Technical Interview Case – Consigli

Situation: I am an analyst specializing in the analysis of Annual Reports within the automotive sector. My daily tasks involve extracting key financial metrics such as revenue, EBITDA, growth numbers, and conducting comparative analyses across different car companies, as well as comparing them to other sectors. To do this efficiently, I need a tool that automates the extraction and analysis process from reports.

Functional Requirements:

I would like to see a chatbot-like application built using a Retrieval-Augmented Generation (RAG) model with the provided data. The application should allow me to quickly obtain answers to questions like the examples below. While creating a chatbot interface is optional, the functionality can be demonstrated effectively using a console or terminal-based output.

- What was BMW's total revenue in 2023?
- How much revenue did Tesla generate in 2023?
- What was Ford's revenue for the year 2020?
- Can you provide the revenue figures for BMW in 2017?
- What key economic factors influenced Ford's performance in 2021?
- Which Tesla product is currently in the development stage?
- What were BMW's profit figures for 2020 and 2023?
- Between Tesla and Ford, which company achieved higher profits in 2022?
- What were Tesla's profit numbers for 2022 and 2023?
- Which company recorded better profitability in 2022 overall?
- Provide a summary of revenue figures for Tesla, BMW, and Ford over the past three years.
- What were the growth trends for BMW's financial performance from 2020 to 2023?

The application must support follow-up questions based on the answers provided (if you are unable to implement it fully, you can explain your ideas during the interview).

Requirements: (Choose only one approach)

- The application should be implemented as a simple web application packaged as a Docker image.
- Alternatively, you may demonstrate a simple model within the terminal or console of your IDE.

Data:

A collection of documents has been provided for this purpose. These documents include the annual reports for three car companies: Tesla, BMW, and Ford. You can download the documents from the following link:

Interview Task Dataset

Deployment and Preparations:

Build the application before the interview and be prepared to demonstrate it during the interview.

- If you have access to a cloud environment (Azure, AWS, GCP), you can deploy it there. Otherwise, run it on your local machine.
- To run the model, you can either use your own GPT API KEY. If you do not have one, please send an email to rohan@consigli.no, and I will provide you with a temporary API KEY that is valid for 48 hours.

During the Interview:

Be prepared to:

- Demonstrate the application, showcasing as many questions above as possible.
- If you can't implement solutions for every question, it's okay. We're interested in understanding your thought process and approach.
- Answer general questions about the application, the Large Language Model (LLM), and the AI space in general.
- Discuss typical challenges related to these types of solutions, especially those associated with AI.

You can use whichever tools you feel most comfortable with.

FYI: Please share your solution code with us as GitHub repository link at least 2 hours before the interview schedule.