94-844 Generative AI Lab

Prof. Beibei Li
Heinz College
Carnegie Mellon University

ASSIGNMENT 2 – Retrieval Augmented Generation (RAG) with LLMs

Retrieval Augmented Generation (RAG) is the process of creating and using custom data-sets as contextual information to include in the prompts submitted to LLMs (or in use with other types of ML models) in order to improve the quality of information provided in response to user queries. Building a high-quality RAG system involves multiple data engineering steps, and design considerations. These steps include: (i) breaking up text data (or other media formats) into smaller units of measurement (in NLP, this is often called 'chunking'); (ii) tokenizing these units; then (iii) transforming the tokens into numerical representations – these are called embeddings; (iv) creating a special database that can handle storing high-dimensional data like embeddings – this type of database used in a RAG is called a vector store; and (v) incorporating an LLM or other machine learning model into this workflow at one or more points. To achieve high-quality output from a RAG system, developers must experiment with the multiple ways of building a RAG. For example, how a text dataset is broken up into chunks (or units of measurement) will affect what the embeddings for the text represent, and thus, what a user can retrieve from the store. For instance, breaking a text document up into words would enable the RAG to retrieve individual words most similar to those in a user question, whereas chunking the document semantically would enable the RAG to potentially retrieve more or any of the text dataset that is associated semantically with the user question.

This assignment is designed to familiarize you with the different components and features of RAG that you need to conduct data-driven experiments to inform how to design the RAG system. Toward this objective, you will be provided with a dataset; you will need to choose a method of breaking up the dataset, tokenizing it, creating embeddings from the tokens, and then uploading them to a vector store. After this, you will have the basic components of a RAG system build (minus an LLM or ML model). To help you learn how different choices you've made in designing these system components might affect the quality of information retrieval when using the RAG, you will submit various queries to the vector store (e.g. embedding database), and analyze the relevance of the output from the vector store. Next, you will experiment with changing parameters in your user query submitted to the store, and analyze how that affects the quality of the output. Next we will have you reconstruct the vector store after changing the way you break up the text dataset into units of measurement; the purpose of doing this is to observe how your choice of 'chunking strategy' affects the output. Optionally, you may earn extra points by completing bonus tasks that will have you utilize each of the main components of a RAG system (text chunking (and tokenizing); embeddings; vector store / similarity metrics; LLM generation; output evaluation) by crafting and changing the parameters of calls to LLMs, and evaluating the LLM responses. Ultimately, the goal of this assignment is to give you a detailed understanding of how to conduct data-driven experiments to inform how you design a RAG system, and instill the confidence to use these data analytics skills in everyday practice.

Homework Submission FAQs:

1. ***Am I required to submit any Python code or the Python notebook I use for the assignment?*** You are not required to submit your Python notebook or code. If you do submit your Python code, the code will not be reviewed or graded.

2. ***What if the quality of information retrieved from the vector store is really low quality – will this affect my grade?*** No. We are not grading you based on the quality of the output retrieved from the vector store. You will be graded on following the homework instructions – which include copying/pasting the output retrieved from the vector store into a Google spreadsheet (provided below and on Canvas). You will also be graded on the quality of your written analyses (provided in response to the homework questions). In your analyses, our hope is you will display strong reasoning skills in thinking about how different features of the RAG design might affect the quality of information retrieval. *Additional information* about the **Grading rubric for this assignment** is provided in the **last page of this document.**

3. ***What is the required format for the homework submission?*** You are required to submit a Google document or PDF of your written responses to the homework questions. *Please make sure to grant Prof. Li and TAs access to your Google document *before* submitting it.*

Materials Required in your homework submission:

1. You are required to submit a Google document or PDF of your written responses to the homework questions.
    a. *Please make sure to grant Prof. Li and TAs access to your Google document *before* submitting it.*

2. You are required to copy this Google spreadsheet template and provide copies of the responses to your queries that were submitted to the vector store. https://docs.google.com/spreadsheets/d/1T2_fV0LKw9s_BEDKAvYZKKugDD0iBKSUlsnaui3Dmlg/edit?gid=0#gid=0

    a. *Please make sure to grant Prof. Li and TAs access to your Google document *before* submitting it.*

A. Experimenting with Vector Store Query Design (50 points)

Python coding, Data Engineering & Experimental Design Instructions:

