in many LLMs, known for its attention mechanism.	Powering models like GPT-3 with parallel processing capabilities.
Encoder	A Part of a transformer that helps it understand and remember input.	Encoding a sentence to represent its meaning in a numerical form.
Decoder	A Part of a transformer that helps it generate a response or answer.	Generating a creative and coherent story based on a given prompt.
Tokenization	Breaking text into words or parts (tokens) for language processing.	Segmenting a paragraph into individual words for analysis.
Attention Mechanism	Allowing computers to focus on important input parts when generating output.	Emphasizing keywords when generating a document summary.
Beam Search	A method to find the best word sequence when generating text.	Choosing the most suitable words to generate a poem.
Data Augmentation	Expanding and diversifying a dataset by creating new samples.	Creating variations of sentences to improve machine translation.
Pre-training	The initial phase of training a language model on a large corpus of text data.	Training a language model on a vast collection of books.
Fine-tuning	A Subsequent phase of training where a pre-trained model is optimized for a specific task.	Adapting a language model for sentiment analysis in the financial domain.
Language Generation	A Process of generating human-like text or dialogue using LLMs.	Generating a news article based on the given input.
Vectorization / Embedding	Vectorization is the conversion of human-readable data into machine-readable format using numbers, enabling machines to comprehend and process the content and context of the information.	Transforming a textual document containing customer reviews into numerical vectors.

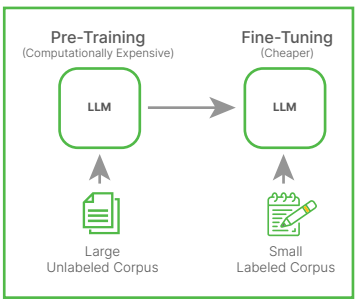
LLM Concepts

1. PROMPT & PROMPT ENGINEERING



Prompt engineering is the process of designing and refining specific text prompts to guide Large Language Models (LLMs) in generating desired outputs. It involves crafting clear and specific instructions to get the desired output.

2. PRE-TRAINING & FINE-TUNING LLM



A computer understands only numbers. When a computer is trained to learn a language, it means given a set of words, it can predict the next word with accuracy. Pre-training an LLM is feeding it a large amount of data (few billion words) with the intent of accurately predicting words in a sentence. After pre-training, LLM assigns weight to each node of neural network which decides how the next word is predicted. Pre-training process is very expensive (costs a few thousands to a million dollars) and takes a long time.

When we use Pre-trained Language Models (PLMs) for specific tasks like text summarization or question-answering we need to fine-tune them. During the fine-tuning process as well, all the model weights are updated. Dataset required for fine-tuning is much smaller so it takes lesser time and is cheaper.

3. CHAINING, FRAMEWORKS & AGENTS

LLMs can effectively perform single task at a time, however, many real-world applications involve complex or multi-step tasks that cannot be handled with a single run of an LLM. Chaining multiple LLM runs together with the output of one step being the input to the next can help LLMs perform more complex tasks. These chains also make the tasks more transparent and controllable.

Frameworks like Langchain enable developers to “chain” different components together to create more advanced use cases around LLMs.

Agents use LLMs’ outputs to decide what actions should be taken. Tools like web search or calculators are packaged into a logical chain of operations. Use of agents involves, a base LLM, a tool that it will be interacting with, an agent to control the interaction.

How have LLMs impacted industries?

LLMs can perform tasks such as text summarization, entity extraction, answering questions as a chatbot, machine translation, document classification, text generation, code generation and can even perform search.

LLMs make it possible to analyze huge amounts of unstructured data using just natural language. A well trained LLM can "read" through thousands of pages of text and come back with the insights you asked for within just a few seconds.

LLMs have the potential to revolutionize the way tedious manual tasks related to text processing are performed across industries. LLMs are helping businesses unleash the wave of hyperproductivity.

Over the last two years, LLM neural networks have been expanding AI's impact in fields such as healthcare, education, finance, software and manufacturing. However, it was Chat-GPT which really created ripples in the industry with its Generative capabilities. Let's take a look at LLM use cases that have been implemented across various industries.

LLMs in Healthcare Use Cases:

1. Questions and Answers based enquiry about medical topics. Example "How to treat rashes?"
2. Interpret medical research papers and extract important results. Example, input PubMed abstract and ask about key results.
3. Automatically generate detailed medical reports based on key inputs from the doctor.
4. Summarize long form clinical notes such as discharge summaries, visit summaries, pathology and other test reports into a short paragraph or bulleted list.

LLMs in Finance Use Cases:

1. Customer facing chat-bot for better user experience. Real-time assistance to customers about basic queries.
2. Customer onboarding and product walkthroughs.
3. Summarizing financial results and statements. LLMs can go through annual reports and financial results and provide actionable summary.
4. Financial sentiment analysis – LLMs like Bloomberg GPT can gauge financial sentiment which is useful in investment decisions and risk management.

Continued...

Software Engineering Use Cases:

1. AI co-pilot to help write code which adheres to the syntax of the programming language.
2. Code generation based on natural language prompts.
3. Documentation generation – software release notes, change logs, product documentation can be generated using LLMs.
4. Error finding and Testing.

Education Use Cases:

1. AI co-pilot as a tutor for personalized tutoring: Language models provide customized tutoring and feedback to students, adapting to their unique needs and learning styles.
2. Language learning: Language models are used to build intelligent language learning tools, providing personalized lesson plans and feedbacks to help students learn a foreign language.
3. Adaptive learning: Language models are used to build adaptive learning systems that adjust the difficulty and content of lessons based on a student's progress and needs.

