Intern Evaluation Use Case: Validating Python and Gen AI Skills

Introduction

In order to assess the capabilities of an intern in the realms of Python programming and Generative AI skills, we have devised two comprehensive use cases. These use cases revolve around handling large volumes of data and ensuring scalability. The intern is expected to demonstrate their proficiency in developing frameworks using Python and implementing Gen AI solutions for efficient data processing and retrieval.

Use Case 1: Mutual Fund Allocation Change Tracker

Objective: Develop a framework to track changes in mutual fund allocations over time and provide insightful parameters for users.

Description

Mutual fund holding data is available across various mutual funds such as Zerodha, HDFC, and Axis funds. The fund allocation changes monthly based on the fund manager's decisions. The intern needs to create a Python framework to manage this large dataset and offer parameter options for users to identify and gain insights into fund changes.

Action Points

- Develop a framework using Python to handle the large volume of mutual fund holding data.
- Provide parameter options for users to choose the fund name and date range.
- Generate insights on the changes in fund allocations between the given date range.
- Display changes by fund and by month if a multi-month date range is chosen.

Example

User Input:

- Fund Name: Zerodha Midcap Fund
- Date Range: Last 5 months

Expected Output:

- Funds that changed during the specified date range.
- Details of changes by fund and by month.

Illustration:

For instance, if the user selects a 5-month date range, the output should display the changes in fund allocations for each month within that range.

Use Case 2: Gen AI RAG-Based Solution for Incremental Data

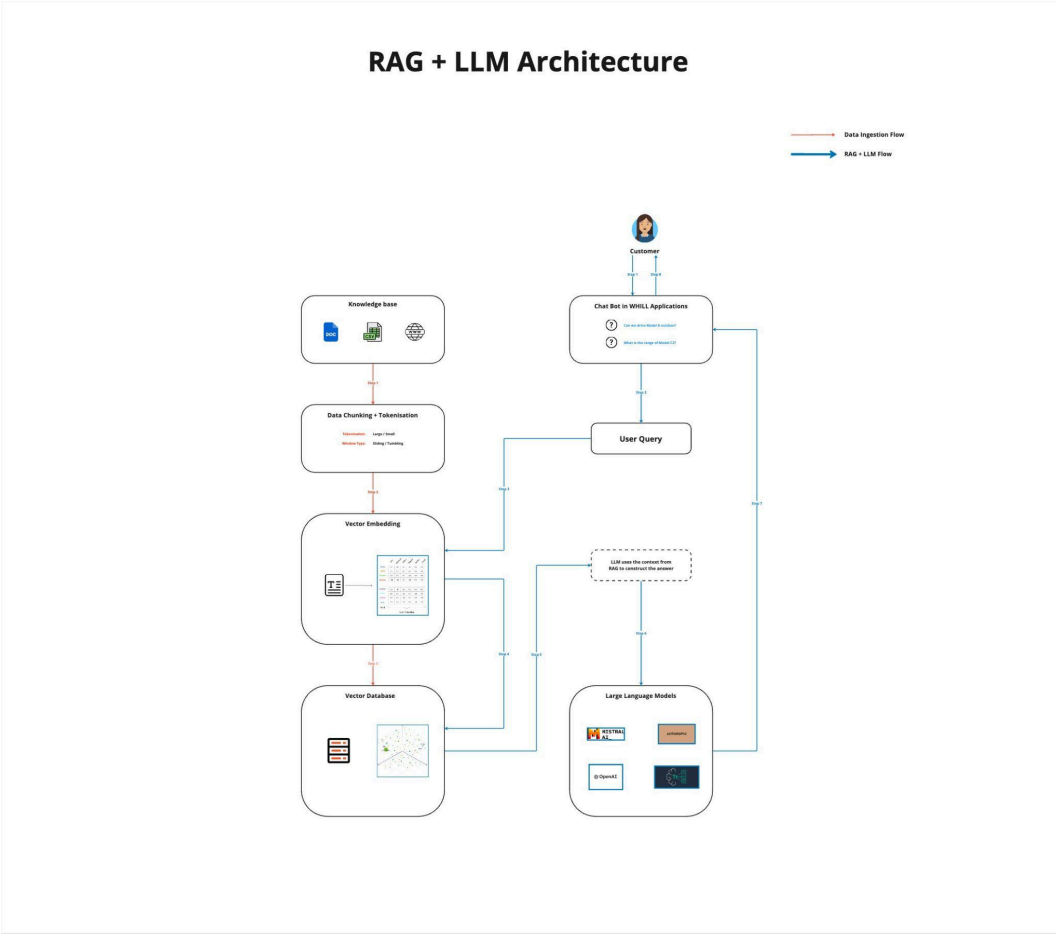
Objective: Implement a Gen AI RAG-based solution to enable users to query and fetch expected fund change percentages or allocations efficiently.

Description

Over time, the dataset of mutual fund holdings increases incrementally, making code management challenging. To address this, the intern is expected to implement a Gen AI Retrieval-Augmented Generation (RAG) based solution. This solution should allow users to query and fetch insights regarding fund changes. Implement the solution in AWS. You can easily create a AWS account by using your student email_id and can get access to AWS services for free for 12 months.

Action Points

- Use AWS Bedrock Knowledge Base and a vector database (Chroma DB or Pinecone) to store vectors.
- Implement an efficient retrieval mechanism to fetch the required data insights.
- Use any LLM & embedding model supported by AWS bedrock. Below is the data flow architecture for the RAG + LLM implementation
- You don’t have to implement any UI. AWS bedrock has playground, You can use it to show us the implementation, just record the video and send us the results.



Example Queries

- "Provide me fund change by % of allocation increase or decrease in the last 4 months for Zerodha Midcap fund."
- "Recommend the funds that exist in all the funds or exist in most of the funds, considering instruments like HDFC large cap, Zerodha large cap, Axis mid cap."

Expected Output

Query 1:

- Fetch the percentage change in allocation for the specified fund over the last 4 months and share the results.

Query 2:

- Recommend funds that consistently appear across multiple mutual funds.

Implementation Requirements

- Efficient storage and retrieval of large volumes of data using vector databases.
- Parallel query handling capability for approximately 20 users.
- Scalable solution to accommodate incremental data growth.
- Use AWS Bedrock with Cohere model as LLM and this should have an API call based approach to get the prompt from the user and provide the insights.

Evaluation Criteria

The following criteria will be used to evaluate the intern's performance:

- Technical Proficiency: Demonstrate sound knowledge of Python programming, data handling, and vector databases.
- Framework Development: Effectively develop a scalable and efficient framework for tracking mutual fund allocation changes.
- Gen AI Integration: Successfully implement a Gen AI RAG-based solution for querying and retrieving data insights.
- Problem-Solving Skills: Ability to address challenges related to large volume data processing and scalability.
- Innovation: Introduce innovative approaches to enhance the overall performance and user experience of the solution.

Conclusion

The use cases provided offer a comprehensive evaluation of an intern's Python and Gen AI skills. By developing a framework for tracking mutual fund allocation changes and implementing a Gen AI RAG-based solution for efficient data retrieval, the intern will demonstrate their technical proficiency, problem-solving abilities, and innovative thinking. This evaluation will serve as a valuable benchmark for assessing the intern's capabilities and readiness for more complex projects in the future.