DEVCONF.us

Self Hosted LLMs

Aakanksha Duggal
Senior Data Scientist

Hema Veeradhi
Principal Data Scientist

Hema Veeradhi

Principal Data Scientist

Emerging Tech, Office of the CTO

Aakanksha Duggal

Senior Data Scientist

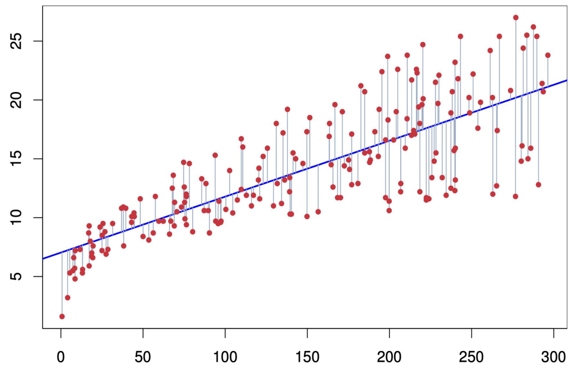
InstructLab

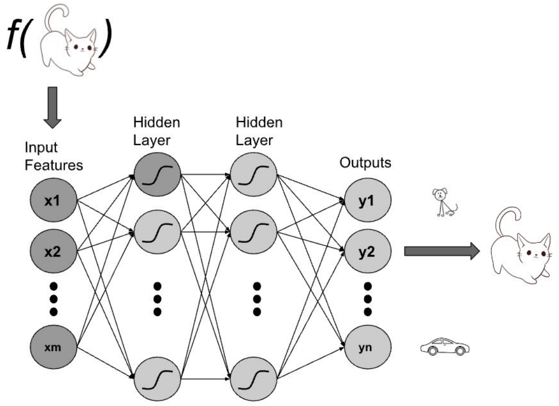
Agenda

- Introduction to LLMs
- Open source LLMs
- Steps for building an LLM application
- Concept of self-hosting LLMs
- Setting up LLMs using Podman
- Demo
- Q&A

Language Models

A **language model** is a type of machine learning model trained to conduct a probability distribution over words.





Types of Language Models:

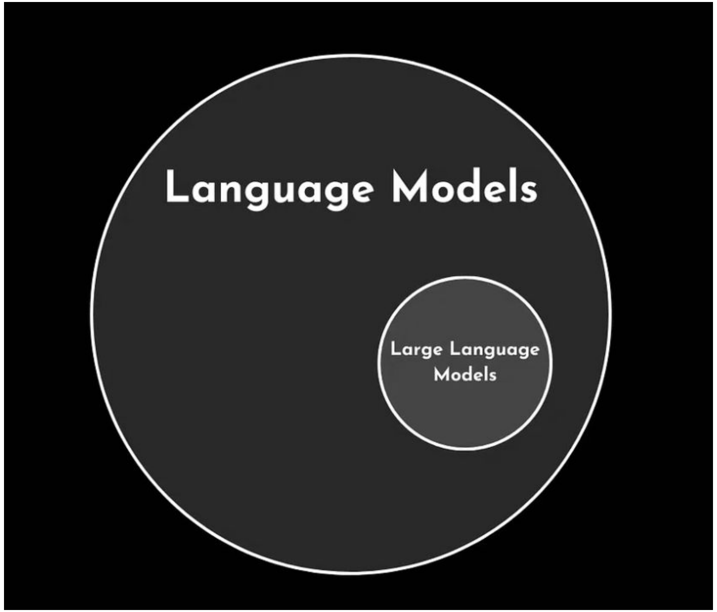
- **Statistical language models**
- **Neural language models**
  - ○ **RNN**
  - ○ **LSTM**
  - ○ **Transformers**

Source: Numerary

Large Language Models

- **L**arge **L**anguage **M**odel - **LLM** is just a larger version of a language model

- **WHY LLMs?**
  - **Quantitative :** Number of Parameters, **10–100 billion** parameters
  - **Qualitative :** Self-supervised learning



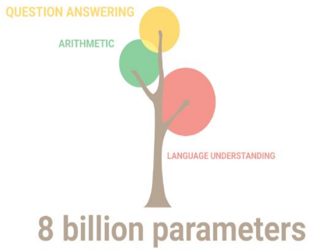
Source: Medium Blog

LLM Applications

A Large Language Model is a type of Artificial Intelligence that is trained on a massive dataset of text and code. This allows the model to learn **statistical relationships between words and phrases which in turn allows it to generate text, translate languages, write creative content and answer your questions in an informative way.**

Here are common LLMs:

- **GPT3.5 and GPT 4**
- **Gemini**
- **Llama, Llama2**



Source: Medium Blog

Open Source vs Closed Source Models

[List of Open Source Models available for commercial use]

