Assignment 13: Financial Statement Question-Answering with a Retrieval-Augmented Generation (RAG) System

Name: Shiqi Hu

GTID: 904061372

0. Important Instructions for Running the Code

- In the system.py row 84, you must set up your together api key:
    - os.environ['TOGETHER_API_KEY'] = "Replace_with_Your_Own_TogetherAI_API_Key"

1. Processing Details

1.1 Data Organization

Original Data

- The original data was downloaded using the sec_edgar_downloader tool, which stores the SEC Form 10-K filings in a pre-defined hierarchical structure under ORIGINAL_DATA_DIR.

Original Directory Structure:

ORIGINAL_DATA_DIR/ sec-edgar-filings/ Ticker1/ 10-K/ FilingFolder1/ full-submission.txt

- Ticker1: The stock ticker symbol.
- 10-K: Directory containing 10-K filings for the respective ticker.
- FilingFolder1: Each subdirectory represents a single 10-K filing.
- full-submission.txt: The raw HTML content of the filing.

Processed Data

- The processed data is organized by ticker-year format in PROCESSED_DATA_DIR. Each 10-K filing is cleaned, and the essential content is stored as content.txt.

Processed Directory Structure:

PROCESSED_DATA_DIR/ Ticker1_YYYY/ content.txt

- Ticker1_YYYY: Combines the ticker symbol and the year of the filing.
- content.txt: Cleaned and processed content of the filing.

1.2 Data Processing Workflow

Step 1: Extraction of Text Content and Creation of Processed Directories

- Identifying the Filing Year: The filing year is extracted from the folder name of each filing using a regular expression pattern that identifies a two-digit year (e.g., -19- in the folder name translates to 2019).

- Creating Processed Directories: A new directory for each filing is created in PROCESSED_DATA_DIR, named using a combination of the ticker symbol and filing year (e.g., Ticker1_2019).
- Dynamic Path Construction: Paths for input (full-submission.txt) and output (content.txt) are dynamically constructed using Python's os.path module.
- Reading Files: The content of each full-submission.txt file is read using Python's open() method with UTF-8 encoding.
- Preprocessing: The raw HTML content is passed to the cleaning function for further processing and standardization.

Step2: Data Cleaning

The cleaning function (clean_10k_content) performs multiple steps to ensure the extracted content is free of unnecessary elements and organized for analysis:

- Remove non-printable characters: Special and control characters are removed using re.sub, leaving only printable ASCII characters.
- Extract main content: Content is trimmed to focus on meaningful sections using markers like < as splitting points to eliminate preamble or unrelated text.
- Remove standard noise elements: Using BeautifulSoup, unnecessary HTML tags (e.g., script, style, meta) are stripped to reduce clutter and focus on textual content.
- Remove SEC headers and footers: Patterns such as "UNITED STATES SECURITIES AND EXCHANGE COMMISSION" and "FORM 10-K" are identified and removed with regular expressions to eliminate boilerplate sections.
- Enhanced table processing: Only process tables with actual content, remove empty cells and strips unnecessary attributes from tags.
- Improve paragraph handling: Adds a newline at the end of each paragraph if not already present.
- Preserve important section headers: Extracts headers (e.g., "ITEM 1. BUSINESS") using regular expressions to facilitate structured navigation.
- Normalize whitespace: Collapses multiple spaces into single spaces. Condenses excessive newlines.
- Final cleanup: Splits sentences into paragraphs and removes redundant newlines.

Step 3: Saving Processed Content

- The cleaned and structured content is saved as content.txt within the corresponding directory in PROCESSED_DATA_DIR. Each directory is named using the combination of the ticker symbol and filing year (e.g., Ticker1_2019).

2. Vector Store Construction

2.1 Chunking Strategy

- Sentence Splitting: The text was initially divided into sentences using the SentenceSplitter utility. This ensured that chunks were formed from semantically complete units rather than arbitrary divisions.
- Fixed-Window Chunking with Overlap:
    - A fixed window of approximately 512 tokens was used to create chunks, balancing content size and embedding efficiency.
    - Overlap of 50 tokens was implemented to provide context continuity between consecutive chunks. This overlap mitigates issues of context loss at chunk boundaries, improving the embeddings' quality for vector search.
- Chunk Construction:
    - Sentences were added to a chunk until the token limit was reached. When the limit was exceeded:
    - The current chunk was finalized and added to the list.
    - The next chunk was initialized with overlapping sentences from the previous chunk.
- TextNodes: Each chunk was converted into a TextNode instance, which serves as the fundamental unit for storing in the vector database.

