Challenge Overview

Objective
Build a conversational AI (bot) that can answer a variety of questions about restaurants, their menus, and their ingredients, leveraging both an internal proprietary dataset and external public datasets. The bot should demonstrate:

1. **Retrieval-Augmented Generation (RAG)**:
    - The ability to fetch relevant context from a knowledge base (including the internal dataset + external sources) to enhance LLM responses.
    - Proper chunking, embedding, and indexing of documents.
2. **Text-to-LLM Pipeline**:
    - How queries are processed and fed to the LLM.
    - Prompt engineering, handling user context, and ensuring factual consistency.
3. **External Data Integration**:
    - Seamless combination of proprietary Menudata.ai assets and supplemental data sources (e.g., news articles, Wikipedia entries).

Provided Datasets

1. **Proprietary Restaurant Dataset (Structured CSV)**

Example:

Link to CSV's:

Ingredient Data
https://docs.google.com/spreadsheets/d/1yd4dXqJZLFCXFMS6oLYCtuBiY5VpUUGZFUWKv2ytc3M/edit?usp=sharing

Restaurant Data

restaurant_name	menu_category	item_id	menu_item	menu_description	ingredient_name	confidence
20 spot	no proof	24932147	"amaro" spritz	pathfinder amaro, tonic	pathfinder amaro	0.95
20 spot	no proof	24932146	"gin & tonic"	lyre's	gin	0.8
20 spot	no proof	24932145	amalfi spritz	lyre's	amalfi spritz	0.95

20 spot	no proof	24932145	amalfi spritz	lyre's	lyre's	0.8
20 spot	pet-nat & sparkling wine	24932165	athenais de beru, ‘love joy’, chardonnay		athenais de beru, 'love joy', chardonnay	0.9
20 spot	no proof	24932150	blood orange	san pellegrino	blood orange	0.9

External Datasets (Unstructured)

- Download any supplemental information that might be interesting to include

    - **Wikipedia**: For background info on ingredients, cuisines, or cultural/historical context around dishes.
    - **News Articles / Blogs**: For trending flavors, recent restaurant reviews, new ingredient popularity, etc.

User Stories & Queries the Bot Should Handle

Below are sample questions to illustrate the type of queries your bot should answer. The solutions should integrate data from both the internal dataset and external sources.

1. **Ingredient-Based Discovery**
    - *“Which restaurants in Los Angeles offer dishes with Impossible Meat?”*
    - *“Find restaurants near me that serve gluten-free pizza.”*
2. **Trending Insights & Explanations**
    - *“Give me a summary of the latest trends around desserts in San Francisco.”*
3. **Historical or Cultural Context**
    - *“What is the history of sushi, and which restaurants in my area are known for it?”*
        - (Requires pulling historical info from Wikipedia or other sources, plus relevant restaurants from the internal dataset.)
4. **Comparative Analysis**
    - *“Compare the average menu price of vegan restaurants in San Francisco vs. Mexican restaurants”*
        - (Requires some basic analytics on the structured data plus possibly external cost-of-living data or news for extra context.)
5. **Menu Innovation & Flavor Trend**
    - *“How has the use of saffron in desserts changed over the last year, according to restaurant menus or news articles?”*
        - (Might require time-series or trending data from your internal dataset, plus related culinary news references.)

Technical Requirements

1. **Data Ingestion & Indexing**
    - Demonstrate how you would ingest the proprietary restaurant dataset and unstructured external documents.
    - Use an embedding model to vectorize the text.
    - Store embeddings in a vector database (e.g., Pinecone, FAISS, Weaviate, or a similar solution).
2. **Prompt & Retrieval Pipeline**
    - Outline a retrieval pipeline that, given a user query, searches the vector database for relevant context.
    - Show how to *chunk* content (from news articles, Wikipedia, etc.) effectively so the LLM only sees relevant excerpts.
3. **LLM Integration & Prompt Engineering**
    - Incorporate the retrieved context into prompts sent to the LLM.
    - Demonstrate how you handle chain-of-thought or internal reasoning so that final user outputs are factual and concise.
    - Provide examples of system messages or intermediate prompts to direct the LLM's style and logic.
4. **Factual Consistency & References**
    - The system should show (or be able to show) references for the sources used in the generation (e.g., restaurant dataset record ID, relevant external article link, or a Wikipedia URL).
    - Handle the scenario where data is limited or conflicting (graceful fallback responses, disclaimers, or clarifications).
5. **Scalability & Future Extensions**
    - Consider how the pipeline would scale to larger datasets, real-time ingestion (e.g., new restaurants or menu items), and more frequent updates from external news sources.
    - Outline potential improvements such as incremental indexing, caching of embeddings, or real-time scraping.

Expected Deliverables

1. **Technical Design Document**
    - Description of the data ingestion workflow, retrieval approach, pipeline architecture, and prompt strategy.
    - Explanation of how relevant text is identified, chunked, embedded, and retrieved.
2. **Working Prototype or Code Samples**
    - Core scripts/notebooks showing how data is indexed into the vector store.

    - A minimal API or CLI-based demonstration where queries are processed and answered via the LLM.
3. **Sample Conversations / Queries**
    - Demonstration queries (from the user stories above) with successful outputs.
    - Show references or footnotes for where the system retrieved data.
4. **Optional Bonus**
    - Chatbot interface
    - Additional “analytics” style Q&A (e.g., top 5 trending ingredients, average price comparisons, etc.).

Final Note

This exercise is designed to mimic the kind of work Menudata.ai does: synthesizing proprietary menu data with global trends, cultural context, and real-time info from media sources.

After you are done with this small project, feel free to book a time here:

https://calendly.com/sunny-menudata/30min

Good luck!