

Database Project (Fall 2023)

Homework #10 (50pts, Due date: Dec 6)

Student ID: 2020315798

Student Name: Choi Jin Woo

Instruction: In this homework, we provide a jupyter notebook file (DBP_Homework10.ipynb). You should follow the instructions in these documents. Only the provided code's '*EDIT HERE*' sections must be edited. You must *DELETE* your *GOOGLE_API_KEY* before submitting.

Submission Guide: Submit two files as follows:

- DBP_Homework10_StudentID.zip
- DBP_Homework10_StudentID.ipynb
- DBP_Homework10_StudentID.pdf

1. [20pts] You want to provide a Question Answering service based on Retrieval-Augmented Generation (RAG) for a given webpage. To raise the satisfaction of users, you have to do prompt engineering. Prompt engineering is to make an optimal prompt for a given task.

So, you should implement the RAG pipeline as given parameters.

Parameters
Splitter: RecursiveCharacterTextSplitter Chunk size for splitter: 800 Chunk overlap size for splitter: 100 Vector store: Chromadb Embedding for vector store: GooglePalmEmbeddings Retriever: Same as vector store Retriever search type: similarity Retriever search kwargs k: 6

Next, you should do prompt engineering for the given query string "*What is Task Decomposition?*". **You should apply three prompts to the RAG pipeline: two prompts from Langchain hub (<https://smith.langchain.com/hub>) and one prompt on your own.**

- Langchian hub - rlm/rag-prompt

[Answer]

Enter your code and result here. You must show your result (captured image or string).

```

127 # Langchain hub - gregkamradt/test-question-making
# =====
prompt = hub.out("gk/test-question-making")
# =====
print(f"Prompt template: {prompt}")
print(f"Input variables: {prompt.input_variables}")
print(f"Template: {prompt.template}")
print()

# ===== EDIT HERE =====
rag_chain = (
    {"context": retriever | format_docs("question") | RunnablePassthrough()
    } | prompt
    ) | StrOutputParser()
# =====

print(f"Generated string: {rag_chain}")
for chunk in rag_chain.stream(query):
    print(chunk, end=" ", flush=True)

## Prompt template ##
Input variables: ["context", "question"]
Template: You are an assistant for question-answering tasks. Use the following pieces of retrieved context to answer the question. If you don't know the answer, just say that you don't know. Use three sentences max/row and keep the answer concise.
Question: {question}
Context: {context}
Answer:

## Generated string ##
Task decomposition is the process of breaking down a complex task into smaller, more manageable tasks. This can be done by identifying the subtasks that need to be completed, and then ordering them in a logical way. Task decomposition can be helpful for planning and executing complex tasks, as it can help to ensure that all of the necessary steps are taken. There are a few different ways to decompose a task. One way is to start by identifying the main goal of the task. Once the goal is clear, you can then identify the subtasks that need to be completed in order to achieve the goal. Another way to decompose a task is to start by listing all of the steps that need to be taken. Once you have a list of steps, you can then start to plan and execute the subtasks. It is important to make sure that all of the subtasks are completed in the correct order, and that they are all completed to the required standard. Task decomposition can be a helpful tool for planning and executing complex tasks. It can help to ensure that all of the necessary steps are taken, and that the task is completed in a logical and efficient way. Here are some examples of task decomposition:

- If you want to bake a cake, you could decompose the task into the following subtasks:
    - Preheat the oven.
    - Grease and flour a cake tin.
    - Cream together the butter and sugar.
    - Beat in the eggs.
    - Stir in the flour and baking powder.
    - Pour the batter into the cake tin.
    - Bake the cake in the oven for 30 minutes.
    - Allow the cake to cool before icing it.

- If you want to write a paper, you could decompose the task into the following subtasks:
    - Choose a topic.
    - Do research on the topic.
    - Create an outline for the paper.
    - Write the introduction.
    - Write the body of the paper.
    - Write the conclusion.
    - Revise and edit the paper.
    - Proofread the paper.

- If you want to build a house, you could decompose the task into the following subtasks:
    - Get a building permit.
    - Clear the land.
    - Dig the foundation.
    - Pour the concrete.
    - Build the frame of the house.
    - Install the roof.