Across Industries Use Cases:

1. Marketing – Ad copy generation, blog generation, SEO keyword generation, text consistency and grammatical correctness checking.
2. Legal – Contract summaries, Contract text generation, Entity extraction from legal documents.
3. Research report summarization – Analyze research reports and highlight important findings.
4. Customer service – Q&A based chatbots for customer service.
5. Localization using machine language translation.

New use cases are emerging even as this book goes to publishing.

02 LLM LANDSCAPE

We have established the fundamentals of Large Language Models (LLMs) in Section 1. Let's shift our focus to the diverse array of operational LLMs. In this section, we take you through the LLM landscape and help you understand which different LLMs are available, their sizes and the companies that are building those models. We will cover ways in which you can implement LLM, which will help you gain a deeper understanding of the implementation approaches. This section concludes with an important article on whether the size of LLMs should be the only consideration while choosing it. After reading this section, you will have a nuanced understanding of the LLM landscape.

Various LLMs available

#	Name	Full form	Company	Parameters	Open/Closed Source	Notes
1	GPT-3	Generative Pre-trained Transformer	Open AI	175B	Closed	GPT-4 released in March 2023 with unknown parameter size.
2	LLaMa	-	Meta	7-65B	Open	Downloadable model. Access to Model only available to researchers and non-commercial personnel.
3	LaMDA	Language Model for Dialogue Applications	Google	173B	Closed	Designed to have more natural and engaging conversations with users.
4	ChatGPT	Chat- Generative Pre-trained Transformer	Open AI	20B	Closed	Provides API access only.
5	MT-NLG	Megatron-Turing Natural Language Generation	Microsoft/ Nvidia	530B	Open	One of the largest and the most powerful monolithic transformer language model trained with 530 billion parameters till date.
6	BLOOM	BigScience Large Open-Science Open-Access Multilingual	BigScience	176B	Open	Multilingual LLM created by a collaboration of over 1,000 researchers from 70+ countries and 250+ institutions.
7	PaLM	Pathways Language Model	Google	540B	Closed	Based on Google's Pathways AI architecture which aims to build models that can handle many different tasks and learn new ones quickly.
8	Dolly 2	-	Databricks	12B	Open	First open source, instruction-following LLM, fine-tuned on a human-generated instruction dataset licensed for research and commercial use.
9	Chinchilla	-	Deepmind	70B	Closed	Requires much less computer power for inference and fine-tuning.
10	Claude	-	Anthropic	Unknown	Closed	Positioned as "Next generation AI assistant".
11	Alpaca	Named after an animal	Stanford	-	Open	It's fine-tuned from Meta's LLaMA 7B model.
12	Vicuna	-	LM-Sys/ University Researchers	13B	Open	Improved version of the Alpaca model, based on the Transformer architecture, but fine-tuned on a dataset of human-generated conversations.

Information valid as of June 2023*

3 ways to implement LLMs

A 2020 study from AI21 Labs pegged the expenses for developing a text-generating model with only 1.5 billion parameters at more than $1 million. So developing your own LLM from scratch isn't really an option for most businesses. Then how do you use LLMs in your business?

1. Use a full-fledged LLM

You can use an LLM like Chat-GPT with API key or an open source LLM. When you use any LLM via API every query has a cost associated with it. Cost for Chat-GPT is typically $0.002 per 1000 queries

When to use:

- When you need a single model that can be used for multiple tasks.
- Need to make predictions based on just a handful of labeled examples.
- When versatility and enterprise access are more important than latency.

Downsides of this approach:

- Model parameters are not released for closed LLMs - so it is a black box
- These models may not work the best on your specific data.
- Data privacy and security concerns exist.

2. Fine-Tuned LLMs

Fine-tuning improves the ability of the model to complete a specific task. You start with an existing LLM and fine-tune it for your specific context. Fine-tuned models are generally smaller than their large language model counterparts.

When to use:

- Good for mature skills with lots of training data.
- Take shorter time and lesser data to train as compared to training full-fledged LLMs.
- Fine-tuning on smaller language models allows users more control to solve their specialized problems using their own domain-specific data.

Downsides of this approach:

- Unlike full-fledged LLMs, fine-tuning still requires a dataset.
- Works only for a narrow problem.

3. Edge LLMs

Purposefully small in size, they can take the form of fine-tuned models. Edge models run offline and on-device, there aren't any cloud usage fees to pay. They offer privacy, no need to connect to cloud.

When to use:

- When you need to run the model on a device or in an offline mode.
- When there are constraints on hardware or size of model.
- When you need privacy and don't want data to go outside your network.

Downsides of this approach:

- You need to train the model from scratch.
- Prone to algorithmic bias.
- Complex models cannot fit on small devices.

Does Model Size Matter?

When considering the evolution of Large Language Models (LLMs) over time, we observe a trend of increasing size, parameters, and corpus used for training. This growth results in larger and larger LLMs. However, when it comes to selecting an LLM for your specific use case, **opting for the largest model may not be the optimal choice.** Model size does matter, but it is not the sole determinant of performance.

