

Assignment 2

Building a LLM-based Agent for Reviewing Lecture Material based on Retrieval Augmented Generation

Name - Hugo Collins
Student ID - 203311486

Abstract

In this report I detail the creation of my LLM-based Agent for Reviewing Lecture Material based on Retrieval Augmented Generation. During this assignment I used a curated set of lectures on human sensing by Prof. Alan Smeaton which served as the foundational corpus for my RAG system. My RAG system was created through google colab and used llamaIndex, Milvus, OpenAI and TruLens. In this report I detail the different approaches and technologies I researched, the steps taken to build a number of different prototypes and the evaluation of all my different results. Initially I ran a baseline RAG system and then through a cycle of iteration and evaluation, I explored a number of different RAG applications and compared the results of each to assess performance. This report comprehensively documents each step of the process and details some of the interesting findings and insights I discovered during the process.

Introduction

Prior to this assignment I had little to no experience in the field of Retrieval Augmented Generation. Therefore the first step was to do some research into the topic and see what possible approaches I could take. First, I read over the lectures from Stanford University which were recommended to us in the assignment outline. I found these lectures to be extremely useful and something I would definitely recommend providing to students who are attempting this project in the future. They provided a base understanding of the technologies I'd be using and were definitely of benefit to my project.

On top of exploring and gaining an understanding of how RAG models work, I started looking at the best approaches to building my RAG model. While the approach outlined in the Assignment briefing seemed to be a popular approach taken by people when building their LLM-Based Agent, I wanted to explore a few different approaches to see if any appealed to me. I explored the possibilities of using Huggingface and Langchain as possible alternatives to the technologies mentioned in the assignment outline. However, upon deeper examination I decided against using either. The reasons for this are detailed below in the report.

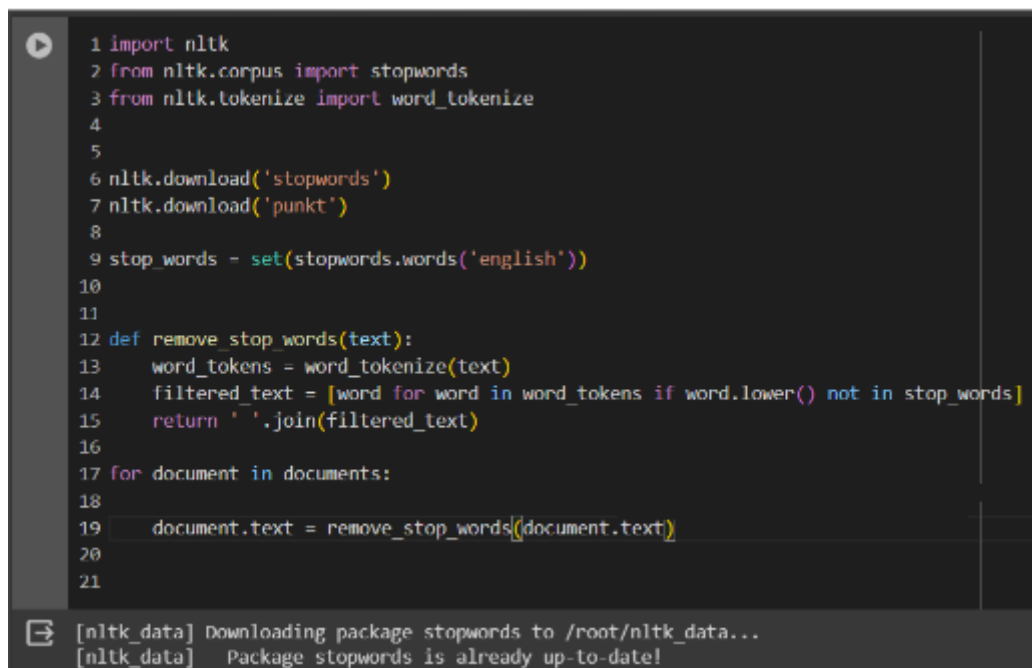
Choosing the right platform for running my code was the first critical decision in my project. I explored several options, including using Replit, running the code locally on a Jupyter Notebook, or using Google Colab. My investigation revealed that all the tools and technologies I intended to use for this project—LlamaIndex, Milvus, OpenAI, and TruLens—were fully supported by Google Colab. Also, considering the substantial computational resources required for an efficient Retrieval-Augmented Generation (RAG) operation, and recognising that my laptop fell short in this aspect, running the code locally wasn't a viable option. These factors steered me towards choosing Google Colab as my primary coding environment. Colab's seamless integration with the tools I planned to use and its provision of the necessary computational power, making it the most suitable platform for my assignment.

The next step I needed to take was getting my OpenAI API key in order to use the OpenAI embeddings needed for my agent. This key allows me to access advanced functionalities offered by GPT-4 and GPT-3.5 Turbo models, known for their exceptional performance in natural language understanding and generation. Recognising the potential limitations caused by my initial subscription plan, I decided to upgrade my account. Before upgrading, I had attempted to find a solution without having to upgrade my plan. I tried using the open source lambda models provided by OpenAI but I was running into problems when I would try to evaluate this

with TruLens. There appeared to be some compatibility issues between the models and TruLens and I encountered a lot of problems here. This was becoming a very time consuming process so I decided an upgrade in my plan was necessary. By paying a small bit extra, I was able to get access to the top of the range features of these advanced models which I felt was important for this project and also allowed me to use TruLens correctly. This decision increased my API usage limits and allowed me to leverage the cutting-edge capabilities of GPT-4 and GPT-3.5 Turbo without constraints which is what I wanted.

Data Preprocessing:

One avenue I explored to help improve my RAG prototype was to preprocess the data being taken in from the curated set of lectures on human sensing by Prof. Alan Smeaton. I was reading the data into colab using a function which used the PyPDF2 library to convert the PDF to text and save it under the variable documents. Each document had an ID, a page label and the text from the PDF slide. The document object had a length of 216, which reflected the 216 pages in the curated set of notes. Once I knew the data had been read in correctly I decided to experiment if some data preprocessing could affect results in a positive way. I ran one notebook which removed stop words from the text using the NLTK library.



```
1 import nltk
2 from nltk.corpus import stopwords
3 from nltk.tokenize import word_tokenize
4
5
6 nltk.download('stopwords')
7 nltk.download('punkt')
8
9 stop_words = set(stopwords.words('english'))
10
11
12 def remove_stop_words(text):
13     word_tokens = word_tokenize(text)
14     filtered_text = [word for word in word_tokens if word.lower() not in stop_words]
15     return ' '.join(filtered_text)
16
17 for document in documents:
18
19     document.text = remove_stop_words(document.text)
20
21
```

[nltk_data] Downloading package stopwords to /root/nltk_data...
[nltk_data] Package stopwords is already up-to-date!

