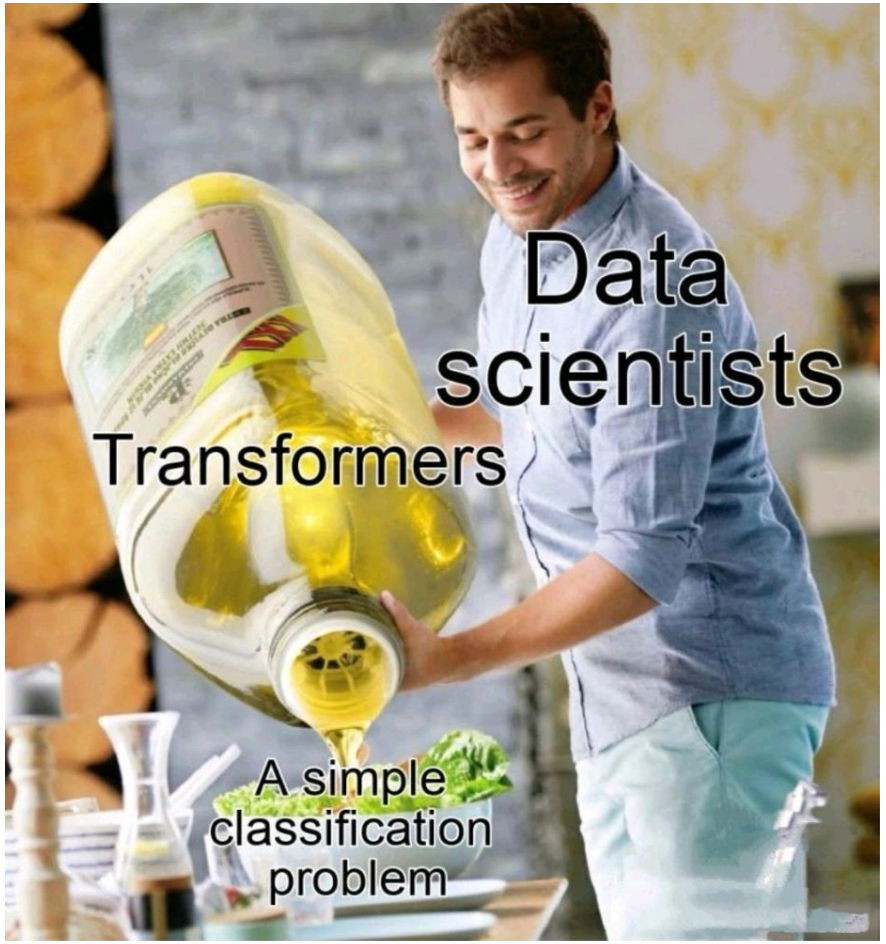
Gen AI - C1

Module 2,
Introduction to Prompt Engineering

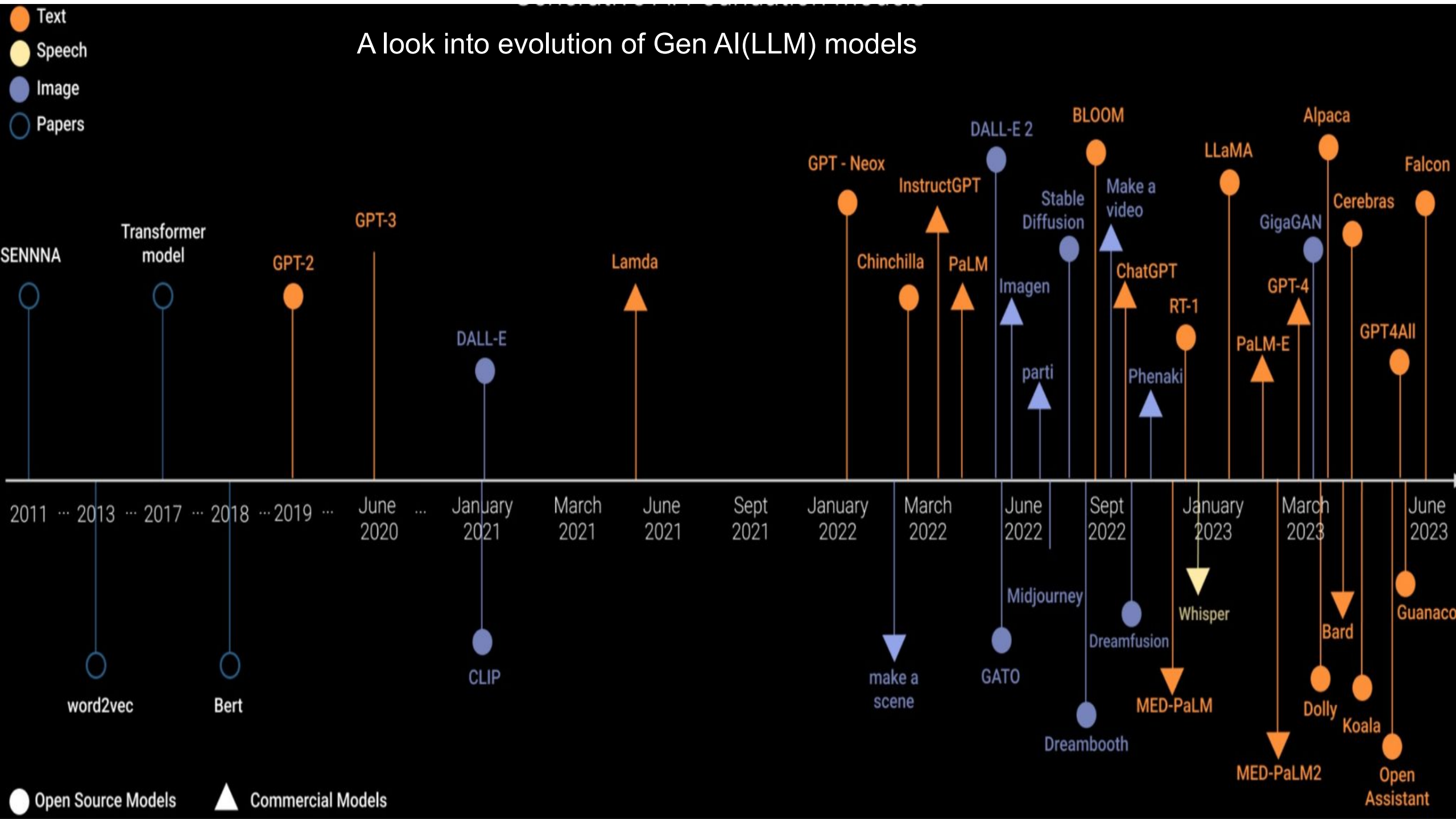
Funny,
But true!