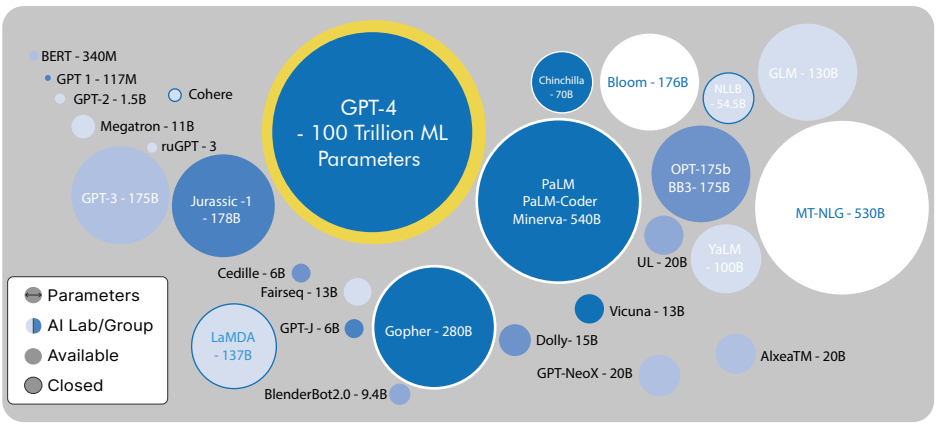
LLM choice should be based on your use case requirements. Larger LLMs offer advantages such as improved performance in language generation and context understanding. Nevertheless, they also come with drawbacks, including higher computational demands, longer inference times, and increased costs.

Instead of focusing solely on size, it is crucial to assess LLM suitability based on factors such as task complexity, available computational resources, and quality of training data. By considering these factors, you can choose an LLM that strikes the right balance between performance and practicality, rather than simply opting for the largest model available.

In many cases, using smaller open-source models, like Alpaca or Vicuna, fine-tuned for your specific context, can be a more efficient choice. These models offer high performance at a fraction of the size and cost of larger LLMs. They also provide transparency as their code and weights (parameters) are publicly available.

However, it's important to note that smaller models may not excel at tasks involving reasoning or mathematics. If your use case requires such capabilities, careful evaluation of your options is necessary.

In summary, model size is a factor to consider when selecting an LLM, but it should not be the sole determining factor. Choose an LLM that aligns with your use case requirements and balances performance, cost, and available resources.



03

LLMs AND YOUR BUSINESSES

In this section, we evaluate the influence of LLMs on business with their transformative potential. We begin with the benefits LLMs can drive for your businesses. Next, we cover the emerging trends in LLMs. This article will help you understand how other businesses are looking at LLMs. We close this section with a practical framework to stimulate brainstorming on how LLMs could be harnessed to benefit your unique business needs. The aim is to empower you to not just understand, but to strategize and innovate using LLMs in your business landscape. After reading this section, you will have a deeper understanding of how LLMs can impact your business.

Business Benefits you can drive with LLMs

LLMs have become the center of attraction ever since Chat-GPT took the world by storm. Every business we interacted with wants to infuse the power of LLMs into their organizations. Any technology gets widely adopted only when there are business benefits. Here are business benefits you can derive from LLMs

Agility / Speed of Operation

Hyper Productivity

LLMs can improve the productivity of your teams anywhere between 2x-10x depending on use case.

Automation 24×7 availability

Automating tedious manual tasks means 24×7 availability as well as improved speed of operations.

Reduced Cycle time

LLMs drastically reduce the cycle time for activities like content generation, research, data collation from documents etc.

Efficiency

Reduced Costs

Automation of manual tasks results in reduction of personnel costs, faster cycle time also reduces overheads and other costs.

Superior Knowledge Management

LLMs empower natural language based document querying and superior knowledge discovery, drastically improving knowledge management.

Improved Accuracy

Automation of any sort results in elimination of human errors. LLM led automation also has this effect.

Organizational Impact

Enhanced Personalization

LLMs can be used for personalization at scale - something which was not possible in the past.

Democratization of AI

Thanks to LLM based natural language querying systems everyone can use the power of AI via simple prompt based chats.

Innovation

LLM based automation helps knowledge workers significantly reduce time spent in non-core activities freeing up bandwidth for innovation.

Emerging Trends in LLMs

The field of Large Language Models has become one of the fastest evolving fields over the last few months. So it becomes very difficult to predict anything as things can change within a matter of days if not hours. However, we still want to share our insights based on what we have seen during our extensive research and experimentation on this topic.

Here are 7 trends we see emerging in the field of LLMs.

1. Companies want to build their own LLMs on their proprietary data to implement use cases like Q&A Bots, Document Search, Co-pilots and Agent led support.

2. Companies want to start with open source LLMs for their first LLM experiments and then based on their use cases and requirements plan to evaluate commercial options.

3. ROI calculation is still evolving - in most cases, LLMs are saving your time and improving productivity. Hyper-productivity is the only theme right now.

4. LLM frameworks are becoming popular. These frameworks like langChain allow you to build end-to-end chains - there is agent, there is code and there is LLM. So it is a complete application.