My thought process behind this was that eliminating stop words would reduce the noise in my dataset. This would then in turn enable my model to focus on more content-rich words likely to enhance the document's relevance. I expected this to improve answer accuracy and also groundedness as answers are more likely to be directly based on the most relevant information from the source documents, as the noise from common but less informative words is reduced. The results of this process can be found in the result section of the report below.

Docker Issues

A major problem I encountered during this process was my attempt to run Milvus standalone through docker. This was a process which I decided I could follow after finding material about it online. I downloaded and installed docker and Milvus standalone onto my laptop. In order to run this, I had originally been running my code using

```
vector_store = MilvusVectorStore(index_params={
    "index_type": "IVF_FLAT",
    "metric_type": "L2"
},
    search_params={"nprobe": 20},
    overwrite=True)
llm = OpenAI(model="gpt-3.5-turbo")
```

This however was looking for a localhost and I could not get a connection. I attempted to link my local jupyter notebook to my colab and run the colab locally to access this docker-localhost but unfortunately I didn't have the necessary computational resources to achieve this. I spent a significant amount of time researching this and trying different ways to implement this using WSL and docker. This ultimately was unsuccessful and was quite time consuming but it was an effective learning experience for me. It also helped me find what turned out to be my final solution which was to run everything through colab and to avoid docker all together.

Building my RAG System Prototypes

In this section of the report, I outline my methodology used in constructing a Retrieval Augmented Generation (RAG) model, specifically focusing on the integration of advanced language models, vector storage solutions, and feedback mechanisms. The core of this methodology revolves around using OpenAI's GPT-3.5 Turbo model, Milvus as a vector store, and the TruLlama framework for model evaluation and feedback.

To begin, I used the OpenAI class from the llama_index.llms.openai module to set the GPT-3.5 Turbo model with a specific temperature setting of 0.1. I tried a few different values for this setting but chose 0.1 as it gave the best balance of creativity while keeping coherence in the model's responses. I then used the OpenAIEmbedding class to transform textual data into embeddings that capture the content of surrounding words which would be essential for the retrieval phase of the RAG model and also important to improve the context relevance metric in my evaluation.

When deciding on the best way to store and manage the vector data for my RAG system, I looked into several options, including Milvus, FAISS, and Annoy. During my research I read that FAISS, developed by Facebook, is great for quickly searching through dense vectors and has a lot of options for different kinds of searches. However, fitting it with OpenAI's embeddings and making sure it worked smoothly with the rest of the RAG setup was a bit tricky. Annoy on the other hand, I didn't attempt to use, as I read that it struggles slightly with vector embeddings from complex models like GPT 4. Milvus' ability to work well with OpenAI's embeddings plus its compatibility with colab were the key reasons behind choosing Milvus as my memory store.

In Colab, I accessed MilvusVectorStore from the llama_index.vector_stores.milvus module. The Milvus server was initiated to ensure the vector store was ready to receive and index the embeddings. See code below:

```
1 from llama_index.vector_stores.milvus import MilvusVectorStore
2 from milvus import default_server
3
4 default_server.start()
5
6 vector_store = MilvusVectorStore(dim=1891, overwrite=False)
```

There were a few issues with importing the necessary packages here but I solved it with the following:

```
#!pip install grpcio==1.60.0 --force-reinstall
```

The dataset for the RAG model consisted of a collection of question-and-answer (QA) pairs, loaded from a CSV file. I created these questions myself by reading through the notes and selecting a wide variety of effective questions. I tried to choose questions that covered a wide variety of topics and would give my model the best chance of performing well across all four evaluation metrics. I tried two different sets of questions but the final set I chose helped my model to perform better so I chose that set. These questions can be found [here](#). I settled on 10 different prompts/questions in total. These question answer pairs served as a basis for evaluating the model's performance using TruLens.

To construct the RAG model and integrate the components, I employed the TruLlama framework from the `trulens_eval` package. This framework allows the creation of RAG systems by providing modules for feedback and evaluation. I used various feedback modules such as `GroundTruthAgreement` and `Groundedness` to assess the model's performance across different metrics, including answer correctness and the relevance of the generated content to the query context. These feedback mechanisms were crucial in iterating and refining the model to improve its accuracy and relevance. The `VectorStoreIndex` class, coupled with the `MilvusVectorStore` (which was stored under variable `vector_store`), was used to index my dataset, allowing the RAG model to efficiently retrieve relevant information during the generation process. The indexed data was then used by my query engine to support the RAG process, enabling the model to draw from lots of information to improve its outputs.

```
Successfully installed importlib_metadata-7.0.1 zipp-3.17.0

1 import trulens_eval.feedback as feedback
2 import numpy as np
3 from trulens_eval import TruLlama, Feedback, Tru, feedback
4 from trulens_eval.feedback import GroundTruthAgreement, Groundedness
5
6 tru = Tru()
7 gpt3_5feedback = feedback.OpenAI()
8 hugs = feedback.Huggingface()
9 storage_context = StorageContext.from_defaults(vector_store=vector_store)
10 index = VectorStoreIndex.from_documents(documents, storage_context=storage_context)
11 basic_query_engine = index.as_query_engine()
12

```bash
pip uninstall -y trulens_eval
pip install trulens_eval
```

WARNING:trulens_eval.utils.imports:Package ipython is installed but has a version conflict:
```

Evaluation

In order to evaluate the RAG prototype I decided to use TruLens. TruLens, as stated on their webpage, is a software tool that helps you to objectively measure the quality and effectiveness of your LLM-based applications using feedback functions. For this assignment I decided to leverage four main feedback functions using TruLlama - Groundedness, Context Relevance, Answer Relevance and Answer Correctness. In order to implement these functions I used the following pieces of code -

```
grounded = Groundedness(groundedness_provider=gpt3_5feedback
feedback_groundness = Feedback(grounded.groundedness_measure_with_cot_reasons, name =
"Groundedness").on(
    TruLlama.select_source_nodes().node.text.collect() # context
).on_output().aggregate(grounded.grounded_statements_aggregator)
```

```
feedback_contextrelevance = Feedback(gpt3_5feedback.qs_relevance, name = "Context
Relevance").on_input().on(
    TruLlama.select_source_nodes().node.text
).aggregate(np.mean)
```

```
feedback_answerrelevance = Feedback(gpt3_5feedback.relevance_with_cot_reasons, name = "Answer
Relevance").on_input_output()
```

```
feedback_answercorrectness = Feedback(
    GroundTruthAgreement(data).agreement_measure, name="Answer Correctness"
).on_input_output()
```