Traditional AI vs Generative AI

Aspect	Traditional AI	Generative AI
Purpose and Functionality	Designed to perform specific tasks, excels in pattern recognition, classification, prediction, and decision-making.	Aims to create new content (text, images, music), generating novel outputs based on learned data patterns.
Approach and Architecture	Uses models like decision trees, SVMs, or basic neural networks, with simpler architectures focused on task-specific domains.	Utilizes complex models like transformers or GANs, capable of generating data by understanding patterns.
Learning Process	Primarily uses supervised learning with labeled datasets, optimizing for accuracy and efficiency.	Often employs unsupervised or semi-supervised learning, focusing on understanding and generating new instances.
Output Nature	Produces deterministic and predictable outputs, commonly used in recommendation systems, fraud detection, and more.	Produces probabilistic and creative outputs, applied in content creation, style transfer, synthetic data generation.
Examples	Examples include IBM Watson, Google's AlphaGo, Spam filters, Credit scoring systems.	Examples include OpenAI's GPT models, DALL-E, DeepMind's AlphaFold, and DeepFake technology.
Challenges and Risks	Challenges include bias in data, lack of interpretability, scalability, and ethical concerns in decision-making.	Challenges include ethical concerns about deepfakes, copyright infringement, generation of harmful content, and misinformation risks.

A look into evolution of Gen AI(LLM) models



Understanding capabilities of LLMS

Input-Based Capabilities
- Textual Inputs
- Multimodal Inputs
- Audio Inputs (Specialized Models)

Output-Based Capabilities
- Textual Outputs
- Multimodal Outputs
- File-Based Outputs
- Interactive & Conversational Outputs

Task-Based Capabilities
- Content Creation
- Information Processing
- Problem Solving
- Educational Support

Domain-Specific Capabilities
- Healthcare
- Legal
- Finance
- Marketing and Sales
- Education