```

b. Langchain hub – gregkamradt/test-question-making

[Answer]

Enter your code and result here. You must show your result (captured image or string).

```

128 # Langchain hub - gregkamradt/test-question-making
# =====
prompt = hub.out("gk/test-question-making")
# =====
print(f"Prompt template: {prompt}")
print(f"Input variables: {prompt.input_variables}")
print(f"Template: {prompt.template}")
print()

# ===== EDIT HERE =====
rag_chain = (
    {"context": retriever | format_docs("question") | RunnablePassthrough()
    } | prompt
    ) | StrOutputParser()
# =====

print(f"Generated string: {rag_chain}")
for chunk in rag_chain.stream(query):
    print(chunk, end=" ", flush=True)

## Prompt template ##
Input variables: ["context", "question"]
Template: Your goal is to create a well-crafted set of questions for a test for a specific question. Your questions will be asked on a test to assess a student's knowledge. You will be given the question and your goal is to follow the output format below of guidelines.

Answer-Choice Guidelines:
Position the correct option so that it appears approximately the same number of times in each possible position for a quiz.
Answer-Choices should be written clearly and unambiguously to each other in content, length, and grammar, avoiding giving clues through the use of faulty grammatical construction.
Make all distractors plausible: they should be common misconceptions that learners may have.
In answer choices, avoid "all of the above" and "none of the above," which can lead to artificially higher levels of performance.
In answer choices, avoid references to answer choices by letter (i.e., "both A and B"), as our answers are revealed.
When numeric options are used, the options should be listed in numeric order and in a single format (i.e., as terms or ranges).
Rationale Guidelines:
All rationales should begin with "Correct," or "Incorrect."
All answer options (including correct answer(s) and distractor(s)) must have their own rationale.
Rationales should be unique for each answer option when appropriate. Rationales for distractors should clearly point out a learner's error in understanding and provide context to help them go back and figure out where they went wrong.
Rationales should not refer to the answer by letter (i.e., "option A is incorrect because...") because answer options will be randomized in our quiz.
Rationales for distractors should not give away the correct answer to the question.
For multiple-choice questions (which occur after each module), should include a sentence at the end of each rationale that points the learner back to the relevant video to review the information. E.g., "See 'Why Data Governance'."
Sumative Quiz questions (which occur at the end of the course), should include a sentence at the end of each rationale that points the learner back to the relevant module to review the information. E.g., "See Module 1: What is Data Governance?". They should also include the module's learning objective at the end of each rationale.

Example of a Quiz Question Substitution
Below are examples for each component of a multiple-choice question item.

Quiz Example:
A company is storing an access key (access key ID and secret access key) in a text file on a custom VM. The company uses the access key to access DynamoDB tables from instances created from the AMI. The security team has audited a more secure solution. Which solution will meet the security team's mandate?

Answer Choices (Distractors A-C and Correct Answer D) Examples:
A. Put the access key in an S3 bucket, and retrieve the access key on boot from the instance.
B. Pass the access key to the instances through instance user data.
C. Store the access key in a key server launched in a private subnet.
D. Create an IAM role with permissions to access the table, and launch all instances with the new role. (correct)

Rationale Example:
[Feedback] Incorrect: Data governance is not something specific to big data technologies. See "Why Data Governance."
[Feedback] Incorrect: It is not relevant to clarify the size of the big data test. Learning Objective: Apply Redox and use new tools to manage and control without compromising the platform's basic value. Review Module 1: "What is Data Governance?"

Context:
Question: {question}
Valid answer:

## Generated string ##
Task decomposition is the process of breaking down a complex task into smaller, more manageable tasks. This can be done by identifying the subtasks that need to be completed in order to achieve the overall goal, and then organizing them into a logical sequence. Task decomposition can be helpful for a variety of reasons, including:

```

c. Create a new prompt to help your QA service.