During the evaluation process I followed a quite time consuming process of lots and lots of iterations, creating lots of different prototypes and trying different methods in order to find my optimal RAG prototype. Some of the steps I followed are documented below:

Step 1 - Run a Basic RAG

The initial step I took was to run a basic RAG prototype with no enhancements and to assess the output. This was done using

```
1 tru_query_engine_recorder = TruLlama(basic_query_engine,
2   app_id='RAG 1 - Basic',
3   feedbacks=[feedback_answercorrectness, feedback_groundedness, feedback_answerrelevance, feedback_contextrelevance ])

1 tru.run_dashboard()

Starting dashboard ...
npm: installed 22 in 8.892s

Go to this url and submit the ip given here. your url is: https://floppy-ends-wave.local.it

Submit this IP Address: 34.138.72.213

<Popen: returncode: None args: ['streamlit', 'run', '--server.headless=True'...>
```

A simple for loop then iterated through the questions and TruLens would return a score for each of the feedback functions which I specified above. When viewing my output, I decided to avail of the more user-friendly interface of using the TruLens dashboard rather than viewing the results through a colab cell. This link would lead to a separate dashboard which prompted me to input the IP Address given and then I would be able to view my results there. This can be seen in the later section of this report where I detail my findings.

Step 2 - SetenceWindowNodeParser

The next step I took to try to improve my RAG prototypes was to implement a SentenceWindowNodeParser approach. The idea of this is to segment documents into smaller parts (windows) of a specified size. This helps in focusing on specific sections of text for more accurate retrieval and generation, enhancing the model's ability to generate contextually relevant and grounded responses. I did this in an attempt to improve groundedness and context relevance as after the initial iteration I noticed that these were two areas that were lacking. In order to test this parser I created a number of different prototypes all of varying window sizes. The idea behind this was to analyse which window sizes gave the best results. I decided to set the window sizes to 2, 3, 5 and 6. I also wanted to add a prompt alongside a number of these prototypes to really get a sense for which prototype to use and to have a wide variety to choose from. The results can be seen below step 4.

Step 3 - Run with simple query

In order to try to improve my RAG prototype I decided to implement a simple system prompt. To start off I wanted to analyse the effect a simple basic prompt could have so I decided to use -

“You are an expert in human sensing. Answer the questions as well and as accurately as possible based on instructions and context provided”

I passed this prompt to the prototype by adding the extra parameter to the feedbacks variable

```
feedbacks=[f_groundtruth, f_groundedness, f_qa_relevance, f_qs_relevance], system_prompt=simple_prompt)
```

The results of the separate prototypes can be seen below when I compare all the results. The different prototypes I created throughout my iterations are as follows:

- 1: Basic RAG
- 2: Basic RAG with prompt
- 3 Basic RAG with window of 2 plus prompt
- 4 Basic RAG window of 3
- 5 Basic RAG window of 3 plus prompt
- 6 Basic RAG window of 5
- 7 Basic RAG window of 5 plus prompt
- 8 Basic RAG window of 6 plus prompt

Results of running with basic prompt: (May need to zoom in to see figure. Apologies. It was the only way I could fit in results of all 8 prototypes)

| | | | | | | | | |
|---|----|-----|--------|-------|--------------------|--------------------|--------------------|--------------------|
| RAG 1 - Basic | 10 | 4.1 | \$0.01 | 4.4k | 0.72
▲ not good | 0.76
▲ not good | 0.96
■ not good | 0.83
■ not good |
| RAG 2 - Basic with Prompt | 10 | 4.1 | \$0.01 | 4.37k | 0.7
▲ not good | 0.75
▲ not good | 0.94
■ not good | 0.83
■ not good |
| RAG 5 - sentence window of 5 | 10 | 4.1 | \$0 | 3.11k | 0.79
▲ not good | 0.63
▲ not good | 0.94
■ not good | 0.83
■ not good |
| RAG 6 - sentence window of 5 plus simple prompt | 10 | 4.1 | \$0 | 3.08k | 0.78
▲ not good | 0.6
● bad | 0.94
■ not good | 0.81
■ not good |
| RAG 7 - sentence window of 6 plus simple prompt | 10 | 4.1 | \$0 | 3.19k | 0.81
■ not good | 0.7
▲ not good | 0.94
■ not good | 0.86
■ not good |
| RAG 8 - sentence window of 2 plus simple prompt | 10 | 4.1 | \$0 | 3.1k | 0.75
▲ not good | 0.6
● bad | 0.92
■ not good | 0.91
■ not good |
| Rag 3 - sentence window of 3 | 10 | 4.1 | \$0 | 3.15k | 0.73
▲ not good | 0.53
● bad | 0.94
■ not good | 0.83
■ not good |
| Rag 4 - sentence window of 3 plus simple prompt | 10 | 4.1 | \$0 | 3.2k | 0.8
▲ not good | 0.65
▲ not good | 0.91
■ not good | 0.88
■ not good |

We can see from these results that my prototypes perform very well on answer relevance and answer correctness. Context relevance is actually not too bad especially for those prototypes using both the SentenceWindowNodeParser and the prompt together. The prototypes really struggle with groundedness however, which needs to be tackled.

Step 4 - Improving System Prompt

The next step I decided to take was to tackle improving the prompt. From the results above its clear that groundedness and context relevance were areas where the model struggled. I needed to improve the groundedness results significantly to improve my prototypes all-round. Through an iterative and time consuming process of running and rerunning the prototypes I eventually created the prompt -

“Imagine you're the go-to person for all things related to Human Sensing technology. Your task is to answer questions with accuracy, ensuring your responses are not only correct but also rich with contextually relevant insights. Delve into the specifics of human sensing technologies, discussing their applications, impacts, and the principles behind them in a manner that's accessible to everyone. Strive to break down complex concepts by closely linking them to the context of the question, making sure every explanation is well-founded and directly applicable. Your objective is to provide answers that are not just informative but also leverage the detailed context available to you.

