

**User:**

Hello

**Assistant:**

Hello! Thank you for taking the time to speak with me today about the Entry-Level RAG AI Engineer role. I'd like to ask you a few questions to learn more about your experience and skills. Could you start by telling me about your experience with retrieval-augmented generation (RAG) pipelines?

**User:**

Sure, I have two projects under me built with RAC. One of the projects is the one I'm currently working on, that is an interview evaluation system. For this system, I'm using RAC specifically to verify the accuracy of the candidates' answers, since most of the LLMs have been trained on data till 2022. And to have relevant up-to-date data, I'm using RAC to fetch the current information from certain websites and add it to the context window of the LLM, so the LLM has up-to-date knowledge of the topics being discussed. Apart from this, I also use RAC in conjunction with an LLM to fetch general up-to-date information, so that we can use the intelligence of the LLM combined with the database that is available from the World Wide Web.

**Assistant:**

Thank you for sharing that information. It's interesting to hear about your current project. Can you tell me more about your experience with different large language models (LLMs)? Have you worked with or compared models like OpenAI's GPT, Llama, or Claude?

**User:**

Yes, I've worked with all three. OpenAI's ChatGPT, Llama, and Claude. Among the three, I found Claude's latest iteration of Sonnet, that is Claude Sonnet 3.5, to be the most accurate. That is, it can generate code within one shot, that means in the first attempt. Even complicated codes such as UI generation, which OpenAI's ChatGPT fails to do in one attempt, and Llama is even further behind. Even for textual reasoning, I find Claude gives a more comprehensive summary and understands my logical reasoning better than OpenAI's GPT. However, OpenAI has certain advantages that it can parse through documents really quickly and it has a multimodal system, that means I can upload text, upload audio files, upload images, and it can read it very well. And GPT also has the advantage of having cut down models, that is GPT 4.0 mini, as opposed to GPT 4.0, which is more cost efficient and also is faster. Right now Claude 3.5 Sonnet does not have that. Llama, on the other hand, is an open source implementation of an LLM, which can be locally run, so that comes with its advantages, that it can be locally run, it can be fine tuned, and we do not have to pay money for it. So each of the three LLMs has its pros and cons.

**Assistant:**

Thank you for that detailed comparison. You've mentioned some experience with Python in your projects. Could you elaborate on your experience with Python, particularly in the context of machine learning frameworks like TensorFlow or PyTorch?

**User:**

Yes, sure. I have implemented several machine learning projects using Keras which is implemented on top of TensorFlow. Some of the projects that I worked on are building a convolution neural network to classify the minced fashion set and the minced number set. Also, I have worked on beating a game called Chunking with Neat which is new revolution of augmented topologies which is a package in and of itself which uses genetic algorithms that is it uses a vast set of species which is the AI agent in this case and reproduces amongst them to find out the with each generation to find out the best solution. Apart from this, I have also used Keras in designing recurrent neural networks and LSTMs for prediction tasks and currently I am using the open source version of Whisper which is built on top of PyTorch to convert speech to text and speech to text in any language.

**Assistant:**

Thank you for sharing your experience. One last question: How familiar are you with AWS services, particularly in a production environment?

**User:**

I have deployed applications in AWS before. AWS mainly is used for scalability purposes, which it automatically creates scalable deployments from as and when required, as and when the traffic is increased when we deploy a model, not just a model, any application on AWS. Apart from this, I also have experience in uploading Docker image files to AWS, so that as and when required, the Docker image can automatically be used to run an application and then scale the application accordingly.

**Assistant:**

Thank you for your time. We'll be in touch soon regarding the next steps in the interview process.