RAG on Complex PDF using LlamaParse, Langchain and Groq

Retrieval-Augmented Generation (RAG) is a new approach that leverages Large Language Models (LLMs) to automate knowledge search, synthesis, extraction, and planning from unstructured data sources.

The RAG data stack consists of several key components:

- **Loading Data**: Initially, data is ingested from various sources, such as text documents, websites, or databases. This data can be in a raw or preprocessed format.
- **Processing Data**: The data undergoes preprocessing steps to clean and structure it for further analysis. This may include tasks like tokenization, stemming, and removing stop words.
- **Embedding Data**: Each piece of data is converted into a numerical representation called an embedding. This embedding captures semantic information about the data, making it easier for the LLM to understand and process.
- **Vector Database:** The embeddings are stored in a vector database, which allows for efficient retrieval based on similarity metrics. This database enables quick access to relevant data points during the generation process.
- **Retrieval and Prompting:** During the generation process, the LLM can retrieve relevant data points from the vector database based on the context of the current input. This retrieval mechanism helps the LLM provide more accurate and contextually relevant outputs.

What is Groq ?

Groq, founded in 2016 and headquartered in Mountain View, California, is an AI solutions startup focused on ultra-low latency AI inference. The company has made significant advancements in AI computing performance and is a notable participant in the AI technology sector.

Groq has registered its name as a trademark and has assembled a global team dedicated to democratizing access to AI

Groq's **Language Processing Unit (LPU)** is a cutting-edge technology designed to significantly enhance AI computing performance, especially for Large Language Models (LLMs).

The primary goal of the Groq LPU system is to provide real-time, low-latency experiences with exceptional inference performance.

One of Groq's achievements includes surpassing the benchmark of over 300 tokens per second per user on Meta AI's Llama-2 70B model, which is a significant advancement in the industry.

The Groq LPU system is particularly notable for its ultra-low latency capabilities, which are crucial for supporting AI technologies.

It is specifically tailored for sequential and compute-intensive GenAI language processing, outperforming traditional GPU solutions. This makes it highly efficient for tasks such as natural language creation and understanding.

At the core of the Groq LPU system is the first-generation GroqChip, which features a tensor streaming architecture optimized for speed, efficiency, accuracy, and cost-effectiveness. This chip has set new records in foundational LLM speed, measured in tokens per second per user, surpassing existing solutions in the market.

Groq has ambitious plans to deploy 1 million AI inference chips within two years, showcasing its dedication to advancing AI acceleration technologies.

What is the LPU Inference Engine ?

An LPU Inference Engine, with LPU standing for Language Processing Unit™, is a new type of end-to-end processing unit system that provides the fastest inference for computationally intensive applications with a sequential component to them, such as AI language applications (LLMs).

Why is it faster than GPUs ?

The LPU is designed to overcome the two LLM bottlenecks: compute density and memory bandwidth. An LPU has greater compute capacity than a GPU and CPU in regards to LLMs. This reduces the amount of time per word calculated, allowing sequences of text to be generated much faster. Additionally, eliminating external memory bottlenecks enables the LPU Inference Engine to deliver orders of magnitude better performance on LLMs compared to GPUs.

Getting Started with Groq

Right now, Groq is providing free-to-use API endpoints to the Large Language Models running on the Groq LPU — Language Processing Unit.

Groq guarantees to beat any published price per million tokens by published providers of the equivalent listed models. Other models, such as Mistral and CodeLlama, are available for specific customer requests.

To get started, visit this page and click on login.

What is Langchain?

LangChain is an open-source framework designed to simplify the creation of applications using large language models (LLMs). It provides a standard interface for chains, lots of integrations with other tools, and end-to-end chains for common applications.

Langchain Key Concepts:

- Components: Components are modular building blocks that are ready and easy to use to build powerful applications. Components include LLM Wrappers, Prompt Template and Indexes for relevant information retrieval.
- Chains: Chains allow us to combine multiple components together to solve a specific task. Chains make it easy for the implementation of complex applications by making it more modular and simple to debug and maintain.
- Agents: Agents allow LLMs to interact with their environment. For example, using an external API to perform a specific action.