**ENPM 808 Independent Study Project**

**RAGRate**

**By: Pavan Sainath Atukuri**

**UID: 119397428**

**Index**

1. **What is RAGRate and Why do we need it?**
2. **What is RAG and How does this works?**
3. **Components in a RAG**
4. **Evaluation Metrics**
5. **Prompt Engineering**
6. **Evaluation Framework Implementation**
7. **Testing Against a Custom RAG**
8. **What is ReAct Prompting**
9. **Results**
10. **Future Work and Conclusion**
11. **References**
12. **What is RAGRate why do we need it?**

RAGRate is a comprehensive automated evaluation framework designed to assess the performance of Retrieval-Augmented Generation (RAG) models, which combine language models with external knowledge retrieval capabilities. RAG models have gained significant traction in recent years, as they enable language models to leverage external information sources, leading to more reliable and factual responses.

**The need for a framework like RAGRate arises from several key factors:**

**Growing Adoption of RAG Applications:** With an increasing number of enterprises and organizations adopting RAG applications to enhance their language models' capabilities, there is a pressing need for robust evaluation tools. RAGRate provides a systematic approach to evaluate the performance of RAG models, enabling organizations to make informed decisions about their deployment and optimization.

**Boosting Confidence and Faithfulness:** RAG applications heavily rely on the retrieval and integration of external knowledge sources. However, the quality and relevance of the retrieved information can significantly impact the model's outputs. RAGRate aims to boost the confidence and faithfulness of RAG applications by evaluating the Context Relevance, Faithfulness, and Accuracy of the generated responses. This evaluation helps identify potential weaknesses or biases in the knowledge retrieval process, allowing for targeted improvements.

**Finetuning and Optimization:** RAG models often require extensive finetuning and optimization to achieve optimal performance across different domains and tasks. RAGRate's comprehensive evaluation metrics provide valuable insights into the strengths and limitations of different RAG approaches, prompting techniques, and model configurations. These insights can guide the finetuning process, enabling developers to make informed decisions about model selection, prompting strategies, and parameter tuning.

**Benchmarking and Comparison:** With multiple open-source and proprietary RAG models available, RAGRate offers a standardized framework for benchmarking and comparing their performance. By evaluating different models under consistent conditions and metrics, the framework facilitates objective comparisons, promoting the development and adoption of more robust and reliable RAG solutions.

**Explainability and Trust:** RAG models' ability to leverage external knowledge sources can enhance the explainability and trustworthiness of their outputs. RAGRate's evaluation metrics, particularly the Context Relevance and Faithfulness measures, provide insights into the model's reasoning process and the alignment between the retrieved knowledge and the generated responses. This transparency can foster trust in RAG applications, enabling stakeholders to understand and interpret the model's behavior better.

By providing a comprehensive and systematic evaluation framework, RAGRate aims to support the growing adoption of RAG applications across various industries and domains. Its metrics and insights can contribute to the development of more reliable, faithful, and trustworthy RAG models, ultimately enhancing the capabilities of language models and their ability to leverage external knowledge sources effectively.

1. **What is RAG and How does this work?**

RAG, which stands for "Retrieval-Augmented Generation," is a type of neural architecture used in natural language processing (NLP) tasks. It combines the strengths of retrieval-based and generation-based models to produce more accurate and informative outputs. The RAG architecture works as follows

First, the retrieval component of the model searches through a large corpus of text (e.g., Wikipedia) to find relevant passages related to the input query or context. This step is similar to how a search engine retrieves relevant documents based on a user's search query

Next, the retrieved passages are fed into the generation component, which is typically a large language model trained on a vast amount of text data. The language model then generates an output response based on the input query and the retrieved passages, effectively augmenting its knowledge with the relevant information from the corpus.

The key advantage of RAG is that it allows language models to leverage external knowledge sources, mitigating the limitations of their fixed training data. By incorporating relevant information from the retrieval corpus, RAG models can produce more accurate and informative responses, especially for queries that require specific factual knowledge or domain expertise.

**Here's a step-by-step breakdown of how RAG works:**

1. **Input:** The model receives an input query or context from the user.

2. **Retrieval:** The retrieval component searches through the large corpus of text (e.g., Wikipedia) to find the most relevant passages related to the input.

3. **Context Selection:** The top-k most relevant passages are selected and passed on to the generation component.

4. **Generation:** The generation component, which is a large language model, takes the input query and the selected context as input. It then generates an output response by conditioning on both the input and the retrieved context.

5. **Output**: The generated response, which incorporates relevant information from the retrieved context, is presented to the user.

The combination of retrieval and generation components in RAG allows the model to leverage the strengths of both approaches. The retrieval component provides access to a vast knowledge base, while the generation component can produce responses that are made to the specific input query.

A diagram of a search engine

Description automatically generated

1. **Components in a RAG based application**

**Retriever**: This component is responsible for retrieving relevant passages or documents from a large corpus of data (e.g., Wikipedia, databases, or other text sources) based on the input query.

This Retriever also converts the documents into chunks and then converts the chunks into embeddings. The retriever also helps to convert the chunks into vector embeddings and store them in a vector database.

**Generator**: This component uses the retrieved context along with the input query to generate a final output response, typically using a language model or a sequence-to-sequence model.

1. **EVALUATION METRICS**

Several evaluation metrics can be inferred for RAG-based applications. However, in this framework, we are concentrating only on a few important metrics that will help us evaluate the performance of our RAG application.