[Answer]

Enter your code and result here. You must show your result (captured image or string).

```

# Create new prompt to help your QA service
# ===== EDIT HERE =====
from langchain.prompts import HumanMessagePromptTemplate, SystemMessagePromptTemplate, MessagePromptTemplate
prompt = ChatPromptTemplate.from_template("""
You are an assistant for question-answering tasks. Use the following pieces of retrieved context to answer the question.
If you don't know the answer, just say that you don't know. Use three sentences to exclude and keep the answer concise.
Question: {question}
Context: {context}
Answer:
""")

# ===== EDIT HERE =====
# Create LLM
llm = ChatOpenAI(model_name="gpt-4")

# Create Retriever
retriever = Retriever.from_embeddings(vectorstore, retriever)

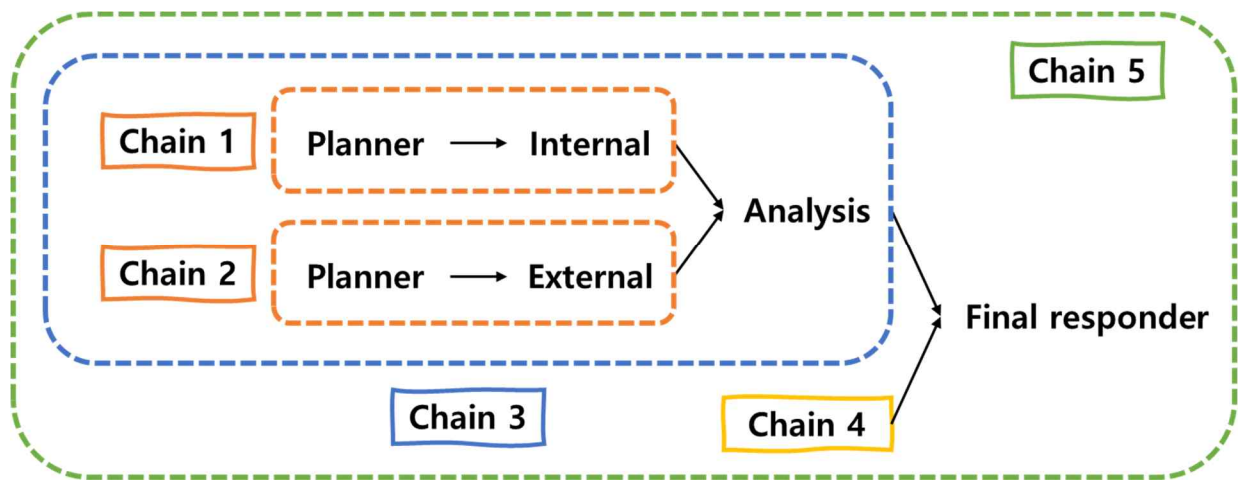
# Create Chain
chain = chain(prompt, llm, retriever)

# Run Chain
result = chain.run(question)

print(result)

```

2. [30pts] Implement a RAG multiple chain pipeline. The purpose of this pipeline is to find information about stakeholders for a given project. **The pipeline contains five chains.** The details of the chains are shown in the below figure.



The implementation parameters are same as question 1 in text_splitter, vectorstore, retriever, llm variables. The prompts for planner, internal and external are in the given ipynb file. You can use from_temptate() method for them. But you have to use from_messages() method for analysis and final_responder. The specific messages are shown in the table below.

analysis	("system", "Generate a stakeholder analysis map for a given project."), ("human", "The details of a given project is following: {result3}"), ("ai", "Internal stakeholders:\n{result1}\n\nExternal stakeholders:\n{result2}")
final_responder	("system", "Generate a final response given the information."), ("ai", "{a_response}"), ("human", "Common pitfalls:\n{a_pitfalls}")

Read the given ipynb file carefully and write all your codes and results here.