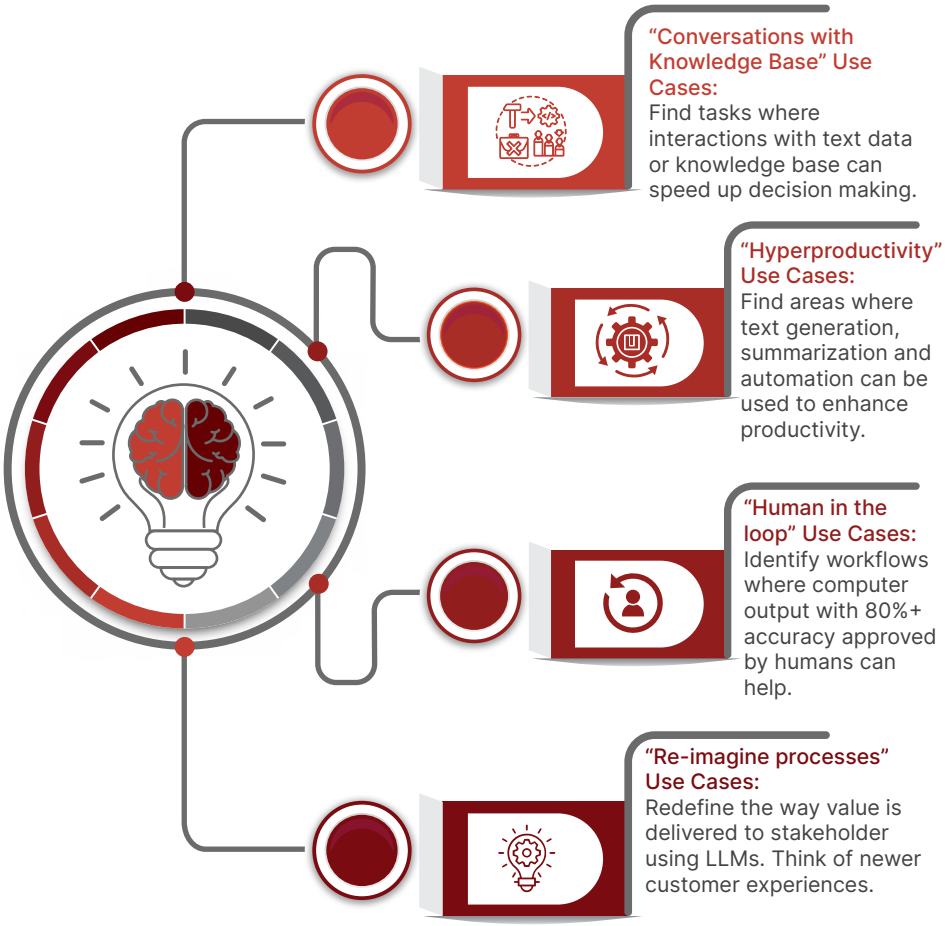
5. Prompt engineering that is how to pose questions to LLMs to get the best answers - is becoming an emerging must have skill.

6. LLMs are democratizing AI due to their simplicity and ease of use. As a famous AI scientist has said "Simpler LLMs will become more widely available and may even come to edge hardware like mobile devices."

7. LLMs, especially generative models, are ushering in an era of hyper-productivity in businesses and there is a sense of urgency to introduce generative AI in workflows. It sometimes seems that there is an actual fear of missing out (FOMO) which is driving businesses towards LLMs and AI.

Framework for brainstorming LLM use cases

Most businesses we talk to focus only on the most common LLM use cases. Here's a framework for you to brainstorm how you can leverage LLMs to unearth new use cases. This framework enables you to think on four different lines to consider the use cases which you can evaluate to prioritize. This framework does not take into consideration the impact of the use cases. It just helps you think in the right direction.



The best way to brainstorm about LLM use cases is to consider LLM strengths like superior context-based search, automated text generation, content summarization, question and answers based conversational information flow etc. can be built into your workflows. Avoid mission critical use cases where anything below 100% accuracy will be an issue, in such cases consider Human-in-the-loop approach.

04

PRACTICAL CONSIDERATIONS WHILE IMPLEMENTING LLMs

This section underscores the practical aspects of implementing Large Language Models. We provide comprehensive articles discussing common pitfalls to avoid with LLMs, the paradigms you need to consider while productionizing LLMs. Additionally, we delve into the profound influence LLMs can exert on your organizational culture and present essential change management considerations. Our objective here is to offer a pragmatic approach towards LLM adoption, emphasizing its strategic impact on your organization and its culture. After reading this section you'll have a well rounded practical perspective on LLMs.

Mistakes to avoid

While LLMs are easy to use and most often correct, any mistakes can prove to be very expensive. Here's our list of mistakes to avoid while using LLMs.

01

Don't trust all the answers

Answers generated by LLMs are the most probable next words. So don't trust all answers. Always fact check.

02

Don't ignore biases

LLM outputs can have inherent biases from the training data. Check the output for any biases.

03

Follow ethical guidelines

LLMs are machines with no understanding for ethical considerations. Always consider ethical guidelines.

04

Overfitting

LLMs can provide correct answers for a majority of questions, yet not fully validating the model can be a grave mistake.

05

Intellectual property

Sometimes LLMs are trained on datasets that may have IP protection. This results in IP infringement, so be cautious about IP.

06

Context

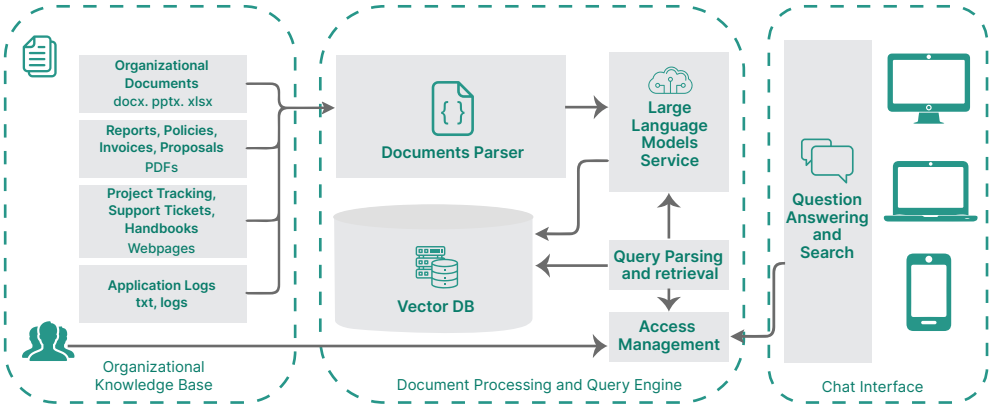
LLMs just generate answer which is probabilistically the best alternative. It may be devoid of context, so never ignore context.