Open Source models

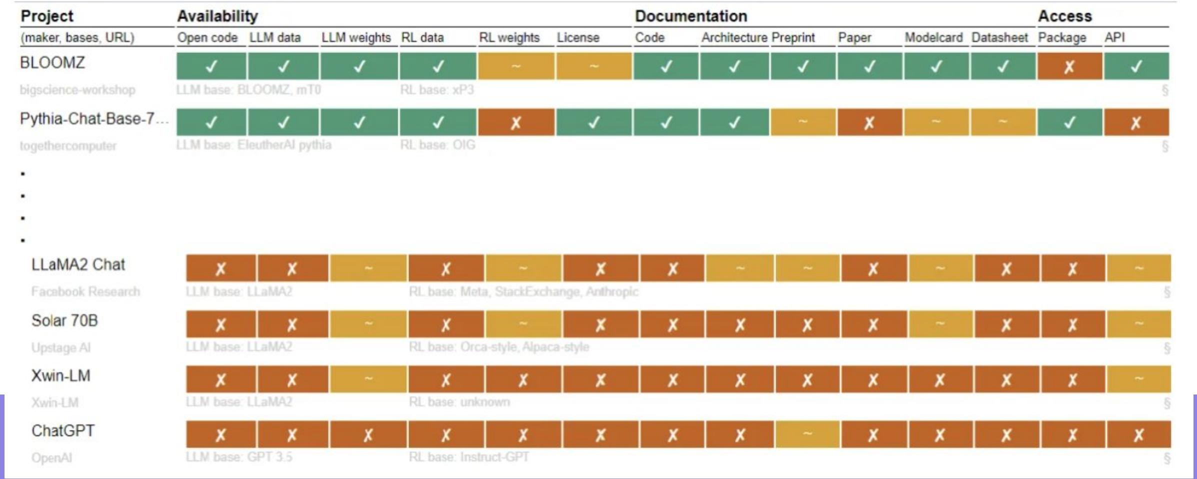
- LLaMA-2 by Meta (claim to be)
- Falcon by Technology Innovation Institute in Abu Dhabi

Mistral by Mi

Closed Source models

- GPT-4 by OpenAI
- Gemini by Google

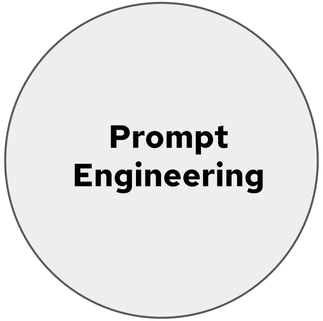
Claude by A



Project	Availability						Documentation						Access	
(maker, bases, URL)	Open code	LLM data	LLM weights	RL data	RL weights	License	Code	Architecture	Preprint	Paper	Modelcard	Datasheet	Package	API
BLOOMZ	✓	✓	✓	✓	~	~	✓	✓	✓	✓	✓	✓	✗	✓
bigscience-workshop	LLM base: BLOOMZ, mT0			RL base: xP3										§
Pythia-Chat-Base-7…	✓	✓	✓	✓	✗	✓	✓	✓	~	✗	~	~	✓	✗
togethercomputer	LLM base: EleutherAI pythia			RL base: OIG										§
.														
.														
.														
.														
LLaMA2 Chat	✗	✗	~	✗	~	✗	✗	~	~	✗	~	✗	✗	~
Facebook Research	LLM base: LLaMA2			RL base: Meta, StackExchange, Anthropic										§
Solar 70B	✗	✗	~	✗	~	✗	✗	✗	✗	✗	~	✗	✗	~
Upstage AI	LLM base: LLaMA2			RL base: Orca-style, Alpaca-style										§
Xwin-LM	✗	✗	~	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	~
Xwin-LM	LLM base: LLaMA2			RL base: unknown										§
ChatGPT	✗	✗	✗	✗	✗	✗	✗	✗	~	✗	✗	✗	✗	✗
OpenAI	LLM base: GPT 3.5			RL base: Instruct-GPT										§

Source, Source 2

Levels of LLMs

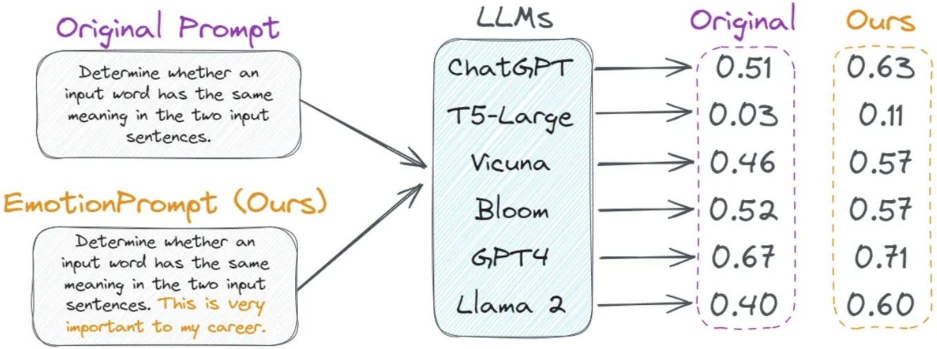


Prompt Engineering

Levels of LLMs

Prompt Engineering

- “Let’s think step by step”
- “Take a deep breath and work on this problem step-by-step“
- “This is very important to my career”