[Answer]

Enter your code and result here. You must show your result (captured image or string).

```

[125] planner_prompt_template = "I want to do the project management for {input}"
planner_prompt = ChatPromptTemplate.from_template(planner_prompt_template)

# ===== EDIT HERE =====
# Planner
# Use 'from_template' method
planner = (
    planner_prompt
    | llm
    | StrOutputParser()
)
# =====

```

```

[126] internal_prompt_template = ""Answer the question based only on the following context:
{context}

Question: {question}

Answer in the following: {base}
""
internal_prompt = ChatPromptTemplate.from_template(internal_prompt_template)

# ===== EDIT HERE =====
# Internal
# Use 'from_template' method
# Use 'question', 'retriever', 'format_docs' for 'context'
internal = (
    ("context": itemgetter("question") | retriever | format_docs, "question": itemgetter("question"), "base": planner)
    | internal_prompt
    | llm
    | StrOutputParser()
)
# =====

```

```

[127] external_prompt_template = ""Answer the question based only on the following context:
{context}

Question: {question}

Answer in the following: {base}
""
external_prompt = ChatPromptTemplate.from_template(internal_prompt_template)

# ===== EDIT HERE =====
# External, it is similar to 'internal'
# Use 'from_template' method
# Use 'question', 'retriever', 'format_docs' for 'context'
external = (
    ("context": itemgetter("question2") | retriever | format_docs, "question": itemgetter("question2"), "base": planner)
    | internal_prompt
    | llm
    | StrOutputParser()
)
# =====

```

```

# ===== EDIT HERE =====
# Analysis
# Use 'from_messages' method
analysis = (
    ChatPromptTemplate.from_messages(
        [
            ("system", "Generate a stakeholder analysis map for a given project."),
            ("human", "The details of a given project is following: {result3}"),
            ("ai", "Internal stakeholders: {result1} External stakeholders: {result2}")
        ]
    )
    | llm
    | StrOutputParser()
)
# =====

```

```

[128] # ===== EDIT HERE =====
# Final responder
# Use 'from_messages' method
final_responder = (
    ChatPromptTemplate.from_messages(
        [
            ("system", "Generate a final response given the information."),
            ("ai", "{a_response}"),
            ("human", "Common pitfalls: {a_pitfalls}")
        ]
    )
    | llm
    | StrOutputParser()
)
# =====

```

```

[129] # You can check that you implement 'planner' correctly using this cell
planner.invoke({"input": "scrum"})

```

Scrum is a framework for developing and managing complex products. It is a lightweight framework with a focus on iterative development, team collaboration, and accountability. Scrum teams are cross-functional and self-organizing, and they work in sprints of two weeks or less. The Scrum Master is responsible for ensuring that the Scrum process is followed and that the team is able to deliver the product backlog. The Product Owner is responsible for prioritizing the product backlog and ensuring that the team is working on the most important features. The Development Team is responsible for developing the product according to the Scrum process. To do project management for Scrum, you need to understand the Scrum framework and the roles and responsibilities of each team member. You also need to be able to manage the Scrum process, including sprint planning, sprint execution, and sprint retrospectives. Here are some tips for project management for Scrum:

- Understand the Scrum.

```

[131] # ===== EDIT HERE =====
# Chain 1
chain1 = internal
# =====

# You can check that you implement 'chain1', 'internal' correctly using this cell
chain1.invoke({"input": "scrum", "question": "List the internal stakeholders."})

```

Internal stakeholders are people within an organization who are affected by or have an interest in the outcome of a project. They can include:

- Project manager
- Project team members
- Project sponsor
- Executive leaders
- Other cross-functional internal team members

It is important to identify and manage internal stakeholders effectively in order to ensure the success of a project. This can be done by:

- Communicating regularly with stakeholders
- Keeping stakeholders informed of progress and changes
- Involving stakeholders in decision-making
- Managing stakeholder expectations
- Taking the time to manage internal stakeholders effectively, project managers can help to ensure that projects are completed on time, within budget, and to the satisfaction of all stakeholders.