Productionizing LLMs

For business managers, productionizing LLMs is driven by the Return on Investment (ROI). Everyone is talking about AI induced hyper productivity. Most LLM use cases are evaluated on the ROI they generate in terms of time savings. In this article we introduce another ROI paradigm you need to consider while productionizing LLMs - **"Conversational Insights" use case for Productionizing LLMs.**

Conversational bots are popular because of its simplicity and ease of use. We recommend building your own version of conversational chatbot or AI copilot with an open source LLM and a framework such as Langchain. You can use this framework for deriving business insights using your own knowledge base.

We've developed an architecture where you can choose your own model (we have tried Vicuna 13B and Databricks Dolly), your own vector database to train the model on your own knowledge base. This model can be queried using Natural Language queries to get insights on your own data. This conversational chatbot will complement your BI dashboard layer which is generally the data consumption layer in a traditional analytics architecture. In this approach 80% of your insights can come from conversational chatbot or AI co-pilot while the remaining 20% can continue to depend on dashboarding.



Architecture Diagram for Typical Conversational Insights LLM Productionization Approach

This approach where we focus on LLM capabilities to drive data driven decision making for businesses has several advantages over the traditional BI dashboards driven decision making. In addition to the time savings, this approach provides flexibility as well as enhanced user experience.

On the next page, we enlist all the advantages of the Conversational Insights approach.

Productionizing LLMs

Advantages of this approach:

1. Compared to a traditional dashboard development approach, the LLM based approach saves between 30-50% costs while providing more flexibility and enhanced user experience.

2. This approach is modular so you can choose your own model and vector database. You can even evaluate combinations to choose the one that suits you the best.

3. The open-source models mean that you know the parameter weights, and the model provides the transparency businesses need.

4. Using a framework like LangChain enables you to build an end-to-end application without the hassles of providing access to your databases to LLMs.

5. Reduced complexity of information retrieval as compared to the dashboard approach. With Natural Language Q&A, user can directly ask for an insight instead of infering those from a dashboard.

Let's now take a look at use cases driven by time savings as most businesses are focused on using LLMs to drive Hyper-productivity in their businesses.

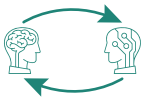
- Developers coding 10x lines of codes with a co-pilot.
- Researchers being able to dig into corpus of 100s of papers.
- Bloggers generating blogs faster using ChatGPT or writing tool.
- An AI ChatBot delivering key product information to consumer.
- Automating responses to FAQs and providing real-time assistance for customer.

All of these use cases along with almost 90% of other use cases are such that their ROI is connected with time savings. The return on investment is justified by the shortened cycle times or reduced costs due to time saved. This is the common ROI paradigm considered by businesses while considering LLM use cases. We recommend considering both these paradigms while planning to productionize LLMs.

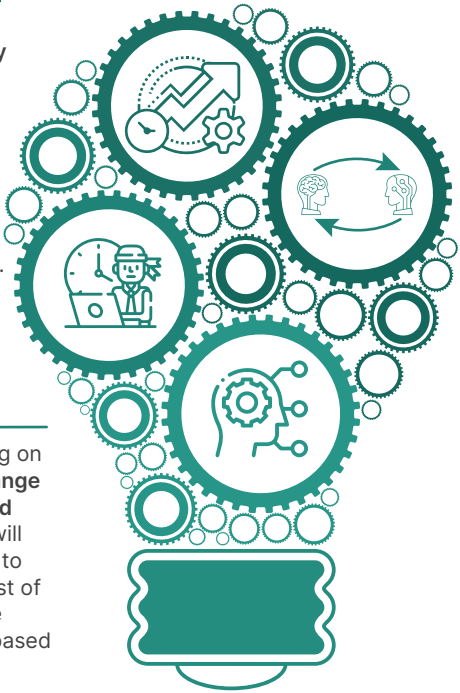
Impact LLMs can have on your org culture

LLMs are increasingly being integrated into specialized applications in fields like writing assistance, coding, and legal research. Risks such as model hallucination, IP related problems and privacy concerns might put a question mark on LLMs, however most businesses want to embed LLMs in their workflows. So how are LLMs likely to impact your organization culture? Let's take a look.

AI powered hyper-productivity will become norm in businesses with up to 10x increase in productivity in tasks such as content writing, research, graphic design, coding etc.



Specialized **human-in-the-loop systems** will have to be implemented in areas where LLMs are likely to provide incorrect outputs.



Employees working on **information exchange tasks or rule based decision making** will have to be moved to other tasks as most of these tasks will be replaced by LLM based applications.

Tasks involving **critical thinking and specialized domain understanding** will be least impacted by LLMs, so organizations will have to focus on hiring personnel with these skills.

- **Prompt Engineering** will become a key skill. Employees who can create good prompts will be more productive.
- Employees will have to be **sensitized on workings of LLMs** so that they fully understand where to trust the inputs provided by LLMs and which cases to cross verify.

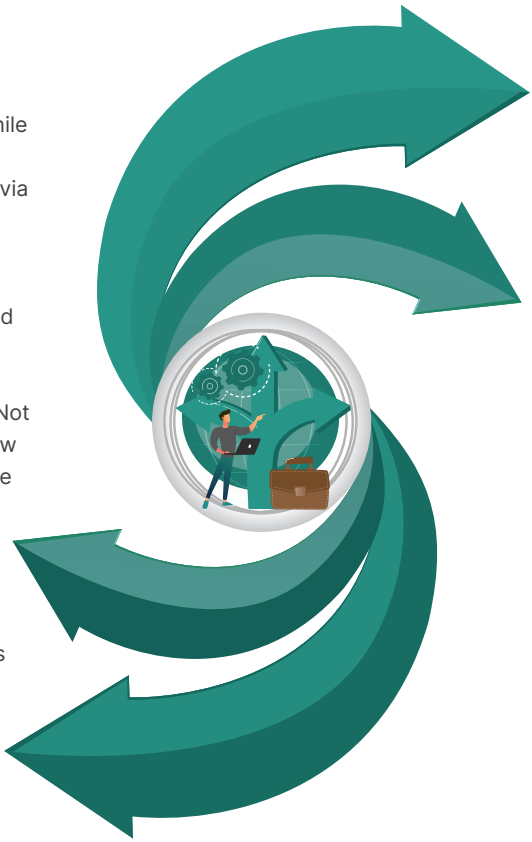
In addition to these changes, organizations will have to become extremely careful about IP while using LLM generated content in their workflows.