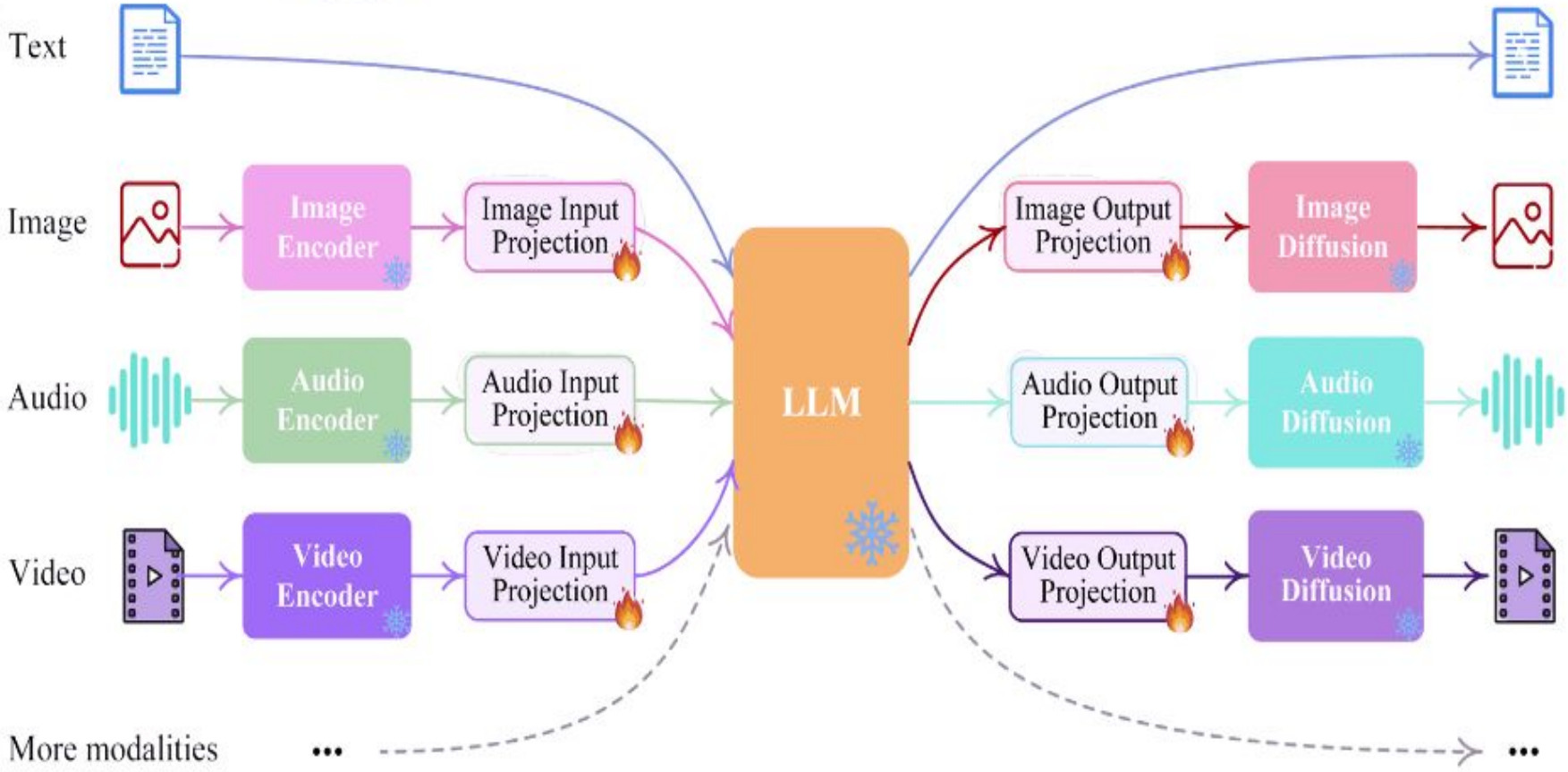
Capability-Based Use Cases
- Automation
- Personalization
- Research Assistance
- Creativity and Innovation

Cross-Functional Capabilities
- Collaboration
- Customization
- Data Interaction
- Security and Compliance

Potential Limitations & Ethical Considerations
- Bias and Fairness
- Ethical Use
- Model Limitations

Any-to-Any LLM models

- text to text (GPT),
- text to image (DALL-E),
- image to text (BLIP),
- speech to text (Whisper)
- text to 3D (DreamFusion)
- text to short video (Make A Video),
- text to longer video (Phenaki),
- video to video (Gen1)
- text to 3D video (Make a video 3D).



Input-Based Capabilities

Textual Inputs

- Natural Language Prompts Interaction using questions, commands, or descriptions.
- Structured Text Input like lists, tables, or formatted documents.
- Code Snippets Input in programming languages.
- Mathematical Expressions Formulas or equations expressed in text.