Original Prompt

Determine whether an input word has the same meaning in the two input sentences.

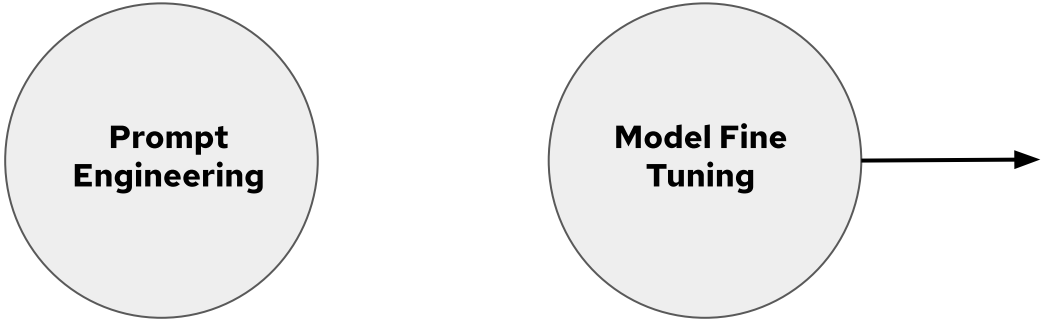
EmotionPrompt (Ours)

Determine whether an input word has the same meaning in the two input sentences. This is very important to my career.

LLMs	Original	Ours
ChatGPT	0.51	0.63
T5-Large	0.03	0.11
Vicuna	0.46	0.57
Bloom	0.52	0.57
GPT4	0.67	0.71
Llama 2	0.40	0.60

