Brance Founding ML Engineer Task

Name: Harsh Gautam

Linkedin Profile: https://www.linkedin.com/in/harsh-gautam-96276828

Date Challenge Received: 5th July 2023

Date Solution Delivered: 9th July 2023

1. Problem Statement

Design and host an AI Q&A App, which can answer queries provided as per “knowledge document provided”. Rest API should be hosted on cloud based infra which can handle 1M > request per day. Also, it should use some vector DB. Moreover, RAG based module is a plus.

2. Approach

Assumptions: Asssuming the question asked by user can fit the lambda/Databricks API max character limit.

I have created the code based on kedro (ML framework). It has three pipelines.

a) create\_embeddings pipeline →This pipeline will read the given knowledge document, create embeddings and save them as a csv dataset. ( In actual prod based scenario, this can be saved as a parquet file on S3) ( batch ).

b) upsert pipeline → This pipeline will read the embed dings file and upsert the embeddings to Pinecone Vector DB. ( batch )

c) qa pipeline → This pipeline will read the json payload from Lambda and give the output. ( Near Real-time)

3. Solution

Solution will have three parts:

a) Payload will go AWS SQS Service.

b) From SQS, lambda will be triggered and send the request to Databricks API.

c) On Databricks High Concurrency cluster, project wheel file will be installed and it will process the payload and return the output json.

Why:

a) Used Pinecone Vector DB because I have used other services which cloud manged as well, ( Ex. SQS, Lambda, API gateway & Databricks).

Flowchart:

AWS API Gateway

Lambda

SQS

Databricks API

OpenAPI

Pinecone DB

4. Future Scope

a) If let’s say payload is larger than the max char limits of Lambda/SQS/Databricks API, in that case payload read/write happen via S3. And request payload will only have message-id and S3 path.

b) Lambda code can be written to check the status of each request and stop processing of duplicate requests.

c) SNS/Bridge service with AWS load balancers can be introduced as the scale( no of requests) of App increases along with number of SQS queues and corresponding Lambda.