Multimodal Inputs

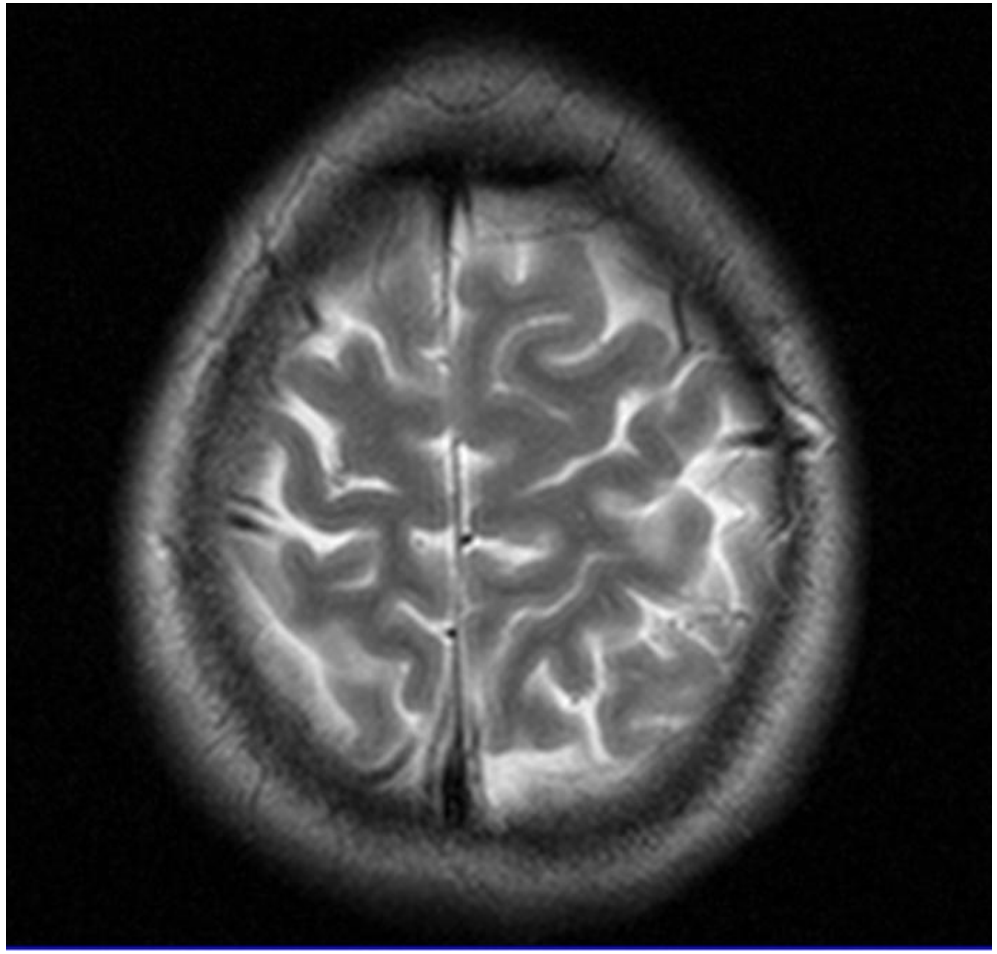
- Images Visual content that can be described or analyzed.
- Text within Images Optical Character Recognition (OCR) to extract and process text from images.
- Tabular & Structured Data: CSV/Excel JSON/XML Hierarchical data formats for extraction and processing.

Audio Inputs (Specialized Models)

- Speech-to-Text Transcription

Example

Image Input



Text Input

Explain briefly the image. Additionally, explain any anomalies observed.

Output-Based Capabilities

Textual Outputs

- Natural Language Responses Coherent answers, explanations, or conversation.
- Summary of text
- Creative Writing Stories, poems, and other creative text.
- Technical Writing Reports, documentation, or manuals.
- Code Generation in various programming languages.
- Mathematical Solutions
- Lists and Tables Organized presentation of data or ideas.

Multimodal Outputs

- Image Descriptions Textual descriptions of visual content.
- Annotated Diagrams Explanations or labels on visual elements.

File-Based Outputs

- Generate PDFs, spreadsheets, or JSON files with structured data.

Interactive & Conversational Outputs

- Dynamic Conversations Contextual responses that adapt to ongoing dialogue.
- Simulations and Role-Playing Engaging in hypothetical scenarios or role-playing based on user inputs.

Example

Text Input

Return Output in .txt format

File Output

- I have created a **.txt** file with the explanation. You can download it using the link below:

Download MRI_scan_explanation.txt

Task-Based Capabilities

Content Creation

- Creative Writing Generation of stories, poetry, and scripts.
- Article and Blog Writing Producing informative or opinion-based content.
- Marketing Copy Crafting ad copy, taglines, or promotional content.

Information Processing

- Summarization Condensing documents, articles, or conversations.
- Data Analysis Parsing and interpreting data, generating reports.
- Translation Converting text between languages.
- Sentiment Analysis Assessing the tone or emotional content of text.

Problem Solving

- Mathematical Problem Solving Solving equations, providing detailed steps.
- Code Debugging in various languages and Generation
- Logic and Reasoning Tasks Solving puzzles, answering logic-based questions.

Educational Support

- Explaining Concepts Providing detailed explanations of complex topics.
- Quiz, educational content and assignment generation.

Example

Code Debugging

Identify the error in the code below:

```
import pandas as pd

df = pd.read-csv("/content/spam.csv",
encoding='latin-1')

df.head(7)
```

Text Output

There is a typo in the code. The correct function name should be read_csv, not read-csv.

Domain-Specific Capabilities

Healthcare

- Medical Text Summarization Summarizing patient records or research papers.
- Diagnostic Assistance Providing potential diagnoses based on symptoms.
- Medical Report Generation Drafting detailed medical documentation.