1. **Data Selection & Engineering:**
    a. Upload the CMU Student Handbook to the Python notebook.
    b. Decide how you will 'chunk' the text data (e.g. at the word, sentence, paragraph level), and then chunk the CMU Handbook.
2. **Choose a Similarity Metric to Use in the Vector Store** (e.g. cosine, dot product, euclidean distance).
3. **Choose an embedding model.**
4. **Create embeddings for the text datasets** (CMU handbook).
5. **Create a vector store** (follow the steps in the Python notebook provided in Lab tutorial video and Lab session).
6. Load the embeddings into the Vector Store.
7. **Experiment with Query Parameters & Queries in the Vector Store.**
   In this part of the homework, we will experiment with submitting pre-defined queries, provided below, to the vector store, and recording the responses to the queries.
    a. **Requirement**: Specifically, use the 5 queries below to query the vector store, and retrieve documents.
    b. **Requirement**: Record the information retrieved from using the queries in this Google spreadsheet: https://docs.google.com/spreadsheets/d/1T2_fV0LKw9s_BEDKAvYZKKugDD0iBKSUlsnaui3Dmlg/edit?usp=sharing
    c. **Required Queries and K parameter setting:** Submit these 5 queries to the vector store – setting the 'k' parameter in the query function to 5 – and **record the responses in the Google spreadsheet (linked to above). Submit this spreadsheet to Canvas along with your PDF responses to the homework questions**.
        i. **Query 1**: What is the policy statement for the academic integrity policy?
        ii. **Query 2**: What is the policy violation definition for cheating?
        iii. **Query 3**: What is the policy statement for improper or illegal communications?
        iv. **Query 4**: What are CMU's quiet hours?
        v. **Query 5:** Where are pets allowed on CMU?

Homework Questions – Part A:

A.I. Required: Explain your rationale for choosing the similarity metric you decided to use in the vector store.

Specifically, please provide answers to the following questions in your homework submission:

- What is one advantage of using the metric you chose?
- What is one difference between using the metric you selected and the other similarity metrics? (e.g., cosine, dot product, and euclidean similarity metrics).

A.II. Required: Copy and paste the results or information retrieved from the vector store in response to each of the queries you submitted to the vector store into this spreadsheet: https://docs.google.com/spreadsheets/d/1T2_fV0LKw9s_BEDKAvYZKKugDD0iBKSUlsnaui3Dmlg/edit?gid=0#gid=0

A.III. Required: Qualitatively analyze the responses to your queries submitted to the vector store.

Specifically, please provide answers to the following questions in your homework submission:

- Did the queries retrieve the information you were expecting to obtain? Why or why not?
- Why do you think the queries were successful / unsuccessful in retrieving the information you expected or needed?
- What features or components of the vector store design do you think affected the success or failure of retrieving information that was relevant to your queries to the vector store?

B. Vector Store Embeddings & Query Parameter Experiments (50 points)

Python coding, Data Engineering & Experimental Design Instructions:

1. **Required: Conduct the following experiment:**
    a. Choose 1 of the 5 queries in A.7, above;
    b. Conduct an experiment by repeatedly submitting the same query that you selected in B.!.A (or the preceding step) to the vector store; but each time you submit the query, change the search parameter, in the following manner:
        i. 1st query submission to the vector store: set the k parameter to be k=1.
        ii. 2nd query submission to the vector store: set the k parameter to be k = 3
        iii. 3rd query submission to the vector store: set the k parameter to be k = 5
        iv. 4th query submission to the vector store: set the k parameter to be k = 10

2. **Required: Record the *unique* responses to each query in this spreadsheet:** *(you do **not** need to record duplicate output. For example, when you change k=1 to k=3, one of the responses should be the same as the response you got when you set k=1)* https://docs.google.com/spreadsheets/d/1T2_fV0LKw9s_BEDKAvYZKKugDD0iBKSUlsnaui3Dmlg/edit?gid=0#gid=0

3. **Required:** Return to step A.1.B., above, and select a different text chunking method (e.g. word, sentence, paragraph). You can choose any chunking method as long as it is not the same method you used previously.
    a. Chunk your text data using the method.
    b. Tokenize the chunks.
    c. Create embeddings of the tokenized chunks.
    d. Load the embeddings into the vector store.
    e. Submit the same query you selected in B.1, and submit it to the vector store 4 different times (using the different 'k' parameter settings defined in B.1 (i-iv), e.g. the preceding homework question). **Record the *unique* responses to each query in this spreadsheet:** https://docs.google.com/spreadsheets/d/1T2_fV0LKw9s_BEDKAvYZKKugDD0iBKSUlsnaui3Dmlg/edit?gid=0#gid=0

Homework Questions - Part B:

B.I. Required: Explain your rationale for selecting the query you choose in B.1.

Please provide answers to the following questions in your homework submission:

- Why did you choose this query vs. the other queries?

B.II. Required: Copy and paste the responses to the queries you submitted to the vector store, as described in B.1 above, here: https://docs.google.com/spreadsheets/d/1T2_fV0LKw9s_BEDKAvYZKKugDD0iBKSUlsnaui3Dmlg/edit?gid=0#gid=0

B.III. Required: Copy and paste the responses to the queries you submitted to the vector store, as described in B.2 above, here: https://docs.google.com/spreadsheets/d/1T2_fV0LKw9s_BEDKAvYZKKugDD0iBKSUlsnaui3Dmlg/edit?gid=0#gid=0

B.IV. Required: In observing the responses from the vector store to the queries created in B.1.:

Please provide answers to the following questions in your homework submission:

- Which 'k' parameter do you think retrieved the highest quality / best result?
- Why do you think this parameter was the best to use with the query?
- What would you recommend doing to improve the quality of the results?