The updated results were as followed:

| App Leaderboard | | | | | | | | |
|---|-----------|----------|-------------------|--------------|-----------|--------------|---------|------------|
| Prototype Performance Comparison (Average Scores) | | | | | | | | |
| Prototype | Questions | Accuracy | Context Relevance | Groundedness | Coherence | Completeness | Clarity | Engagement |
| RAG 1 - Basic | 10 | 4.6 | \$0.01 | 4.4k | 0.74 | 0.57 | 0.96 | 0.84 |
| RAG 2 - Basic with Prompt | 10 | 4.6 | \$0.01 | 4.42k | 0.74 | 0.7 | 0.96 | 0.85 |
| RAG 5 - sentence window of 5 | 10 | 4.6 | \$0 | 3.15k | 0.78 | 0.64 | 0.94 | 0.82 |
| RAG 6 - sentence window of 5 plus simple prompt | 10 | 4.6 | \$0.01 | 3.23k | 0.71 | 0.65 | 0.94 | 0.83 |
| RAG 7 - sentence window of 6 plus simple prompt | 10 | 4.6 | \$0 | 3.08k | 0.76 | 0.57 | 0.96 | 0.88 |
| RAG 8 - sentence window of 2 plus simple prompt | 10 | 4.6 | \$0 | 3.15k | 0.82 | 0.53 | 0.94 | 0.89 |
| Rag 3 - sentence window of 3 | 10 | 4.6 | \$0 | 3.11k | 0.76 | 0.58 | 0.96 | 0.84 |
| Rag 4 - sentence window of 3 plus simple prompt | 10 | 4.6 | \$0 | 3.15k | 0.8 | 0.66 | 0.92 | 0.85 |

There were a few subtle changes across the results but overall, the changing of the system prompt had little impact on the overall scores

Step 5: Run all on preprocessed data

The next step was to run the prototypes on the preprocessed data which removed stop words - (using the improved prompt). The preprocessing stage and the code used was documented earlier in this report under the preprocessing header. The evaluation results of the prototypes run on this preprocessed data can be found below:

Average feedback values displayed in the range from 0 (poor) to 1 (best)