Legal

- Contract Analysis Reviewing and summarizing legal contracts.
- Legal Document Drafting Creating agreements, wills, or legal notices.
- Case Summarization Condensing legal cases or briefs.

Finance

- Financial Reporting Analyzing and summarizing financial data.
- Market Analysis Generating insights based on market trends.
- Risk Assessment Evaluating financial risks based on data inputs.

Marketing and Sales

- Customer Persona Generation Creating detailed customer profiles.
- Ad Copy Creation Writing persuasive sales copy.
- Market Research Summarization Summarizing market research reports.

Education

- Creating educational materials based on curriculum standards.
- Study Guide Creation Summarizing textbooks or academic papers.
- Interactive Learning Providing real-time assistance or tutoring.

Example

Education

Create a detailed script on the working of Poe (an integrated AI platform).

Text Output

- Poe is a platform that allows users to chat with a variety of AI-powered bots and have back-and-forth conversations. It is available on multiple platforms including iOS, Android, MacOS, Windows, and the web.
- Creating bots on Poe provides benefits such as distribution to Poe's user base.
- Overall, Poe aims to be a platform that makes it easy for anyone to explore and utilize the latest advancements in AI technology through interactive conversations with a wide range of bots.

Capability Based Use Cases

Automation

- Automating repetitive tasks like data entry or report generation.
- Workflow Integration: Integrating LLMs into business processes

Personalization

- Custom Content Generation Tailoring content to specific audiences or preferences.
- Conversational Agents Building personalized chatbots or virtual assistants.

Research Assistance

- Literature Review Summarizing and analyzing academic papers.
- Data Extraction Extracting key information from large datasets.
- Hypothesis Testing Assisting in the design and analysis of research experiments.

Creativity and Innovation

- Idea Generation Brainstorming ideas for products, stories, or marketing strategies.
- Design Assistance Providing inspiration or descriptions for design concepts.

Example

Personalization

Create a detailed plan to deliver python knowledge to the working professionals with 5-20 years of experience.

Text Output

- Assess the Audience
- Set Learning Objectives
- Choose the Delivery Method
- Develop a Curriculum
- Engage Experienced Instructors
- Provide Hands-On Practice

Cross-Functional Capabilities

Collaboration

- Co-writing Collaborating on content creation with human users.
- Team Communication Generating summaries of meetings or project updates.

Customization

- Adaptable Outputs Generating content in various styles, tones, or formats.
- Parameter Tuning Adjusting the model's behavior or outputs based on specific goals.

Data Interaction

- Data Visualization Support Generating text that describes data trends or suggests visuals.
- Dynamic Data Queries Interpreting and responding to complex data queries.

Security and Compliance

- Content Filtering Identifying and filtering inappropriate or sensitive content.
- Compliance Checks Ensuring generated content adheres to legal or regulatory standards.

Example

Customization

Check if the language used in the following notebook is inappropriate: M1_AST_01_SklearnPipeline.ipynb

Text Output

Snap: The notebook is well-structured for educational purposes, covering the full pipeline from data loading and preprocessing to model training and hyperparameter tuning. The use of #@title helps in hiding the setup and feedback-related code, making the notebook cleaner and more focused on the core learning objectives.

Potential Limitations & Ethical Considerations

Bias and Fairness

- Understanding Model Bias Recognizing potential biases in LLM outputs.
- Mitigation Strategies Techniques to reduce bias and ensure fair outcomes.

Ethical Use

- Content Authenticity Ensuring the generated content is truthful and reliable.
- Privacy Concerns Managing and protecting sensitive data.

Model Limitations

- Understanding Scope Recognizing the limitations in knowledge, creativity, and reasoning.
- Error Handling Strategies for managing inaccuracies or misinterpretations.

Why Prompt Engineering is essential?

Model Limitation: Hallucinations

```
# Dabur is real company but doesn't have anything as Sandalwood handwash

prompt = f"""
Tell me about Dabur Sandalwood Handwash
"""
```

To ensure LLMs produce outputs that are accurate, ethical, and socially responsible, there is critical need for alignment

In LLMs, alignment is ensured by the following 3 primary strategies:

- **Prompting**: text provided as input to LLM, that can include additional context or requirements
- **Retrieval-Augmented Generation (RAG)**: Introducing a component that fetches relevant documentation based on Question, from knowledge base
- **Advanced fine-tuning**

Maximizing the potential of LLMs:
(By improving prompts)

Prompting Techniques
- Clear and Specific Prompts
- Use of Contextual Prompts
- Chain of Thought Prompting
- Few-Shot Learning

Role-Playing and Simulation
- Assigning Roles to the Model
- Scenario Simulation

Adding Context and Constraints
- Contextual Embedding
- Multi-Prompt Contextualization