Change Management considerations

Given the rapid technology shift after the introduction of LLM-powered Chat-GPT, businesses and knowledge workers are still comprehending how it will impact them. LLMs are likely to transform the nature of the work for most businesses and individuals. Change management plays a crucial role when it comes to adoption of a new technology or tool. In this article, we share some tips on how you should plan and manage your LLM powered AI democratization.

Educate all stakeholders – while most users know how to use LLMs via a chat interface, very few understand the inner workings and risks associated with completely relying on LLMs. Not understanding how LLMs work may be catastrophic in some cases.

Identify strategies for Knowledge Workers to Maximize the Power of LLMs.



Have clear objectives – while beginning your LLM journey, clearly define your objectives.

Define success – what improvement are you trying to achieve in terms of

- Productivity
- Reduced costs
- New capability development
- Process innovation

In the absence of measurable success criteria LLM projects may end up becoming “cool toys” without strategic impact.

Choose the right use case – We recommend beginning with a use case where the potential savings from LLM based workflow will fund the cost of implementing the use case.
For example, if you have a manual process with 10 personnel which you can automate to save 8 person years worth of costs, you should begin with this use case while allocating a budget equivalent to the potential savings.

“As long as you keep the needs of people at the heart of your plan, there are many ways to orchestrate successful and lasting change.”

05

GENERATIVE AI AND LLMs

In this final section, we introduce you to Generative AI which is one of the most hyped fields these days. We also have a small seven question test to test your Generative AI knowledge. We then have some recommendations on how to leverage LLMs. Next, we share insights into how we're aiding our existing customers in harnessing the transformational power of LLMs and Generative AI. We close this section with the answers to the Generative AI test. We recommend that you take a look at our LLM accelerator and our playstore by scanning the QR code on page 32!

Generative AI (GenAI)

What is Generative AI?

- GenAI is a type of Artificial Intelligence that creates new content based on what it has learned from existing content.
- The process of learning from existing content is called training and results in the creation of a statistical model.
- When given a prompt, GenAI uses this statistical model to predict what an expected response might be and this generates new content.

Generative language models

Generative language models learn about patterns in language through training data. Then given some text, they predict what comes next.

Generative image models

Generative image models produce new images using techniques like diffusion. Then given a prompt or related imagery, they transform random noise into images or generate images from prompts.

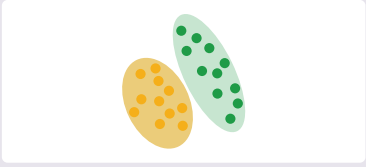
Deep Learning Model Types

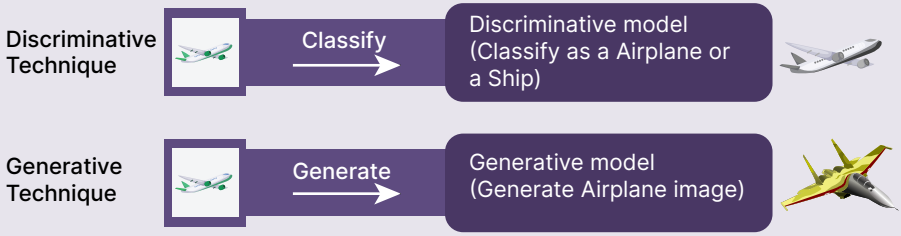
Discriminative

- Used to classify or predict.
- Typically trained on a dataset of labeled data.
- Learns the relationship between the features of the data points and the labels.
- Predicts the class label given features.

Generative

- Generates new data that is similar to data it was trained on.
- Understands distribution of data and how likely a given example is.
- Predict next word in a sequence.
- Predicts features given the class label.



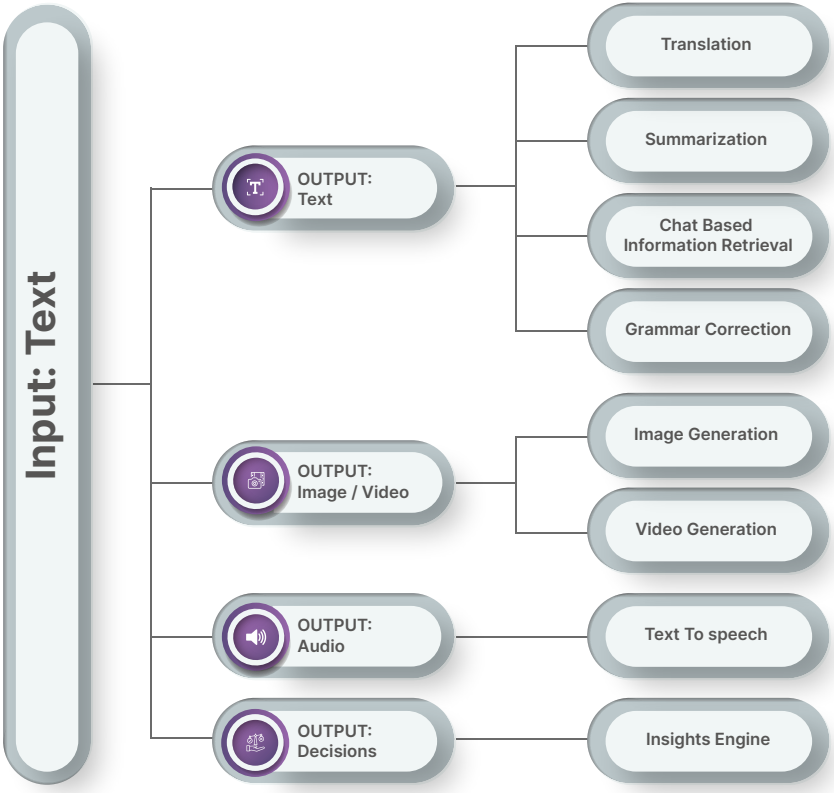


Generative AI uses Generative Models which use existing content to generate new content.