| Row ID | App Type | User Input | Response | Application Tag | Flow Name | Answer Correctness | Content Relevance | Answer Relevance | Granularity | Latency (ms) | Total Cost (USD) | Row Label | Row Details |
|--------|-----------------------|----------------------------|-----------------------------------|-----------------------------------|------------------------|--------------------|-------------------|------------------|--------------------|--------------|------------------|-----------|-------------|
| Row 1 | sentiment-analysis-v1 | Review:terribleproduct.com | "You are analyzing appraisals." | "Analyzing appraisals v1..." | 2024-01-20T10:05:18350 | 1 | 0.9 | 0.8 | 0.8000000000000000 | 3 | 280 | 0.003715 | None |
| Row 2 | sentiment-analysis-v1 | Review:terribleproduct.com | "What challenges are you facing?" | "Challenges associated with..." | 2024-01-20T10:05:20888 | 0.9 | 0.2 | 1 | 0 | 4 | 280 | 0.003096 | None |
| Row 3 | sentiment-analysis-v1 | Review:terribleproduct.com | "You have suggested null." | "Suggested null (00)..." | 2024-01-20T10:05:20974 | 0.9 | 0.2 | 0.8 | 0.8 | 4 | 280 | 0.004005 | None |
| Row 4 | sentiment-analysis-v1 | Review:terribleproduct.com | "What is the significance of...?" | "Significance associated with..." | 2024-01-20T10:05:21888 | 0.9 | 0.2 | 0.8 | 0.8 | 4 | 280 | 0.003096 | None |
| Row 5 | sentiment-analysis-v1 | Review:terribleproduct.com | "You do variable device v..." | "Variable device v1..." | 2024-01-20T10:05:21888 | 0.9 | 0.8 | 0.8 | 0.8000000000000000 | 5 | 280 | 0.003096 | None |
| Row 6 | sentiment-analysis-v1 | Review:terribleproduct.com | "What did you do machine v..." | "Machine learning play v..." | 2024-01-20T10:05:21888 | 0.9 | 0.8 | 0.8 | 0.8 | 4 | 281 | 0.003075 | None |
| Row 7 | sentiment-analysis-v1 | Review:terribleproduct.com | "You do accelerometers b..." | "Accelerometers help in m..." | 2024-01-20T10:05:21888 | 0.9 | 0.8 | 1 | 1 | 4 | 280 | 0.004005 | None |
| Row 8 | sentiment-analysis-v1 | Review:terribleproduct.com | "What are you doing app..." | "What are you doing app..." | 2024-01-20T10:05:21888 | 0.9 | 0.8 | 1 | 1 | 4 | 280 | 0.003096 | None |
| Row 9 | sentiment-analysis-v1 | Review:terribleproduct.com | "You do location setting..." | "Location setting control..." | 2024-01-20T10:05:21888 | 1 | 0.4 | 1 | 0.75 | 7 | 275 | 0.004005 | None |
| Row 10 | sentiment-analysis-v1 | Review:terribleproduct.com | "What are the main reasons..." | "The main reasons for anal..." | 2024-01-20T10:05:21888 | 0.7 | 0.4 | 0.8 | 1 | 6 | 314 | 0.004005 | None |
| Row 11 | sentiment-analysis-v1 | Review:terribleproduct.com | "You are analyzing apprais..." | "Analyzing appraisals v1..." | 2024-01-20T10:05:21888 | 0.9 | 0.2 | 0.8 | 0.8 | 4 | 280 | 0.003096 | None |
| Row 12 | sentiment-analysis-v1 | Review:terribleproduct.com | "What challenges are you facing?" | "Challenges associated with..." | 2024-01-20T10:05:21888 | 0.9 | 0.2 | 0.8 | 0.8 | 4 | 280 | 0.003096 | None |
| Row 13 | sentiment-analysis-v1 | Review:terribleproduct.com | "You have suggested null..." | "Suggested null (00)..." | 2024-01-20T10:05:21888 | 0.9 | 0.2 | 1 | 0.4 | 3 | 262 | 0.004005 | None |
| Row 14 | sentiment-analysis-v1 | Review:terribleproduct.com | "What is the significance of..." | "Significance associated with..." | 2024-01-20T10:05:21888 | 0.9 | 0.8 | 0.8 | 0.8 | 4 | 285 | 0.003075 | None |
| Row 15 | sentiment-analysis-v1 | Review:terribleproduct.com | "You do variable device v..." | "Variable device v1..." | 2024-01-20T10:05:21888 | 0.9 | 0.8 | 0.8 | 0.8 | 4 | 280 | 0.003096 | None |
| Row 16 | sentiment-analysis-v1 | Review:terribleproduct.com | "What did you do machine v..." | "Machine learning play v..." | 2024-01-20T10:05:21888 | 0.2 | 0.8 | 0.8 | 0.8 | 4 | 281 | 0.003075 | None |
| Row 17 | sentiment-analysis-v1 | Review:terribleproduct.com | "You do accelerometers b..." | "Accelerometers aid in d..." | 2024-01-20T10:05:21888 | 1 | 0.4 | 1 | 0.8000000000000000 | 4 | 287 | 0.004005 | None |
| Row 18 | sentiment-analysis-v1 | Review:terribleproduct.com | "What privacy concerns are..." | "Privacy concerns that are..." | 2024-01-20T10:05:21888 | 0.8 | 0.4 | 0.8 | 0.8 | 4 | 275 | 0.004005 | None |
| Row 19 | sentiment-analysis-v1 | Review:terribleproduct.com | "You are analyzing apprais..." | "Analyzing appraisals v1..." | 2024-01-20T10:05:21888 | 0.9 | 0.8 | 0.8 | 0.8 | 4 | 280 | 0.003096 | None |
| Row 20 | sentiment-analysis-v1 | Review:terribleproduct.com | "What are the main reasons..." | "The main reasons for anal..." | 2024-01-20T10:05:21888 | 0.7 | 0.4 | 0.8 | 0.8 | 4 | 281 | 0.004005 | None |
| Row 21 | sentiment-analysis-v1 | Review:terribleproduct.com | "You are analyzing apprais..." | "Analyzing appraisals v1..." | 2024-01-20T10:05:21888 | 0.9 | 0.2 | 0.8 | 0.7 | 3 | 250 | 0.003096 | None |
| Row 22 | sentiment-analysis-v1 | Review:terribleproduct.com | "What challenges are you facing?" | "Challenges associated with..." | 2024-01-20T10:05:21888 | 0.9 | 0.2 | 0.8 | 0.8 | 4 | 280 | 0.003096 | None |
| Row 23 | sentiment-analysis-v1 | Review:terribleproduct.com | "You have suggested null..." | "Suggested null (00)..." | 2024-01-20T10:05:21888 | 0.9 | 0.2 | 0.8 | 0.75 | 3 | 261 | 0.004005 | None |
| Row 24 | sentiment-analysis-v1 | Review:terribleproduct.com | "What is the significance of..." | "Significance associated with..." | 2024-01-20T10:05:21888 | 0.7 | 0.7 | 1 | 0.6000000000000000 | 4 | 287 | 0.003075 | None |
| Row 25 | sentiment-analysis-v1 | Review:terribleproduct.com | "You do variable device v..." | "Variable device v1..." | 2024-01-20T10:05:21888 | 1 | 0.7 | 1 | 0.6000000000000000 | 5 | 312 | 0.004005 | None |
| Row 26 | sentiment-analysis-v1 | Review:terribleproduct.com | "What did you do machine v..." | "Machine learning play v..." | 2024-01-20T10:05:21888 | 0.2 | 1 | 1 | 0.8 | 4 | 281 | 0.003075 | None |