Iterative Refinement
- Feedback Loop
- Output Verification

Leveraging Multimodal Capabilities
- Combining Text and Images
- Data and Text Integration

Creative Use of Output Formats
- Different Content Styles
- Multi-Part Outputs

Domain-Specific Customization
- Tailored Prompts for Different Domains
- Cross-Domain Synthesis

Enhancing Reliability and Accuracy
- Use of External Validation
- Setting Boundaries

Interactive and Iterative Use
- Dynamic Dialogue
- Prompt Chaining

Exploring Ethical and Responsible Use
- Bias Mitigation
- Ethical Prompting

Use of Meta-Prompts
- Prompt Reflection
- Self-Improvement Queries

Leveraging LLM APIs and Tools
- Customization Through APIs
- Integration with Other Tools

Prompting Techniques

Clear and Specific Prompts

- Avoid Ambiguity Ensure the prompt is clear and precise to reduce the chance of irrelevant or incorrect outputs.
- Targeted Instructions Be specific about what you want, such as "write a 200-word summary" instead of just "summarize."

Use of Contextual Prompts

- Provide relevant context to guide the model's responses.
- Scenario-Based Prompts.Create a scenario that the model should follow, e.g., "Imagine you are a teacher explaining photosynthesis to a 10-year-old."

Chain of Thought Prompting

- Step-by-Step Instructions Ask the model to reason through a problem in multiple steps, which can improve accuracy.
- Progressive Detail Start with a broad request and then ask for more specific details based on the initial output.

Few-Shot Learning

- Provide Examples Show the model a few examples of the desired output to guide it in generating similar results.
- Template-Based Prompts Use a template or structure in the prompt, which the model can follow.

Example1

Chain of Thought Prompting

A group of friends goes to a restaurant and the bill is $120. They decide to split the bill equally among themselves. If each person pays $20, how many friends are in the group?

Text Output

- Each friend pays $20, and the total bill is $120.
- Number of friends = Total Bill / Amount per Friend
- Number of friends = $120 / $20
- Number of friends = 6
- There are 6 friends in the group.

Try!

Ask the model to check whether conditions are satisfied

```
text = f"""
Making a cup of tea is easy! First, you need to get some water boiling. While that's happening, grab a cup and put a tea bag in it. Once the water is hot enough, just pour it over the tea bag. Let it sit for a bit so the tea can steep. After a few minutes, take out the tea bag. If you like, you can add some sugar or milk to taste. And that's it! You've got yourself a delicious cup of tea to enjoy."""

prompt = f"""
You will be provided with text delimited by triple quotes.If it contains a sequence of instructions,
re-write those instructions in the following format:
Step 1 - ...
Step 2 - …
…
Step N - …
If the text does not contain a sequence of instructions, \then simply write \"No steps provided.\"
\"\"\"{text}\"\"\"
"""
```

Try!

"Few-shot" prompting

```
prompt = f"""
Your task is to answer in a consistent style.

<child>: Teach me about patience.

<grandparent>: The river that carves the deepest valley flows from a modest spring; the grandest symphony originates from a single note; he most intricate tapestry begins with a solitary thread.

<child>: Teach me about resilience.
"""
```

Role-Playing and Simulation

Assigning Roles to the Model

- Expert Roles Ask the model to act as an expert in a specific field, e.g., "You are a financial advisor. Explain the benefits of diversification."
- Persona Adoption Have the model adopt a persona or character to fit the tone or style needed, such as "Write like a Shakespearean character."

Scenario Simulation

- Interactive Simulations Use the model to simulate dialogues, customer service interactions, or role-play scenarios for training or practice.
- What-If Analysis Pose hypothetical scenarios and ask the model to explore different outcomes.

Example

Assigning Roles to the Model

Explain the newton's laws of motion as Prof. of undergraduate students.

Text Output

Newton’s laws work together to explain how objects move and interact. In real-life situations, these laws are constantly at play. For example, when you drive a car, Newton’s first law explains why you need to keep your foot on the gas to maintain speed (overcoming friction), the second law explains how the car accelerates when you press the gas pedal, and the third law explains why the car moves forward as the tires push backward against the road.

Adding Context and Constraints

Contextual Embedding

- Historical or Cultural Context Provide historical, cultural, or domain-specific context to improve the relevance and depth of responses.
- Detailed Instructions Include specific constraints or guidelines, such as "Respond in a formal tone" or "Limit the answer to three sentences."

Multi-Prompt Contextualization

- Sequential Prompts Break down complex tasks into smaller parts with sequential prompts, refining the model's output step by step.
- Layered Prompts Start with a general prompt and progressively add context or constraints to narrow down the focus.

Try!

