

# Assignment 2

16/03/2025

# 10 Points Possible

Attempt 1



In Progress

**NEXT UP: Submit Assignment**

Add Comment

## 1 Attempt Allowed

17/02/2025 to 16/03/2025

### ▼ Details

## Assignment 2 - RAG Chatbot

Develop a **Retrieval-Augmented Generation (RAG)** model to answer financial questions based on company financial statements (last two years).

### IMPORTANT

- **Use Only Open-Source Embedding Models**
- **Use a Small Open-Source Language Model (SLM) for Response Generation** (No proprietary APIs)
- **Implement One Guardrail** (Either Input-side or Output-side)
- **Develop an Application Interface** (Web-based, CLI, or GUI)
- Each Group Implements a Specific Advanced RAG Technique based on the group number

### Advanced RAG Techniques

1. Multi-Stage Retrieval
2. Chunk Merging & Adaptive Retrieval
3. Re-Ranking with Cross-Encoders
4. Hybrid Search (Sparse + Dense Retrieval)
5. Memory-Augmented Retrieval

Steps to choose Advanced RAG Technique:

For any Group N, the option assigned is:

$(N \bmod 5) \rightarrow \text{Option Number}$

- If the remainder is 1, assign Option 1.
- If the remainder is 2, assign Option 2.
- If the remainder is 3, assign Option 3.
- If the remainder is 4, assign Option 4.
- If the remainder is 0, assign Option 5 (since multiples of 5 should get the last option).

Example, for Group 24:


1.  $24 \bmod 5 = 4$
2. Technique Index 4 → Hybrid Search (Sparse + Dense Retrieval)

## Tasks

Component	Details
1. Data Collection & Preprocessing	Download the last two years of financials (Use any one group member's company earning statements, if nothing is available use any company's data freely available). Clean and structure the data for retrieval.
2. Basic RAG Implementation	Implement a simple RAG model: <ul style="list-style-type: none"><li>- Convert financial documents into text chunks.</li><li>- Embed using a pre-trained model</li><li>- Store and retrieve using a basic vector database</li></ul>
3. Advanced RAG Implementation	Improve retrieval by: <ul style="list-style-type: none"><li>- Using BM25 for keyword-based search alongside embeddings.</li><li>- Testing different chunk sizes &amp; retrieval methods for better accuracy.</li><li>- Implementing re-ranking.</li></ul>
4. UI Development (e.g., Streamlit)	Build an interactive UI: <ul style="list-style-type: none"><li>- Accept user queries.</li><li>- Display answer &amp; confidence score.</li><li>- Ensure clear formatting &amp; responsiveness.</li></ul>
5. Guard Rail Implementation	Implement one guardrail: <ul style="list-style-type: none"><li>- Input-Side: Validate and filter user queries to prevent irrelevant/harmful inputs.</li><li>- Output-Side: Filter responses to remove hallucinated or misleading answers.</li></ul>
6. Testing & Validation	Ask 3 test questions: <ul style="list-style-type: none"><li>- A relevant financial question (high-confidence).</li><li>- A relevant financial question (low-confidence).</li><li>- An irrelevant question (e.g., "What is the capital of France?") to check system robustness.</li></ul>

## Submission Requirements

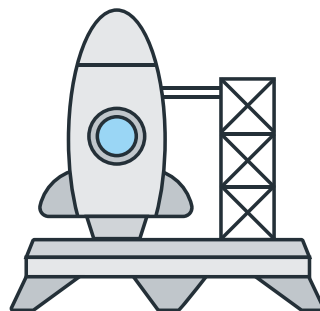
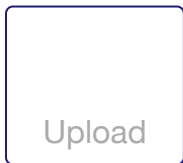
- Each group will submit a zip file.
- The zip file should contain 2 files:
  - Share the **Python Notebook** (the implementations of basic and advanced RAG should be clearly highlighted and explained in comments). This file can be in .py or .ipynb format.
  - A PDF file containing **3 screenshots of the implementation** of the above mentioned questions (this should include the question asked, the answer and the confidence score for

all the 3 questions mentioned above). You are free to use any relevant evaluation metric for confidence score. The PDF file should also include the **hosted app link** (e.g., if you prefer Streamlit, here is how you can host a Streamlit app for free <https://blog.streamlit.io/host-your-streamlit-app-for-free/>  (<https://blog.streamlit.io/host-your-streamlit-app-for-free/>)).

- Each group should submit one zip file only. The names of all the files and the zip file should start with the group number (e.g., Group\_XX\_<filename>.<filetype>).

Keep in mind, this submission will count for everyone in your Assignment Groups group.

### Choose a submission type.



Choose a file to upload

or

 Webcam Photo

 Canvas Files