Here are some additional tips for managing internal stakeholders:

- Identify your stakeholders early on.
- The first step in managing internal stakeholders is to identify who they are. This includes...

```

[132] # ===== EDIT HERE =====
# Chain 2
chain2 = external
# =====

# You can check that you implement 'chain2', 'external' correctly using this cell
chain2.invoke({"input": "scrum", "question2": "List the external stakeholders."})

```

External stakeholders are people or groups outside of the organization that are affected by the project or have an interest in its success. They can include customers, suppliers, partners, regulators, and the public. External stakeholders can play a significant role in the success of a project. They can provide feedback on the project, help to identify risks and opportunities, and advocate for the project within their own organizations. It is important to identify and manage external stakeholders early in the project. If you do this, you will help to ensure that their needs are met and that they are supportive of the project. There are a number of ways to manage external stakeholders. One way is to create a stakeholder register. This is a document that lists all of the project's external stakeholders and their contact information. The stakeholder register should also include information on each stakeholder's interest in the project, their level of influence, and any potential risks or...

```

[133] # ===== EDIT HERE =====
# Chain 3
# For 'result3' use the following line
# 'result3' | itemgetter("question3") | retriever | format_docs
chain3 = (
    [
        ("result3": itemgetter("question3") | retriever | format_docs,
         "result2": chain2,
         "result1": chain1)
    ]
    | analysis
)
# =====

# You can check that you implement 'chain3', 'analysis' correctly using this cell
chain3.invoke(
    {"input": "scrum",
     "question1": "List the internal stakeholders.",
     "question2": "List the external stakeholders.",
     "question3": "How to do a stakeholder analysis?"})

```

Stakeholder analysis is a process of identifying and assessing the needs, interests, and potential impact of all the people or groups who may be affected by a project. It is an essential part of project management, as it helps to ensure that all stakeholders are considered and that their needs are met. Internal stakeholders are those who are directly involved in the project, such as the project manager, project team members, and project sponsor. External stakeholders are those who are not directly involved in the project, but who may be affected by its outcome, such as customers, suppliers, partners, and the public. There are a number of benefits to conducting a stakeholder analysis. First, it helps to identify all of the stakeholders who may be affected by the project. This is important, as it ensures that all stakeholders are considered and that their needs are met. Second, it helps to assess the needs and interests of...

```
[134] # Chain 4 (Don't modify)
chain4 = (
    ("question": llmgetter("question4"),
     "context": llmgetter("question4") | retriever | format_docs)
    | ChatPromptTemplate.from_template("{context} {question}")
    | llm
    | StrOutputParser()
)

chain4.invoke("question4": "List the common stakeholder mapping pitfalls.")
```

Here are the common stakeholder mapping pitfalls: **Lack of boundaries**: Overeager project stakeholders can cause scope creep. To avoid this, implement a change control process. **Not including stakeholders**: During the initial stakeholder identification phase, you forgot about a stakeholder who has a high interest in this initiative. To avoid this, involve your project team during the stakeholder identification process. **Bringing stakeholders in too late**: You already kicked off your project, and then started your stakeholder analysis. To avoid this, create a stakeholder analysis map up front. **Not communicating effectively**: You're not communicating effectively with your stakeholders, which is leading to misunderstandings and frustration. To avoid this, create a communication plan and stick to it. **Not managing expectations**: You're not managing your stakeholders' expectations, which is leading to disappointment and frustration. To avoid this, set clear expectations.

```
[135] # ===== EDIT HERE =====
# Chain 5
chain5 = (
    ("a_response": chain0,
     "a_pitfalls": chain4)
    | final_responder
)
# =====

# You can check that you implement "chain5", "final_responder" correctly using this cell
chain5.invoke(
    "input": "scrum",
    "question1": "List the internal stakeholders.",
    "question2": "List the external stakeholders.",
    "question3": "How to do a stakeholder analysis map?",
    "question4": "List the common stakeholder mapping pitfalls."
)
```

Stakeholder analysis is a critical part of project management. By understanding the needs and expectations of stakeholders, project managers can develop strategies to manage them effectively. However, there are a number of common pitfalls that can derail stakeholder management efforts: **Not including stakeholders**: This can happen if stakeholders are not identified early in the project or if they are not given an opportunity to provide input. When stakeholders are not included, they may feel like they are not being heard and may become resistant to the project. **Another common pitfall is not communicating effectively**. Stakeholders need to know what is going on with the project and they need to know it in a timely manner. If communication is not effective, stakeholders may become frustrated and may start to spread rumors or negative information about the project. **Not managing expectations** is another common pitfall. Stakeholders need to know what to