Specify the steps required to complete a task

```
text = f""" In a charming village, siblings Jack and Jill set out on a quest to fetch water from a hilltop well. As they climbed, singing joyfully, misfortune struck—Jack tripped on a stone and tumbled down the hill, with Jill following suit. Though slightly battered, the pair returned home to comforting embraces. Despite the mishap, their adventurous spirits remained undimmed, and they continued exploring with delight."""

prompt_1 = f"""Perform the following actions:
1 - Summarize the following text delimited by triple backticks with 1 sentence.
2 - Translate the summary into French.
3 - List each name in the French summary.
4 - Output a json object that contains the following keys: french_summary, num_names.
Separate your answers with line breaks.

Text:```{text}```
```

Iterative Refinement

Feedback Loop

- Refine Prompts Based on Output Use the model's initial output to refine the prompt and ask for improvements or clarifications.
- Progressive Refinement Start with a broad question and iteratively narrow down based on the model's responses.

Output Verification

- Ask for Self-Verification Request the model to verify or cross-check its own response, which can help catch errors.
- Cross-Questioning Pose follow-up questions based on the initial output to ensure accuracy and completeness.

Example

Prompt Involving Iterative Refinement

Initial Prompt:
"Write a brief summary of the impact of climate change on agriculture."

Refinement Prompt:
"Can you refine this summary by providing more specific examples of how climate change affects different types of crops in various regions?"

Further Refinement Prompt:
"Could you enhance the summary by including some data or statistics to illustrate the scale of these impacts?"

Leveraging Multimodal Capabilities

Combining Text and Images

- Image Description with Context Provide an image and ask the model to describe it in detail, considering specific contexts or scenarios.
- Visual Prompting Use images as part of the prompt to guide the model's understanding and generation of text.

Data and Text Integration

- Tabular Data Analysis Provide structured data along with specific questions to generate insights, summaries, or recommendations.
- Combining Documents and Queries Upload multiple documents and ask the model to synthesize information across them.

Try!

Format Conversion

```
data_json = { "resturant employees" :[
    {"name":"Shyam", "email":"shyamjaiswal@gmail.com"},
    {"name":"Bob", "email":"bob32@gmail.com"},
    {"name":"Jai", "email":"jai87@gmail.com"}
]}

prompt = f"""
Translate the following python dictionary from JSON to an HTML \
table with column headers and title: {data_json}
"""
```

Creative Use of Output Formats

Different Content Styles

- Formal vs. Informal Specify the style of writing, whether formal, conversational, or creative.
- Format-Specific Outputs Request outputs in specific formats, such as bullet points, numbered lists, or structured documents.

Multi-Part Outputs

- Sectioned Responses Break down outputs into different sections, such as "Introduction," "Body," and "Conclusion."
- Parallel Responses Ask the model to provide different perspectives on the same topic.

Try!

Ask for output in a specified format

```
prompt = f""" Your task is to perform the following actions:
1 - Summarize the following text delimited by <> with 1 sentence.
2 - Translate the summary into French.
3 - List each name in the French summary.
4 - Output a json object that contains the following keys: french_summary, num_names.

Use the following format:
Text: <text to summarize>
Summary: <summary>
Translation: <summary translation>
Names: <list of names in Italian summary>
Output JSON: <json with summary and num_names>

Text: <{text}>
```

Domain-Specific Customization

Tailored Prompts for Different Domains

- Industry-Specific Language Use terminology and language specific to a domain (e.g., legal, medical, technical) to guide more accurate responses.
- Contextual Examples Provide domain-relevant examples or scenarios to guide the model’s responses.

Cross-Domain Synthesis

- Interdisciplinary Insights Ask the model to combine knowledge from multiple domains to generate innovative solutions or perspectives.
- Comparative Analysis Request comparisons across different fields, such as "Compare the marketing strategies in the tech and fashion industries."

Try!

Domain-specific customization can enhance the relevance and depth of information provided, ensuring that the response is tailored to the specific context

```
prompt = f""" Could you elaborate on Agile project management principles using specific examples from software development and highlight the roles of Scrum Master and Product Owner? """
```

Enhancing Reliability and Accuracy

Use of External Validation

- Cite Sources Ask the model to provide references or cite sources when generating factual content.
- Fact-Checking Use external tools or models to validate the information provided by the LLM.

Setting Boundaries

- Limitations in Response Define the scope or limits of the response, such as “Explain the theory, but do not go into advanced mathematics.”
- Avoiding Hallucination Use constraints or prompts that minimize the risk of the model generating incorrect or nonsensical information.

Interactive and Iterative Use

Dynamic Dialogue

- Conversation Continuation Use the model in a conversation mode, where it builds on previous interactions to provide richer and more context-aware responses.
- Real-Time Adjustments Modify the prompts based on the ongoing interaction to steer the conversation in a desired direction.