Source: Medium Blog

Levels of LLMs



Step 1:
A Pre-trained LLM

Step 2:
Update model parameters for a specific task

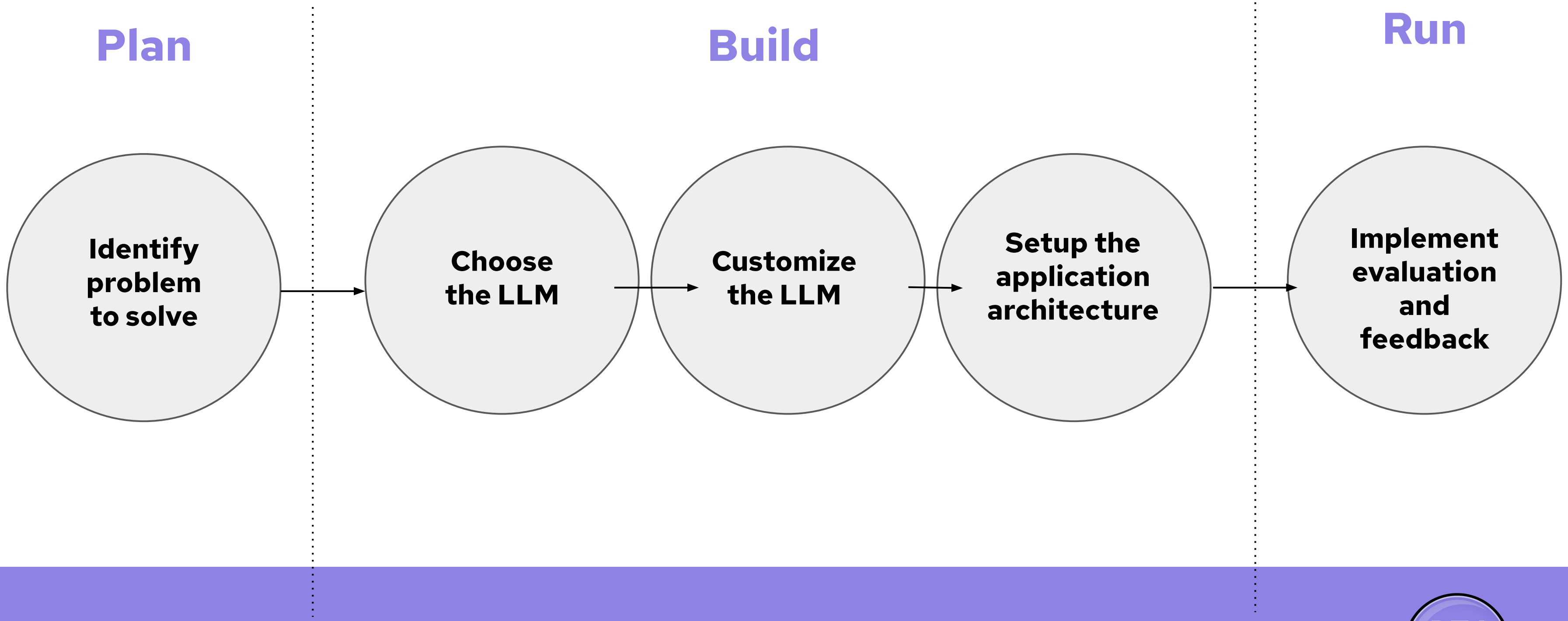
Levels of LLMs

Prompt Engineering

Model Fine Tuning

Build your own LLM

Steps for Building an LLM Application



Plan

Build

Run

Identify problem to solve

Choose the LLM

Customize the LLM

Setup the application architecture

Implement evaluation and feedback

To self-host or not to?

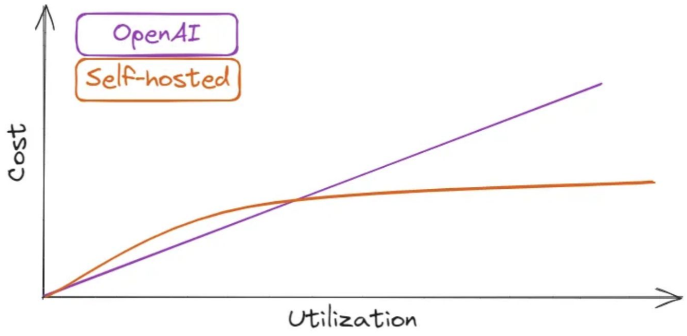
- In the past year, the discussion surrounding LLMs has evolved, transitioning from **"Should we utilize LLMs?"** to "**Should we opt for a self-hosted solution or rely on a proprietary off-the-shelf alternative?"**

- Depending on your use-case, computational needs and engineering architecture availabilities you can decide whether to self-host your LLM

- Hosted models are **necessary** for privacy, reliability, or compliance

Benefits of self-hosting LLMs

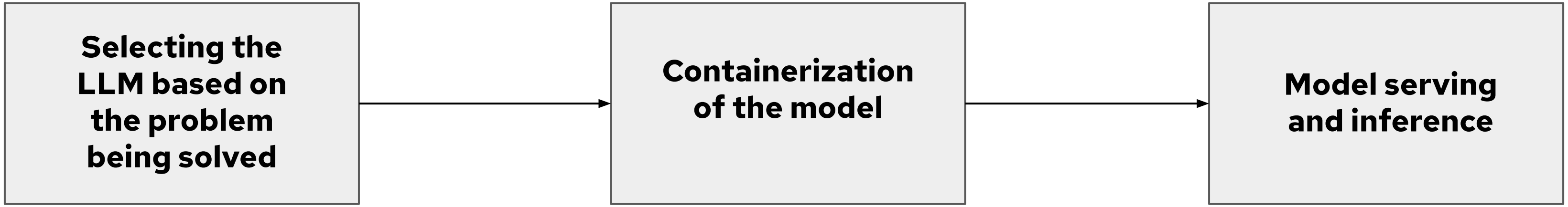
- Greater security, privacy, and compliance
- Customization
- Avoid vendor lock-in
- Save computational costs
- Easy to get started for those new to or just starting their journey with LLM



Schematic comparison of OpenAI GPT-3.5 and self-hosted LLMs

Source: Medium Blog

Self-Hosting Containerized LLMs



Selecting the LLM based on the problem being solved

Containerization of the model

Model serving and inference

Hugging Face

https://huggingface.co/

docker

podman

FastAPI

DEMO

Future Direction

- **Enhanced developer experience** enabling “non-data scientists” to follow a simple workflow for setting up and interacting with LLMs via microservices

- Implement a **seamless workflow** for transitioning from a local development environment to a production grade environment

- **End-end tooling**/framework for setting up LLMs locally for various applications such as text generation, document search, RAG applications etc

Resources

- **GitHub repository:** https://github.com/redhat-et/whisper-self-hosted-llm
- **Slides:** https://github.com/redhat-et/whisper-self-hosted-llm/tree/main/docs
- **HuggingFace models**: https://huggingface.co/ggerganov/whisper.cpp
- **AI Lab recipes with Podman:** https://github.com/containers/ai-lab-recipes

SCAN ME

✉ aduggal@redhat.com
hveeradh@redhat.com