What is RAG

- Rag is retrieval Augmentation Generation.
- Rag is used to query custom document which was not part of LLM Training.

Types of LLM Training or Data Feed.

- LLM Training with Large Corpus of data.
- Model fine tuning with Lora or Qlora
- One shot and few shot prompt training.
- RAG for using custom documents, which was not part of LLM Training.

Cost of training in ascending order

- Prompt training like one shot and few shot(No cost to least cost)
- RAG
- Finetuning with Lora and Qlora
- LLM training with Large Corpus of Data(Costliest)

How to choose between Rag and Lora

- If the data is not present in LLM but can be added and if the data is unchanging then choose Lora.
- If the data is always changing or could be different in different scenarios or customer then choose RAG.

Basic building blocks of RAG.

- Convert the document from PDF, docx, etc to text
- Split the converted text into chunk called as chunking.
- Generate the embeddings(or weights in simpler terms)
- Store the embeddings of documents in Vector Database.
- Get a query from the user to search.
- Generate an embedding of the user search.
- Do a similarity search of user search embedding with the document embedding and get those similar results.
- Feed those similar data to LLM.
- LLM will give the final output as it has now the knowledge of the custom documents.

Tools Used for RAG

- PDF reader like pypdf, pdfMiner, slate, PyMuPDF(Fitz) or doc reader.
- Any reader which can convert the documents to text.
- Vector database like FAISS, Chromadb, Mongodb, Singlestore, Pinecone, Weaviate and others.
- LLM toot chain like llama index or Langchain.
- LLM Models like openAI, llama, Mistral, gemini etc.

Some Famous algorithm for Vector similarity search

- Euclidean Distance
- Cosine Similarity
- K-nearest Neighbors
- Locality-Sensitive Hashing (LSH)
- ….Many More

RAG pipeline