2.2 Document Embeddings

- Model Selection:
    - Embedding Model: sentence-transformers/all-mpnet-base-v2
    - This model provides high-quality sentence-level embeddings, making it ideal for representing individual chunks of text.
- Chunk-Level Embeddings:
    - Each chunk (generated in the chunking step) was passed through the embedding model.
    - The model encoded each chunk into a fixed-size vector (dense embedding), capturing the semantic meaning of the text.
- Metadata Augmentation:
    - In addition to the vector representation, each chunk was enriched with metadata:
        - Ticker Symbol: Identifies the company associated with the filing.
        - Year: Denotes the filing year.
    - This metadata allows for efficient filtering and retrieval during search operations.

Vector Database Construction

- Initialization: A persistent ChromaDB client was initialized with the specified VECTOR_STORE_DIR, ensuring that the database could be reused and updated across sessions.
- Collection Management: A collection named "financial_filings" was created within ChromaDB to store the embeddings and their associated metadata. Pre-existing collections with the same name were deleted to maintain consistency.

- Vector Store: The collection was integrated with the ChromaVectorStore, enabling efficient storage and retrieval of document embeddings.
- Vector Store Index: A VectorStoreIndex was built using the list of TextNodes, the ChromaVectorStore, and the embedding model. This index allows for optimized vector search queries.

Metadata Role

- Search Filters: Metadata fields (ticker and year) allow users to query specific companies or filing periods efficiently.
- Document Traceability: Metadata provides a direct link between embeddings and their original documents, enabling seamless navigation from search results to raw content.
- Enhanced Search Relevance: Metadata facilitates multi-dimensional queries, combining semantic similarity with metadata-based filtering for precise results.

Validation

After constructing the vector store, the following validation steps were performed:

- Node Count: Verified the total number of chunks (TextNodes) processed and stored.
- Database Contents: Retrieved a subset of stored documents and their metadata to ensure correctness.
- Document Previews: Displayed the first 200 characters of a few stored chunks to confirm content integrity.

3. RAG Baseline Implementation

3.1 Initialization

- Embedding Model:
    - Model: sentence-transformers/all-mpnet-base-v2
    - The embedding model generates dense vector representations for semantic similarity searches.
    - It is initialized during system setup and configured to leverage GPU acceleration if available. If GPU is unavailable, it defaults to CPU.
- Vector Store:
    - Database: ChromaDB
    - The vector database stores precomputed embeddings along with metadata for each document chunk.
    - During initialization:
        - A persistent ChromaDB client is connected to the database in VECTOR_STORE_DIR.
        - The relevant collection ("financial_filings") is loaded to enable query-based retrieval.
- Large Language Model (LLM)
    - Model: meta-llama/Llama-3-70b-chat-hf

    - The LLM is integrated using the together API, enabling powerful, conversational query response capabilities.
    - Together API setup:
        - The TOGETHER_API_KEY environment variable is used to authenticate with the Together platform. When user runs in his own system, he needs to replace with his own TOGETHER_API_KEY environment variable.

- Reranker

    - The reranker refines retrieved results based on query-context similarity, ensuring only the most relevant nodes are used for response generation.
    - Configurable parameters:
        - RERANKER_CHOICE_BATCH_SIZE: Number of nodes considered in each batch for reranking.
        - RERANKER_TOP_N: Number of top nodes returned after reranking.

3.2 Response Generation Pipeline

- Query Processing

    - The system takes three inputs:
        - Query: The user's question.
        - Ticker: The company symbol (e.g., AAPL for Apple).
        - Year: The year of the desired SEC filing.
    - The query is converted into a QueryBundle object to enable structured interaction with the retriever.

- Metadata Filtering: Metadata filters ensure retrieval focuses on documents relevant to the specified ticker and year.

- Retrieval

    - The retriever fetches the top RETRIEVER_SIMILARITY_TOP_K nodes from the vector database using the query and metadata filters.
    - Retrieved nodes are validated to ensure they match the specified metadata criteria.