| Row 27 | sentiment-analysis-v1 | Review:terribleproduct.com | "You do accelerometers b..." | "Accelerometers help in m..." | 2024-01-20T10:05:21888 | 0.9 | 0.2 | 0.8 | 0.8 | 4 | 280 | 0.003096 | None |
| Row 28 | sentiment-analysis-v1 | Review:terribleproduct.com | "What are you doing app..." | "What are you doing app..." | 2024-01-20T10:05:21888 | 0.9 | 0.8 | 1 | 1 | 4 | 280 | 0.003096 | None |
| Row 29 | sentiment-analysis-v1 | Review:terribleproduct.com | "You do location setting..." | "Location setting control..." | 2024-01-20T10:05:21888 | 1 | 0.4 | 1 | 0.75 | 7 | 275 | 0.004005 | None |
| Row 30 | sentiment-analysis-v1 | Review:terribleproduct.com | "What are the main reasons..." | "The main reasons for anal..." | 2024-01-20T10:05:21888 | 0.7 | 0.4 | 0.8 | 1 | 6 | 314 | 0.004005 | None |
| Row 31 | sentiment-analysis-v1 | Review:terribleproduct.com | "You are analyzing apprais..." | "Analyzing appraisals v1..." | 2024-01-20T10:05:21888 | 0.9 | 0.2 | 0.8 | 0.8 | 4 | 280 | 0.003096 | None |
| Row 32 | sentiment-analysis-v1 | Review:terribleproduct.com | "What challenges are you facing?" | "Challenges associated with..." | 2024-01-20T10:05:21888 | 0.9 | 0.2 | 0.8 | 0.8 | 4 | 280 | 0.003096 | None |
| Row 33 | sentiment-analysis-v1 | Review:terribleproduct.com | "You have suggested null..." | "Suggested null (00)..." | 2024-01-20T10:05:21888 | 0.9 | 0.2 | 1 | 0.4 | 3 | 262 | 0.004005 | None |
| Row 34 | sentiment-analysis-v1 | Review:terribleproduct.com | "What is the significance of..." | "Significance associated with..." | 2024-01-20T10:05:21888 | 0.9 | 0.8 | 0.8 | 0.8 | 4 | 285 | 0.003075 | None |
| Row 35 | sentiment-analysis-v1 | Review:terribleproduct.com | "You do variable device v..." | "Variable device v1..." | 2024-01-20T10:05:21888 | 0.9 | 0.8 | 0.8 | 0.8 | 4 | 280 | 0.003096 | None |
| Row 36 | sentiment-analysis-v1 | Review:terribleproduct.com | "What did you do machine v..." | "Machine learning play v..." | 2024-01-20T10:05:21888 | 0.2 | 0.8 | 0.8 | 0.8 | 4 | 281 | 0.003075 | None |
| Row 37 | sentiment-analysis-v1 | Review:terribleproduct.com | "You do accelerometers b..." | "Accelerometers help in m..." | 2024-01-20T10:05:21888 | 0.9 | 0.8 | 1 | 1 | 4 | 280 | 0.004005 | None |
| Row 38 | sentiment-analysis-v1 | Review:terribleproduct.com | "What are you doing app..." | "What are you doing app..." | 2024-01-20T10:05:21888 | 0.9 | 0.8 | 1 | 1 | 4 | 280 | 0.003096 | None |
| Row 39 | sentiment-analysis-v1 | Review:terribleproduct.com | "You do location setting..." | "Location setting control..." | 2024-01-20T10:05:21888 | 1 | 0.4 | 1 | 0.75 | 7 | 275 | 0.004005 | None |
| Row 40 | sentiment-analysis-v1 | Review:terribleproduct.com | "What are the main reasons..." | "The main reasons for anal..." | 2024-01-20T10:05:21888 | 0.7 | 0.4 | 0.8 | 0.8 | 4 | 281 | 0.004005 | None |
| Row 41 | sentiment-analysis-v1 | Review:terribleproduct.com | "You are analyzing apprais..." | "Analyzing appraisals v1..." | 2024-01-20T10:05:21888 | 0.9 | 0.2 | 0.8 | 0.8 | 4 | 280 | 0.003096 | None |
| Row 42 | sentiment-analysis-v1 | Review:terribleproduct.com | "What challenges are you facing?" | "Challenges associated with..." | 2024-01-20T10:05:21888 | 0.9 | 0.2 | 0.8 | 0.8 | 4 | 280 | 0.003096 | None |
| Row 43 | sentiment-analysis-v1 | Review:terribleproduct.com | "You have suggested null..." | "Suggested null (00)..." | 2024-01-20T10:05:21888 | 0.9 | 0.2 | 1 | 0.4 | 3 | 262 | 0.004005 | None |
| Row 44 | sentiment-analysis-v1 | Review:terribleproduct.com | "What is the significance of..." | "Significance associated with..." | 2024-01-20T10:05:21888 | 0.9 | 0.8 | 0.8 | 0.8 | 4 | 285 | 0.003075 | None |
| Row 45 | sentiment-analysis-v1 | Review:terribleproduct.com | "You do variable device v..." | "Variable device v1..." | 2024-01-20T10:05:21888 | 0.9 | 0.8 | 0.8 | 0.8 | 4 | 280 | 0.003096 | None |
| Row 46 | sentiment-analysis-v1 | Review:terribleproduct.com | "What did you do machine v..." | "Machine learning play v..." | 2024-01-20T10:05:21888 | 0.2 | 0.8 | 0.8 | 0.8 | 4 | 281 | 0.003075 | None |
| Row 47 | sentiment-analysis-v1 | Review:terribleproduct.com | "You do accelerometers b..." | "Accelerometers aid in d..." | 2024-01-20T10:05:21888 | 1 | 0.4 | 1 | 0.8000000000000000 | 4 | 287 | 0.004005 | None |
| Row 48 | sentiment-analysis-v1 | Review:terribleproduct.com | "What privacy concerns are..." | "Privacy concerns that are..." | 2024-01-20T10:05:21888 | 0.8 | 0.4 | 0.8 | 0.8 | 4 | 275 | 0.004005 | None |
| Row 49 | sentiment-analysis-v1 | Review:terribleproduct.com | "You are analyzing apprais..." | "Analyzing appraisals v1..." | 2024-01-20T10:05:21888 | 0.9 | 0.8 | 0.8 | 0.8 | 4 | 280 | 0.003096 | None |
| Row 50 | sentiment-analysis-v1 | Review:terribleproduct.com | "What are the main reasons..." | "The main reasons for anal..." | 2024-01-20T10:05:21888 | 0.7 | 0.4 | 0.8 | 0.8 | 4 | 281 | 0.004005 | None |
| Row 51 | sentiment-analysis-v1 | Review:terribleproduct.com | "You are analyzing apprais..." | "Analyzing appraisals v1..." | 2024-01-20T10:05:21888 | 0.9 | 0.2 | 0.8 | 0.8 | 4 | 280 | 0.003096 | None |
| Row 52 | sentiment-analysis-v1 | Review:terribleproduct.com | "What challenges are you facing?" | "Challenges associated with..." | 2024-01-20T10:05:21888 | 0.9 | 0.2 | 0.8 | 0.8 | 4 | 280 | 0.003096 | None |
| Row 53 | sentiment-analysis-v1 | Review:terribleproduct.com | "You have suggested null..." | "Suggested null (00)..." | 2024-01-20T10:05:21888 | 0.9 | 0.2 | 1 | 0.4 | 3 | 262 | 0.004005 | None |
| Row 54 | sentiment-analysis-v1 | Review:terribleproduct.com | "What is the significance of..." | "Significance associated with..." | 2024-01-20T10:05:21888 | 0.9 | 0.8 | 0.8 | 0.8 | 4 | 285 | 0.003075 | None |
| Row 55 | sentiment-analysis-v1 | Review:terribleproduct.com | "You do variable device v..." | "Variable device v1..." | 2024-01-20T10:05:21888 | 0.9 | 0.8 | 0.8 | 0.8 | 4 | 280 | 0.003096 | None |