Generative AI - Capabilities

Generative AI is an umbrella term for transformers, large language models, diffusion models and other neural networks which can create text, images, music, software and more.

Fig. Capabilities of Generative AI



Large Language Models based on Transformer architecture generate text output only. For generating image/audio/video Generative Models such as Generative Adversarial Networks (GANs), Diffusion Models, Variational Autoencoders (VAEs), and Flow-based models are used.

Generative Adversarial Networks or GANs-technologies are used. These GANs can create visual and multimedia artifacts from both imagery and textual input data.

A generative adversarial network or GAN is a machine learning algorithm that puts the two neural networks, generator and discriminator, against each other, hence the name "Adversarial".

Diffusion models train by adding gaussian noise to images. The model then learns how to remove this noise. The model then applies this denoising process to random seeds to generate realistic images. Diffusion models can be used for image generation, image denoising, inpainting, outpainting, and bit diffusion.

Generative AI - Test

Generative AI while broadly similar to Large Language Models at high level, is conceptually different. Take this short Generative AI test to check how well you understand its concepts. A score of 5 or more means you know Generative AI fairly well. Take a look at the correct answers and explanations on page 31.

Which of the following is surely not an example of Generative AI?

a. Classification of an image as an image of Cat or Dog by AI
b. Creation of an essay on Cat by AI c. Creation of an image of a black cat by AI
d. Creation of a video of a dog by AI

Generative AI can generate_____.

a. Text and images only b. Text, image and videos
c. Text, audio, video and images d. Only Text and videos

_____ is a short piece of text that is given to the large language model as input, and it can be used to control the output of the model in many ways.

a. Input b. Generator c. Transformer d. Prompt

What term is used to describe an AI model pretrained on vast quantity of data which can be used for performing wide range of Generative AI tasks?

a. Conditional Model b. Neural Network
c. Foundational Model d. Transformer Model

_____ is a term used to describe an occurrence of a factually incorrect output generated by a generative model.

a. Wrong Output b. Model Lying c. Hallucination d. Jailbreak

Generative AI uses _____ Models for generating its output

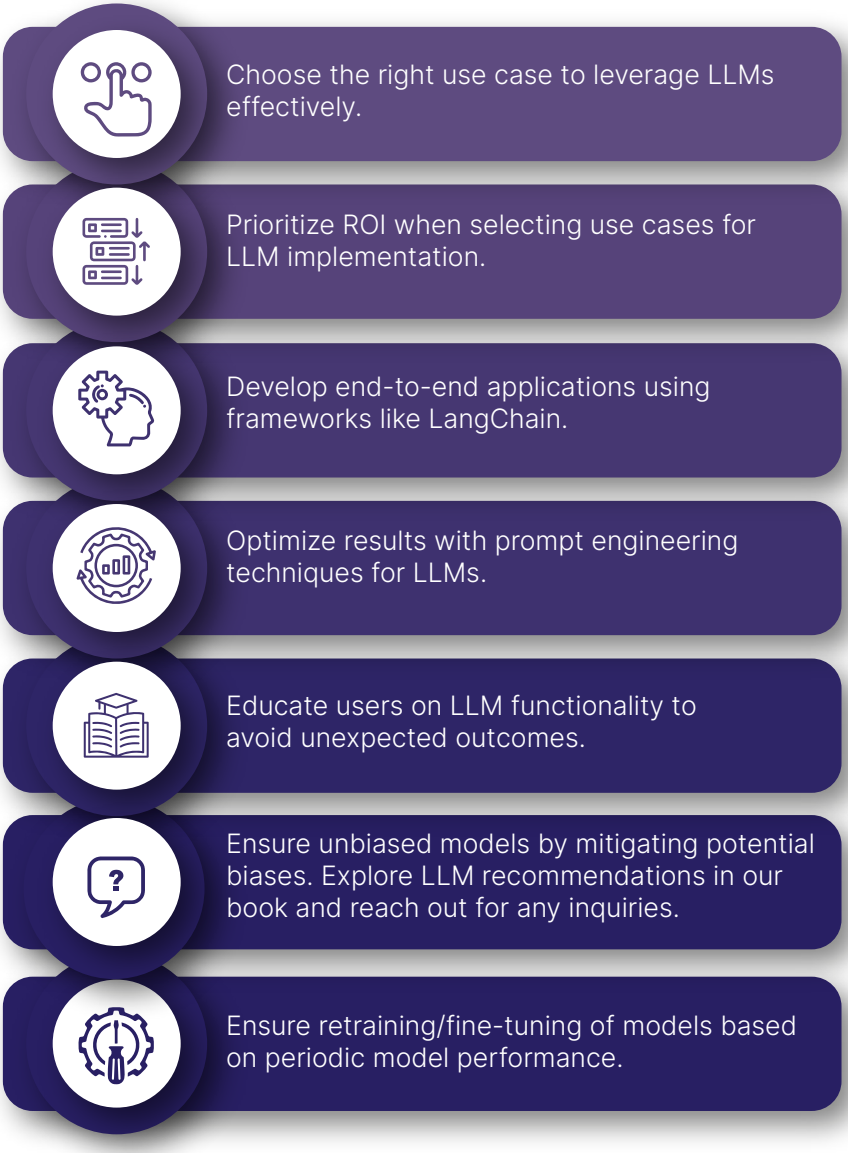
a. Probabilistic b. Deterministic c. Scholastic d. Fantastic

Large Language Models can be used to generate.

a. Text and Image output b. Only Text Output
c. Video and Audio d. Text, Image, Video and Audio

LLM recommendations from Konverge.AI

Many customers seek our advice on implementing LLMs. Here's a compressed, bite - sized dump of our recommendations to our customers when they seek our advice.



