NLP_HW4

NTU b10705013 陳彥廷

Questions

Q1:

Please describe the details of your implementation for the RAG system (please tell us 1. What’s in your RAG system? 2. Which retrieval model you use? 3. What’s your prompt? 4. What’s new in your code in comparison with the code from our lab course?) in this assignment and list your best score for the ten questions.

A1:

1. 我使用了langchain框架，並使用ollama在本地運行llama3.2:1b以及Chroma來做為向量資料庫。依照題目的需求使用RAG並基於cat-facts.txt這個文件來回答關於貓的各種問題。

2. 經過測試，我使用了mmr並將k值設定成3，fetch_k為20。

3. 我使用了TA類型的prompt，內容如下:

```
system_prompt = (
 "You are an expert teaching assistant helping explain concepts using the given context. "
 "When answering, be clear and precise. "
 "Always provide the reasoning for your answer before stating the final conclusion. "
 "Context: {context}"
)
prompt = ChatPromptTemplate.from_messages(
    [
        ("system", system_prompt),
        ("human", "Based on the context, please answer the following
question:\n{input}"),
    ]
)
```

4. 與原本的程式碼相比，我調整了非常多的參數以及測試了各種模型以及 prompt 的方法。

5. 最後的最高結果為 10

Q2:

Please provide analysis for the RAG performance using different prompts

A2:

我總共嘗試了三種方法，包含了 CoT、TA 以及一般的 prompt :

1. 第一種方法我嘗試的是 CoT，這種方式的優點在於可以引導模型做步驟式的思考，增強模型的推理能力，缺點在於答案可能較長且不夠直接。

   ```
   cot_system_prompt = ( "Use the provided context to answer the question. " "Follow this approach: First, think step by step using the context to analyze the question. " "Then, provide your final answer in a concise manner. " "If you don't know the answer, say 'I don't know'. " "Context: {context}" ) cot_prompt = ChatPromptTemplate.from_messages( [ ("system", cot_system_prompt), ("human", "Question: {input}\n\nStep-by-step reasoning:"), ] )
   ```

2. 第二種我嘗試的方式是 TA 類型的 prompt，這種方法的優點在於可以幫助模型明確角色定位。

   ```
   ta_system_prompt = ( "You are an expert teaching assistant helping explain concepts using the given context. " "When answering, be clear and precise. " "Always provide the reasoning for your answer before stating the final
   ```

conclusion. " "Context: {context}") ta_prompt = ChatPromptTemplate.from_messages([("system", ta_system_prompt), ("human", "Based on the context, please answer the following question:\n{input}"),])

3. 最後一種則是沒有經過任何優化的 prompt，優點在於回答會比較直接。

   trivia_system_prompt = ("Answer the question directly using the provided context. " "Keep the response short and factual. "Context: {context}") trivia_prompt = ChatPromptTemplate.from_messages([("system", trivia_system_prompt), ("human", "Question: {input}"),])

從結果來說，最好的 prompt 是 TA 類型的 prompt，平均正確大概在 9-10 左右。

Q3:

Please compare the RAG performance with different retrieval models and the performance without using RAG (note that Llama 3.2 should not be fine-tuned in this assignment).

A3:

以下測試基於使用 TA prompt，mmr (k=5, fetch_k=20) 來做測試

1. jinaai/jina-embeddings-v2-base-en : 9
2. FacebookAI/roberta-base : 4
3. sentence-transformers/all-mpnet-base-v2 : 9
4. without RAG : 0

Q4:

Anything that can strengthen your report.

A4:

這次所使用的程式碼多為助教課上所提供的程式碼，只需修改參數就能讓準確率達到 100 %，因此下面提供我跑過的參數

模型	search_type	k	fetch_k	prompt	result
all-mpnet-base-v2	mmr	3	20	TA	10
all-mpnet-base-v2	mmr	5	10	TA	9
all-mpnet-base-v2	mmr	3	20	CoT	7
all-mpnet-base-v2	similarity	5	x	TA	7
roberta-base	mmr	5	20	TA	4
roberta-base	similarity	5	x	TA	6
roberta-base	mmr	3	20	CoT	7
jina -v2-base-en	similarity	5	x	CoT	8
jina -v2-base-en	similarity	3	x	TA	8
jina -v2-base-en	mmr	5	20	TA	9
jina -v2-base-en	mmr	3	10	TA	8

由上表可知，roberta 這個 embedding model 並不太適合這個 case，且降低 k 的值有助於幫助模型找出重點以及正確答案。

Settings

All code run in colab

References

Copilot in all code

Chatgpt in report