Prompt Chaining

- Linked Prompts Create a series of prompts that are connected, with each one building on the previous output to deepen the interaction.
- Scenario Evolution Start with a simple scenario and gradually add complexity, asking the model to adapt its responses as the scenario unfolds.

How to store/use chat history? (brief)

```
messages =  [
{'role':'system', 'content':'You are friendly chatbot.'},
{'role':'user', 'content':'Hi, my name is Isa'},
{'role':'assistant', 'content': "Hi Isa! It's nice to meet you. \
Is there anything I can help you with today?"},
{'role':'user', 'content':'Yes, you can remind me, What is my name?'}  ]
```

This chat history will be send as a parameter to LLM with every new request. Subsequently, each new user message and chatbot response are appended to this history.

We'll discuss this in-detail in upcoming sessions

Exploring Ethical and Responsible Use

Bias Mitigation

- Awareness of Bias Recognize potential biases in the model's outputs and prompt the model to consider diverse perspectives.
- Inclusive Language Encourage the use of inclusive and non-discriminatory language in outputs.

Ethical Prompting

- Avoid Sensitive Topics Be cautious about prompts that could lead to harmful, biased, or unethical responses.
- Transparency Encourage the model to be clear about the limitations of its outputs, especially in sensitive domains like healthcare or law.

Example

Prompt Encouraging Clarity on Limitations

```
prompt = f""" Provide advice on how to treat mild depression. Also, clarify the
limitations of advice and emphasize the importance of consulting a healthcare
professional in sensitive cases like mental health. """
```

Use of Meta-Prompts

Prompt Reflection

- Ask for Explanation Request the model to explain its reasoning process or how it arrived at a particular conclusion.
- Meta-Discussion Engage the model in a discussion about its own outputs, asking for critiques or improvements.

Self-Improvement Queries

- Improvement Suggestions Ask the model how it could improve its own response or what additional information it would need.
- Prompt Refinement Advice Request the model to suggest how the prompt could be better framed for a more accurate response.

Example

prompt = f""" You are an expert in environmental science. Provide a concise explanation of the impact of plastic pollution on marine life, followed by a brief list of solutions to mitigate this issue """

Meta-Prompt to Encourage Reflection on the Response:
"Can you analyze how comprehensive and accurate your list of solutions are? Are there any aspects that you might have overlooked?"

Leveraging LLM APIs and Tools

(Will discuss in detail in upcoming sessions)

Customization Through APIs

- Fine-Tuning Models Use API features to fine-tune the model for specific tasks or domains, improving relevance and accuracy.
- Parameter Adjustments Experiment with API settings like temperature, max tokens, and top-p to control the creativity and specificity of responses.

Integration with Other Tools

- Toolchain Integration Combine LLM outputs with other software tools for post-processing, such as using LLMs for generating data that is then processed by specialized analytic tools.
- Chaining Models Use outputs from one model as inputs to another, chaining them together for more complex workflows.

Let’s Try Experimenting

(Summarize Product Review)

Text to summarize:

```
"""
Got this panda plush toy for my daughter's birthday, \
who loves it and takes it everywhere. It's soft and \
super cute, and its face has a friendly look. It's \
a bit small for what I paid though. I think there \
might be other options that are bigger for the \
same price. It arrived a day earlier than expected, \
so I got to play with it myself before I gave it \
to her.
"""
```

Let's Try Experimenting contd..

To try: Prompt 1

Summarize with a word/sentence/character limit

```
"""
Your task is to generate a short summary of a product \
review from an ecommerce site.

Summarize the review below, delimited by triple
backticks, in at most 30 words.

Review: ```{prod_review}```
"""
```

Let's Try Experimenting contd..

To try: Prompt 2

Summarize with a focus on shipping and delivery

```
"""
Your task is to generate a short summary of a product \
review from an ecommerce site to give feedback to the \
Shipping deparmtment.

Summarize the review below, delimited by triple
backticks, in at most 30 words, and focusing on any
aspects \
that mention shipping and delivery of the product.

Review: ```{prod_review}```
"""
```

Let's Try Experimenting contd..

To try: Prompt 3

Summarize with a focus on price and value

```
"""
Your task is to generate a short summary of a product \
review from an ecommerce site to give feedback to the \
pricing deparmtment, responsible for determining the \
price of the product.

Summarize the review below, delimited by triple
backticks, in at most 30 words, and focusing on any aspects
\
that are relevant to the price and perceived value.

Review: ```{prod_review}```
"""
```

Let's Try Experimenting contd..

To try: Prompt 4

Try "extract" instead of "summarize"

```
"""
Your task is to extract relevant information from \
a product review from an ecommerce site to give \
feedback to the Shipping department.

From the review below, delimited by triple quotes \
extract the information relevant to shipping and \
delivery. Limit to 30 words.

Review: ```{prod_review}```
"""
```

"Prompt engineering is the art of communicating eloquently to an AI."