B.V. In observing the responses from the vector store to the queries created in B.2.:

Please provide answers to the following questions in your homework submission:

- Which 'k' parameter do you think retrieved the highest quality / best result?
- Why do you think this parameter was the best to use with the query?

- What would you recommend doing to improve the quality of the results?

BONUS TASKS / QUESTIONS. Evaluation of LLM Response Generation with System Prompts Retrieved from Vector Store (20 points)

Bonus Task Instructions:

FAQs:

- ***Where can I find instructions about setting up my Python environment?*** Please refer to the instructions PDF and Lab lecture recording for details about how to set up the Python notebook environment**.**

- ***Where can I find instructions about accessing different LLMs on HuggingFace?*** Please refer to the instructions PDF and Lab lecture recording for details about accessing LLMs via HuggingFace.

Dataset Requirement:

- **What dataset should I use?** For the below tasks, please retrieve information from the vector store that contains the CMU Student Policy Handbook, e.g. the vector store you created for the homework assignment.

Python coding, Data Engineering & Experimental Design Instructions:

Bonus.A. Using the queries created in B.1., submit the queries to 2 other LLMs (can be any LLM but they must be different LLMs) using the rag_llm() function in the Python notebook. ***Record the responses to the queries in your homework submission.***

Bonus.B. Using the queries created in B.2., submit the queries to 2 other LLMs (can be any LLM but they must be different LLMs) using the rag_llm() function in the Python notebook. Record the responses to the queries. ***Record the responses to the queries in your homework submission.***

Bonus.C. Run the responses generated in **Bonus.A** (above) through the 'compare_text_similarity' and 'compare_text_similarity_response2context' functions (provided in the Python notebook). Record the metric results. ***Record the responses to the queries in your homework submission.***

Bonus.D. Run the responses generated in **Bonus.B** (above) through the 'compare_text_similarity' and 'compare_text_similarity_response2context' functions

(provided in the Python notebook). Record the metric results. ***Record the responses to the queries in your homework submission.***

Bonus.E. Research and choose 3 additional evaluation methods/metrics to use for analyzing the RAG retrieval with the LLMs. ***In your homework submission PDF, please list the 3 metrics you researched. What do you think these metrics could help tell you about your design of the RAG system? What decisions would you use these metrics to inform? (hint: what chunking strategy to use? What similarity metric? Where to embed the LLM in the RAG workflow?). Why would this information be helpful to have when designing a RAG?***

Questions (responses required to receive credit for Bonus Tasks):

1. *Quantitative Analysis*: In your homework submission, copy and paste the statistics you produced using metrics you used following the instructions above in **Bonus.A** (above) and **Bonus.B** (above).

2. *Analysis Rationale/Explanation*: Explain what you think the statistics mean (e.g. the ones you produced for **Bonus.A** (above) and **Bonus.B** (above)).

Specifically, answer the following:

2.A. Which LLM's had the **most similar** responses when compared to one another? Why do you think these LLMs had the most similar responses?

2.B. Which LLM's had the **least similar** responses when compared to one another? Why do you think these LLMs had the least similar responses?

3. *Additional Metrics Research*: In your homework submission PDF, please list the 3 metrics you researched for Bonus.E, above.

Specifically, please provide written responses to these questions:

3.A. What 3 metrics did you research? Please list the 3 metrics.

3.B. What do you think the metrics could help tell you about the design of the RAG system?

3.C. What decisions would you use these metrics to inform? (hint: what chunking strategy to use? What similarity metric? Where to embed the LLM in the RAG workflow?)

3.D. Why would this information be helpful to have when designing a RAG?

Please see the next page for Grading Rubric.

Grading

Grade Component	Weight of Total Grade 20%
Part A) Experimenting with Vector Store Query Design	10%
Part B) Experimenting with the Vector Store Embeddings & Query Parameters	10%

Grade Rubrics	Share of Assignment
Accuracy: – Have you followed the instructions correctly? – Have you understood the question correctly?	10%
Experimental Design Evaluation: – Creativity and novelty in analyzing the design of the experiment & executing the tasks.	20%
Completeness: – Have you answered each question (and subquestion)?	30%
Analytics & Insights: – Have you provided a clear rationale and informative explanations for your choices and observations? While we don't require a certain length response, we encourage you to write high-quality responses to the questions asking you to explain your reasoning or analysis of the experiments. A high-quality response is subjective but informed by observations of your experimental results; it may also be informed by facts given in the literature or engineering documents, or other authoritative resources. Grammatical correctness is expected but not as important as sharing your opinion and reasoning about the experiments. There are not necessarily 'right' or 'wrong' answers.	30%
Scientific Rigor: – Documentation, and reproducibility of the results.	10%