- Reranking

    - The retrieved nodes are passed through the LLM-based reranker, which evaluates their relevance to the query.
    - The top reranked nodes (RERANKER_TOP_N) are selected for response generation.

- Response Generation

    - The 2 top nodes are combined to form the context for the LLM.
    - A prompt is constructed using the query and the selected context:
        - Based on the following context, answer the question: {query}
        - Context from {ticker}'s {year} report: {context_text}
        - Answer the question concisely in one sentence
    - The meta-llama/Llama-3-70b-chat-hf model generates a response based on the prompt by using together API.

- Retry logic ensures robustness in case of API failures.

4. Mannual Evaluation

4.1 Retriever Performance

Overall Performance

The retriever achieved an overall accuracy of ~90% (18/20 queries). It performed well in identifying and extracting highly relevant content but struggled with queries requiring synthesis or those reliant on specific data points that were not well-represented in the documents.

Highly Relevant Retrieved Nodes:

- MCHP Financial Metrics:
  - Retrieved exact revenue ($5,349.5M) and gross profit ($2,931.3M).
  - Included detailed quarterly breakdowns and year-over-year comparisons.
- MAR Risk Factors:
  - Comprehensive coverage of risk-related sections.
  - Detailed explanations of industry competition, economic uncertainty, and operational risks.
- SBUX Products/Revenue:
  - Detailed breakdown of revenue sources (e.g., beverages, food, packaged goods).
  - Clear segmentation by product and geographical distribution.

Moderately Relevant Retrieved Nodes:

- STT Competitive Advantages:
  - Relevant sections retrieved but mixed with other corporate information.
  - Some key competitive advantages required synthesis from multiple sections.
- MSI Investment Strategy:
  - Investment policy details were scattered across multiple sections.
  - Missing some context on decision-making processes for investments.

Poor Relevance Retrieved Nodes:

- SPG Capital Expenditures:
  - Retrieved nodes lacked specific capex figures.
  - Included unrelated financial data, diminishing relevance.
- STT Operating Income:
  - Sections did not clearly include operating income.
  - Mixed with other financial metrics, making extraction difficult.

Strengths

- Highly effective at retrieving numerical data and segmented financial metrics.
- Well-suited for broad financial overviews and risk-related information.

Weaknesses

- Struggled with narrowly defined queries requiring precise, single-section data.
- Challenges synthesizing information scattered across multiple sections.

4.2 Response Performance

Overall Performance

The system's response accuracy was ~90% (18/20 queries). It excelled in generating well-structured and precise answers when retrieval was strong but faltered in cases of incomplete or unclear retrievals

Excellent Responses:

- MCHP Revenue/Profit:
    - Precise extraction of $5,349.5M revenue and $2,931.3M gross profit.
    - Clear and accurate presentation of fiscal year data.
- SBUX Revenue Sources:
    - Detailed and structured breakdown of revenue streams.
    - Accurate percentages for each category (e.g., beverages 58%, food 17%).
- MAR Risk Factors:
    - Comprehensive synthesis of risks, logically categorized by competition, economy, and operations.

Adequate Responses:

- A Regional Operations:
    - Identified key markets but omitted market share details.
    - Accurate but lacked comprehensiveness.
- VRSK Debt Analysis:
    - Provided the correct total debt figure but lacked deeper contextual details about composition and implications.

Weak Responses:

- SPG Capital Expenditures:
    - Unable to provide specific capex figures due to weak retrieval.
    - Response highlighted data limitations but lacked value.
- PH Shareholder's Equity:

- Misinterpreted retrieved data, leading to an incorrect figure.
- Mixed fiscal year data added to confusion.

Strengths

- Numerical Precision: Demonstrates consistent accuracy in extracting and presenting key figures, such as revenues, profits, and other financial metrics. For instance, data points like MCHP's revenue ($5,349.5M) and gross profit ($2,931.3M) were presented with exact values and correct associations, ensuring clarity and reliability in financial reporting.
- Contextual Insights: Goes beyond surface-level data by providing meaningful context for the retrieved figures, such as linking risk factors to broader industry trends or breaking down revenue sources into detailed categories like product lines and geographical regions.
- Recognition of Data Gaps: Acknowledges when specific information is missing or incomplete, such as explicitly stating the absence of detailed capital expenditures for SPG.
- Clear Organization: Maintains a logical and reader-friendly structure by grouping related information coherently. For example, responses like SBUX's revenue sources were organized by product categories (beverages, food, packaged goods) with corresponding percentages, ensuring clarity and usability of the information provided.