| Row 56 | sentiment-analysis-v1 | Review:terribleproduct.com | "What did you do machine v..." | "Machine learning play v..." | 2024-01-20T10:05:21888 | 0.2 | 0.8 | 0.8 | 0.8 | 4 | 281 | 0.003075 | None |
| Row 57 | sentiment-analysis-v1 | Review:terribleproduct.com | "You do accelerometers b..." | "Accelerometers help in m..." | 2024-01-20T10:05:21888 | 0.9 | 0.8 | 1 | 1 | 4 | 280 | 0.004005 | None |
| Row 58 | sentiment-analysis-v1 | Review:terribleproduct.com | "What are you doing app..." | "What are you doing app..." | 2024-01-20T10:05:21888 | 0.9 | 0.8 | 1 | 1 | 4 | 280 | 0.003096 | None |
| Row 59 | sentiment-analysis-v1 | Review:terribleproduct.com | "You do location setting..." | "Location setting control..." | 2024-01-20T10:05:21888 | 1 | 0.4 | 1 | 0.75 | 7 | 275 | 0.004005 | None |
| Row 60 | sentiment-analysis-v1 | Review:terribleproduct.com | "What are the main reasons..." | "The main reasons for anal..." | 2024-01-20T10:05:21888 | 0.7 | 0.4 | 0.8 | 0.8 | 4 | 281 | 0.004005 | None |
| Row 61 | sentiment-analysis-v1 | Review:terribleproduct.com | "You are analyzing apprais..." | "Analyzing appraisals v1..." | 2024-01-20T10:05:21888 | 0.9 | 0.2 | 0.8 | 0.8 | 4 | 280 | 0.003096 | None |
| Row 62 | sentiment-analysis-v1 | Review:terribleproduct.com | "What challenges are you facing?" | "Challenges associated with..." | 2024-01-20T10:05:21888 | 0.9 | 0.2 | 0.8 | 0.8 | 4 | 280 | 0.003096 | None |
| Row 63 | sentiment-analysis-v1 | Review:terribleproduct.com | "You have suggested null..." | "Suggested null (00)..." | 2024-01-20T10:05:21888 | 0.9 | 0.2 | 1 | 0.4 | 3 | 262 | 0.004005 | None |
| Row 64 | sentiment-analysis-v1 | Review:terribleproduct.com | "What is the significance of..." | "Significance associated with..." | 2024-01-20T10:05:21888 | 0.9 | 0.8 | 0.8 | 0.8 | 4 | 285 | 0.003075 | None |
| Row 65 | sentiment-analysis-v1 | Review:terribleproduct.com | "You do variable device v..." | "Variable device v1..." | 2024-01-20T10:05:21888 | 0.9 | 0.8 | 0.8 | 0.8 | 4 | 280 | 0.003096 | None |
| Row 66 | sentiment-analysis-v1 | Review:terribleproduct.com | "What did you do machine v..." | "Machine learning play v..." | 2024-01-20T10:05:21888 | 0.2 | 0.8 | 0.8 | 0.8 | 4 | 281 | 0.003075 | None |
| Row 67 | sentiment-analysis-v1 | Review:terribleproduct.com | "You do accelerometers b..." | "Accelerometers aid in d..." | 2024-01-20T10:05:21888 | 1 | 0.4 | 1 | 0.8000000000000000 | 4 | 287 | 0.004005 | None |
| Row 68 | sentiment-analysis-v1 | Review:terribleproduct.com | "What privacy concerns are..." | "Privacy concerns that are..." | 2024-01-20T10:05:21888 | 0.8 | 0.4 | 0.8 | 0.8 | 4 | 275 | 0.004005 | None |
| Row 69 | sentiment-analysis-v1 | Review:terribleproduct.com | "You are analyzing apprais..." | "Analyzing appraisals v1..." | 2024-01-20T10:05:21888 | 0.9 | 0.8 | 0.8 | 0.8 | 4 | 280 | 0.003096 | None |
| Row 70 | sentiment-analysis-v1 | Review:terribleproduct.com | "What are the main reasons..." | "The main reasons for anal..." | 2024-01-20T10:05:21888 | 0.7 | 0.4 | 0.8 | 0.8 | 4 | 281 | 0.004005 | None |
| Row 71 | sentiment-analysis-v1 | Review:terribleproduct.com | "You are analyzing apprais..." | "Analyzing appraisals v1..." | 2024-01-20T10:05:21888 | 0.9 | 0.2 | 0.8 | 0.8 | 4 | 280 | 0.003096 | None |
| Row 72 | sentiment-analysis-v1 | Review:terribleproduct.com | "What challenges are you facing?" | "Challenges associated with..." | 2024-01-20T10:05:21888 | 0.9 | 0.2 | 0.8 | 0.8 | 4 | 280 | 0.003096 | None |
| Row 73 | sentiment-analysis-v1 | Review:terribleproduct.com | "You have suggested null..." | "Suggested null (00)..." | 2024-01-20T10:05:21888 | 0.9 | 0.2 | 1 | 0.4 | 3 | 262 | 0.004005 | None |
| Row 74 | sentiment-analysis-v1 | Review:terribleproduct.com | "What is the significance of..." | "Significance associated with..." | 2024-01-20T10:05:21888 | 0.9 | 0.8 | 0.8 | 0.8 | 4 | 285 | 0.003075 | None |
| Row 75 | sentiment-analysis-v1 | Review:terribleproduct.com | "You do variable device v..." | "Variable device v1..." | 2024-01-20T10:05:21888 | 0.9 | 0.8 | 0.8 | 0.8 | 4 | 280 | 0.003096 | None |
| Row 76 | sentiment-analysis-v1 | Review:terribleproduct.com | "What did you do machine v..." | "Machine learning play v..." | 2024-01-20T10:05:21888 | 0.2 | 0.8 | 0.8 | 0.8 | 4 | 281 | 0.003075 | None |
| Row 77 | sentiment-analysis-v1 | Review:terribleproduct.com | "You do accelerometers b..." | "Accelerometers help in m..." | 2024-01-20T10:05:21888 | 0.9 | 0.8 | 1 | 1 | 4 | 280 | 0.004005 | None |
| Row 78 | sentiment-analysis-v1 | Review:terribleproduct.com | "What are you doing app..." | "What are you doing app..." | 2024-01-20T10:05:21888 | 0.9 | 0.8 | 1 | 1 | 4 | 280 | 0.003096 | None |
| Row 79 | sentiment-analysis-v1 | Review:terribleproduct.com | "You do location setting..." | "Location setting control..." | 2024-01-20T10:05:21888 | 1 | 0.4 | 1 | 0.75 | 7 | 275 | 0.004005 | None |
| Row 80 | sentiment-analysis-v1 | Review:terribleproduct.com | "What are the main reasons..." | "The main reasons for anal..." | 2024-01-20T10:05:21888 | 0.7 | 0.4 | 0.8 | 0.8 | 4 | 281 | 0.004005 | None |
| Row 81 | sentiment-analysis-v1 | Review:terribleproduct.com | "You are analyzing apprais..." | "Analyzing appraisals v1..." | 2024-01-20T10:05:21888 | 0.9 | 0.2 | 0.8 | 0.8 | 4 | 280 | 0.003096 | None |
| Row 82 | sentiment-analysis-v1 | Review:terribleproduct.com | "What challenges are you facing?" | "Challenges associated with..." | 2024-01-20T10:05:21888 | 0.9 | 0.2 | 0.8 | 0.8 | 4 | 280 | 0.003096 | None |
| Row 83 | sentiment-analysis-v1 | Review:terribleproduct.com | "You have suggested null..." | "Suggested null (00)..." | 2024-01-20T10:05:21888 | 0.9 | 0.2 | 1 | 0.4 | 3 | 262 | 0.004005 | None |
| Row | | | | | | | | | | | | | |