What are we doing to help our customers

We work with some of the most cutting-edge startups and growth oriented SMBs. We partner with enterprises to help them drive innovation. All these customer segments are looking at leveraging LLMs and the power of Generative AI to drive a competitive advantage. As the technology partner for our customers, here are 5 things we are doing for our customers:

01 **Readiness Assessment and Use Case prioritization:** Quality and availability of data defines your AI readiness. We are conducting Gen AI readiness assessment and use case prioritization workshops with our customers.

02 **KonGen Framework:** We have come up with our proprietary KonGen framework which helps you accelerate your Generative AI implementations via our ready to use Generative AI accelerators.

03 **PoC factory approach:** For customers who are unsure of the ROI of their Generative AI initiatives, we provide PoC factory approach where our expert teams conduct proof of concepts on customer data to evaluate impact of Gen AI.

04 **LLM Productionizing Consultation:** It's easy to make something cool with LLMs, but very hard to make something production-ready with them. This is why we have our Productionizing Consulting offering where we help customers productionize their LLM use cases using frameworks and agents.

05 **Open Source LLM powered application development:** We have built accelerator on orchestrating LLM powered apps using Open Source LLMs and LangChain Framework. We are using our experience to help our customers build LLM powered apps instead of just Q&A chatbots.

Generative AI Test - Answers

1.A 2.C 3.D 4.C 5.C 6.A 7.B

1. A generative model could generate new photos of animals that look like real animals, while a discriminative model could tell a dog from a cat. In terms of probability given a set of data instances X and a set of labels Y:

   Generative models capture the joint probability p(X, Y), or just p(X) if there are no labels.
   Discriminative models capture the conditional probability p(Y | X).

2. Generative AI generates text, audio, video and images, while large language models can generate only text.

3. Prompt is the technical term used to describe the instructions provided to the generative model.

4. A foundation model (also called base model) is a large machine learning (ML) model trained on a vast quantity of data at scale (often by self-supervised learning or semi-supervised learning) resulting in a model that can be adapted to a wide range of tasks.

5. Model Hallucination occurs when a model generates output which is incorrect and not the expected output.

6. Generative models can generate new data based on probability.
   Discriminative models discriminate between different kinds of data instances.

7. LLMs are fundamentally next word prediction machines. Images and videos are produced by diffusion models which are different from LLMs.

Scan these QR codes to know more about our LLM capabilities and to connect with our team for LLM consultation

Identify where you can apply LLMs - Schedule a meeting with our CEO

Look at Konverge.AI's LLM Capabilities and Accelerators.

If you're finding the book interesting so far, Explore our Play store for more exciting AI content.

AUTHOR

KETAN PAITHANKAR

CO-FOUNDER & CTO - KONVERGE.AI

Ketan Paithankar is the Co-Founder and Chief Technology Officer of Konverge.AI, where he has played a pivotal role since its inception. With a strong focus on leadership, strategy, and management, Ketan oversees the Data & AI Solutioning Advisory & Consulting Practice at Konverge.AI.

Ketan brings a wealth of experience in various fields, including Artificial Intelligence, Deep Learning, Machine Learning & Data Science, Cloud, Data Engineering, Computer Networking, Distributed Systems, DevOps, and Cybersecurity. He holds a B.E. in Electronics Engineering from RKNEC/RCOEM Nagpur and an M.S. in Computer Networking from Wichita State University, Kansas, USA. Additionally, Ketan completed an Executive Programme in General Management from the Indian Institute of Management, Jammu.

Ketan served as a Research Assistant for Cisco Systems during his Master's program. Notably, he received the Mayor's award in a city council meeting for his contributions to building the m-governance platform for the City of Wichita.

Ketan is an accomplished researcher, with published works in the field of AI applied to Cybersecurity and Healthcare. He is a member of the Scientific Committee of the Advances in Distributed Computing and Artificial Intelligence Journal. Ketan's passion for teaching has led him to serve as an adjunct/visiting faculty at prestigious institutions such as IIM Vizag, where he taught MLOps for Managers, as well as various engineering colleges. Ketan is also on Board of Studies (BoS) of multiple engineering colleges and provides recommendations on the structure and contents of the courses related to Artificial Intelligence (AI) and Machine Learning (ML)

Recognized as a dynamic public speaker, Ketan has delivered over 50 talks across different platforms and industries. This book marks Ketan's third publication, following the success of his previous books, "Accelerate your AI Product Journey" and "Modern Data Stack."

Co-Authoring Team

The co-authoring team included **Kartik Vyas** (Independent Marketing Consultant) who is also a visiting faculty at Computer Science Department of VNIT Nagpur. Kartik worked with Ketan to define the structure of the book, framing of articles and provided editorial inputs. **Marvi Chawla** coordinated editorial activities. **Amol Pawar** was the lead designer of this book with support from **Priti Tilloo** and **Jasneet Singh**.

Konverge AI

Scan this QR code to download
an e-copy of this book

LARGE LANGUAGE MODELS