Limitations

- Dependence on Retrieval: The quality of responses heavily depends on the quality of the retrieved data. When the retrieval process brings back irrelevant or incomplete information, the responses lack accuracy and depth. For example, STT's operating income query mixed unrelated financial metrics, weakening the response's relevance.
- Temporal Confusion: Some responses misinterpret fiscal year data versus calendar year data, leading to inaccuracies or inconsistencies. For instance, financial metrics retrieved from different time periods might be conflated, resulting in a lack of temporal alignment in analysis.
- Limited Depth: Responses often focus on summarizing retrieved data rather than providing deeper insights or implications. While adequate for simple queries, this limitation becomes evident in queries requiring analysis or synthesis, such as investment strategies or comprehensive financial trends.
- Generalizations: Responses to queries with insufficient data sometimes resort to overly broad or vague conclusions. For example, SPG's capital expenditures query returned generalized financial information without addressing the specific capital expenditure figures.

4.3 Query Results (saved in query_results.csv)

	ticker	query	year	response	retrieved nodes
0	MCHP	What is the total revenue of this company for ...	2019	The total revenue of this company for this fis...	Note 21.\nQuarterly Results (Unaudited) The fo...

	ticker	query	year	response	retrieved nodes
1	MCHP	What is the gross profit of this company for t...	2019	The gross profit of this company for this fisc...	Note 21.\nQuarterly Results (Unaudited) The fo...
2	MAR	Who is the Chairman of the Board of this company?	2015	The Chairman of the Board of this company is J...	Carl T.\nBerquist Executive Vice President and...
3	MAR	What are the main risk factors of this company?	2015	The main risk factors of this company include ...	14 Table of Contents Item 1A.\nRisk Factors.\n...
4	SBUX	What are the main products of this company?	2017	The main products of this company are coffee b...	We believe, based on relationships established...
5	SBUX	What are the main revenue sources of this comp...	2017	The main revenue sources of this company are c...	Company-operated store revenues are reported n...
6	VRSK	What is the company's total debt for this fisc...	2010	The company's total debt for this fiscal year ...	Debt: The following table presents short-term ...
7	VRSK	What are the primary costs affecting the compa...	2010	The primary costs affecting the company's prof...	There was also an increase in office maintenan...
8	MSI	What is the company's cash flow from operation...	2013	The company's cash flow from operations for th...	This reference is included to help users trans...
9	MSI	What is the company's investment strategy ment...	2013	The company's investment strategy is to invest...	font-size:10pt;">Within the equity securities ...
10	A	What are the key markets or regions where this...	2019	The key markets or regions where this company ...	The following table presents summarized inform...
11	A	What are the company's earnings per share (EPS...	2019	The company's earnings per share (EPS) as repo...	During the year ended October 31, 2018 , cash ...
12	STT	What are the company's primary competitive adv...	2018	State Street's primary competitive advantages ...	; • our ability to create cost efficiencies th...

	ticker	query	year	response	retrieved nodes
13	STT	What is the company's operating income for thi...	2018	The company's operating income for this fiscal...	Depletion and Amortization, Property, Plant, a...
14	PH	What is the total asset of this company for th...	2017	The total asset of this company for this fisca...	(a) Includes an investment in a joint venture ...
15	PH	What is the total shareholder's equity of this...	2017	The total shareholder's equity of this company...	30, 2017 Jun. 30, 2016 Jun. 30, 2015 Schedule ...
16	SRE	What is the net income of this company for thi...	2015	The net income of this company for the fiscal ...	SAN DIEGO GAS & ELECTRIC COMPANY CONSOLIDATED ...
17	SRE	What is the fiscal year with year end of this ...	2015	The fiscal year with year end of this 10K repo...	As provided in Internal Revenue Notice 2007-86...
18	SPG	What are the company's main subsidiaries menti...	2015	The company's main subsidiaries mentioned in t...	Disclosure - Quarterly Financial Data (Unaudit...
19	SPG	What are the company's capital expenditures fo...	2015	The company's capital expenditures for the rep...	font-family:Times New Roman;font-size: 10pt"> ...

5. Streamlit App