Note: The second image shows the output of each model for each test prompt and I studied these in detail to ensure the responses were accurate to those in the notes. The folder containing all the spreadsheets containing the results of these runs of the separate prototypes can be found [here](#).

The next step I took was to try a different approach which involved fetching smaller chunks during retrieval first, then reference the parent IDs, and return the bigger chunks. This was an approach that I came across

during my research and thought it would be an interesting one to trial. The idea is that this dual-level chunking strategy allows me to refine my model's ability to generate responses that are deeply informative and contextually grounded. During the initial stages of retrieving my data I set a fixed chunk size of 1024. Later I then split these into sub chunk sizes of 128, 256 and 512. This involved breaking down larger text blocks into smaller chunks of various sizes and indexing these for retrieval. I used a RecursiveRetriever so my system could navigate between these chunks, ensuring that the retrieved information was not only accurate but also contextually comprehensive.

▼ Chunk References: Smaller Child Chunks Referring to Bigger Parent Chunk

```
1 sub_chunk_sizes = [128, 256, 512]
2 sub_node_parsers = [
3     SimpleNodeParser.from_defaults(chunk_size=c) for c in sub_chunk_sizes
4 ]
5
6 all_nodes = []
7 for base_node in base_nodes:
8     for n in sub_node_parsers:
9         sub_nodes = n.get_nodes_from_documents([base_node])
10        sub_inodes = [
11            IndexNode.from_text_node(sn, base_node.node_id) for sn in sub_nodes
12        ]
13        all_nodes.extend(sub_inodes)
14
15 # also add original node to node
16 original_node = IndexNode.from_text_node(base_node, base_node.node_id)
17 all_nodes.append(original_node)
```

[] 1 all_nodes_dict = {n.node_id: n for n in all_nodes}

RAG 9 - Chunks

| Records | Average Latency (Seconds) | Total Cost (USD) | Total Tokens | Answer Correctness | Context Relevance | language_match | Answer Relevance | Groundedness |
|---------|---------------------------|------------------|--------------|--------------------|-------------------|----------------|------------------|------------------|
| 10 | 4.6 | \$0.01 | 3.77k | 0.85
✔ high | 0.72
⚠ medium | 0.92
✔ high | 0.93
✔ high | 0.69
⚠ medium |

The results showed an improvement in groundedness but overall I was quite disappointed with these results. For this I also included the Huggingface metric of language match to assess the language being used by my prototype as I thought this could be an interesting metric to include. Time permitting I would have liked to explore this avenue further and try to see why and where I perhaps went wrong as I felt there should've been more improvements in my overall score while using this approach.

Step 7: Select best model and run on GPT-4

The final step I decided to take was to select the prototype which was performing best and try to run this using GPT4 instead of GPT3.5 Turbo as my LLM. This was something I decided to try after the deadline got extended as I was interested to analyse if there would be a significant difference in results. After assessing my current prototypes across all the different steps, the most well rounded RAG agent was the prototype with a SentenceWindowNodeParser with a window of 6 and also used the system prompt.

RAG 7 - sentence window of 6 plus simple prompt ©

| Records | Average Latency (Seconds) | Total Cost (USD) | Total Tokens | Answer Relevance | Context Relevance | Groundedness | Answer Correctness |
|---------|---------------------------|------------------|--------------|------------------|-------------------|------------------|--------------------|
| 10 | 4.4 | \$0 | 3.16k | 0.94
✔ high | 0.79
⚠ medium | 0.73
⚠ medium | 0.84
✔ high |

During the evaluation process I changed the system prompt eight times, experimented with removing stop words, tried a chunking method, tried a number of different window sizes and eventually this was the prototype I selected. As a final step, I decided I wanted to use GPT-4 as my model and assess any changes that I noticed. The results were as follows.

RAG 7 - sentence window of 6 plus simple prompt ®

| Records | Average Latency (Seconds) | Total Cost (USD) | Total Tokens | Answer Relevance | Context Relevance | Groundedness | Answer Correctness |
|---------|---------------------------|------------------|--------------|------------------|-------------------|------------------|--------------------|
| 10 | 4.4 | \$0 | 3.16k | 0.93
■ high | 0.79
▲ medium | 0.76
▲ medium | 0.89
■ high |

Here I observed improvements in both Groundedness and Answer Correctness, with only a minor dip in Answer Relevance. This indicates a positive progression in the model's performance which was what I had expected but was still interesting to observe. This will act as my final LLM-based Agent for Reviewing Lecture Material based on RAG.

Conclusion

Overall, I found this assignment quite challenging but also very rewarding. There was an awful lot of work involved both in building and evaluating the prototypes but also in researching the overall topic and how it works. I found the iterative nature of the evaluation section quite challenging. The constant iterations and testing of prototypes were not only time-consuming but also demanded significant computational resources. Despite these challenges, the process was incredibly educational, enhancing my understanding of the field. If given the opportunity to revisit the project, I would allow more time to explore other methods for refining the model and investigating other possible techniques. I regret dedicating excessive time to attempting to use Docker and troubleshooting lambda models without opting for an OpenAI plan upgrade earlier. Moving forward, I would prioritise exploring the chunking method further as I feel this could definitely be a solid avenue to follow. Sections such as creating the questions or even just initialising Milvus correctly were all quite time consuming. I do however feel that it was an incredibly different assignment from our regular assessments and one I learned a lot from.

Future Suggestions

This assignment was a significant learning experience, demanding thorough research and deep understanding to accomplish. I definitely feel it is an assignment that should be used as it is very interesting and students will learn a lot from it. Reflecting on it, I believe it deserves a larger portion of the total marks given its complexity and the extensive time required. An additional week or two could substantially improve the quality of the RAG system developed. A considerable amount of time went into configuring setups and crafting questions, which, while time-consuming, was crucial for gaining a hands-on understanding of the underlying technologies. To address this, extending the timeline to allow for mistakes and following dead end paths and increasing the assignment's weight could offer students more time to spend on developing a refined RAG system rather than using large amounts of the time on initial setup.