The metrics that we are covering are

1. **Context Relevance:** In the context of evaluating Retrieval-Augmented Generation (RAG) models, context relevance refers to the degree to which the retrieved context (passages or documents) from the corpus is relevant and useful for generating the desired output or response. Evaluating context relevance is crucial in RAG models because the quality of the retrieved context directly impacts the performance of the generation component and, consequently, the overall quality of the generated output.

We calculate Context Relevance using a prompt that verifies if the context was useful in arriving at the given answer on a scale of 0 to 10 where 0 means the context is least relevant to answer and 10 is the most relevant to the answer. More details about the prompt are in the below sections.

1. **Answer Correctness**: Answer correctness refers to the degree to which the generated output or response accurately addresses the input query or prompt.

Context Relevance directly impacts the Answer's Correctness.

In the RAGRate framework, the accuracy of the generated answers is evaluated by comparing them against the ground truth or correct answers. The evaluation process is facilitated by employing a language model itself as a judge, leveraging its natural language understanding capabilities.

The accuracy evaluation is performed using a Chain-of-Thought (CoT) prompting technique, where the language model is provided with a detailed prompt that guides its reasoning process. The prompt explains the task of assessing the accuracy of answers generated by an RAG-based LLM application, considering the provided question, answer, context, and ground truth. The language model is instructed to rate the accuracy of the generated answer on a scale of 1 to 10. The criteria are explained in the following sections.

1. **Faithfulness:**  Faithfulness measures the factual consistency of the generated response against the retrieved context.

It assesses whether the claims or statements made in the generated answer can be reasonably inferred or supported by the information present in the provided context.

The evaluation of faithfulness is facilitated by employing a language model as an evaluation expert, guided by a detailed prompt that outlines a step-by-step process. The prompt is more detailed in the next section.

**Faithfulness score** = (Number of claims supported by context) / (Total claims in answer)

1. **PROMPT ENGINEERING**

There are several ways to evaluate RAG-based applications. But taking into account the research conducted prior, Making LLM a judge to evaluate the answers generated by LLM is a scalable and more accurate solution. I have attached the screenshot of the performance of different ways for evaluating RAG applications.

"**scalable**" refers to the ability of an evaluation technique to be applied on a large scale or to handle a significant amount of data or tasks efficiently. The y-axis represents the scalability of different evaluation techniques.

"**Meaningful**," on the other hand, refers to how meaningful or informative the evaluation results are in assessing the performance and capabilities of the language models (LLMs) being evaluated.A diagram of a diagram

Description automatically generated with medium confidence

In the RAGRate framework, we leverage a large language model (LLM) as an evaluation expert or judge. This involves querying the LLM with a carefully crafted prompt that outlines the evaluation task and provides guidelines for assessing the accuracy, faithfulness, and context relevance of the generated responses. The LLM processes this prompt, along with the input data (question, answer, context, and ground truth), and generates an evaluation response following the specified format. The evaluation process does not involve passing results between multiple LLMs; instead, a single LLM is utilized as the judge, leveraging its natural language understanding capabilities to provide interpretable and contextualized evaluations. So, I have conducted several attempts of research using Prompt Engineering on LLMs to see the results and attached the prompts for calculating Context Relevance, Answer correctness, Faithfulness.

As I decided to have LLM as a judge for the evaluation framework, I employed 3 different techniques in prompt engineering to evaluate the metrics of the RAG application using the questions, answers, context, and ground truth.  
  
3 different techniques employed are  
  
**Zero Shot Prompting:**

Zero-shot prompting refers to the ability of large language models to perform new tasks without any task-specific training data, relying solely on the general knowledge acquired during pre-training.

Eg: "Explain the photosynthesis process in plants”

Below are the different prompts that I used to measure different metrics.

**Few Shot Prompting:**

Few-shot prompting involves providing a small number of examples or demonstrations to the language model as part of the prompt, to guide its understanding of the task at hand.

**Example:**

"The movie was fantastic! 😀" - Positive sentiment “  
I really disliked that restaurant. 👎" - Negative sentiment

**Chain of Thoughts Prompting:**

Chain-of-Thought (CoT) prompting enables complex reasoning capabilities through intermediate reasoning steps. You can combine it with few-shot prompting to get better results on more complex tasks that require reasoning before responding.

**Context Relevance:**  
When designing prompts to evaluate the context relevance in RAGRate, the following key aspects make the prompts effective:

**Clearly defined task:** The prompt clearly states that the language model is expected to grade the relevance of a given context to a given question on a scale of 0 to 10. This explicit task definition ensures that the language model understands its objective.

**Step-by-step instructions:** The prompt provides detailed step-by-step instructions for the language model to follow. These instructions guide the language model through the evaluation process, ensuring a structured and consistent approach.

**Score range and guidelines:** The prompt provides a clear scoring range from 0 to 10, along with guidelines for assigning scores within specific ranges. For example, a score of 2-4 is suggested if the context is relevant to some parts of the question, while a score of 9-10 is recommended if the context is relevant to the entire question and helpful for answering it.

**Examples:** The prompt includes examples of question-context pairs along with the expected scores and reasoning. These examples serve as references for the language model, helping it understand how to apply the evaluation criteria and provide supporting evidence for its scores.

**Response format:** The prompt specifies a structured format for the language model to provide its response. This format includes a section for "Supporting Evidence," where the language model is expected to explain its reasoning for the assigned score, and a section for the numerical "Score" itself.

**While Employing Zero Shot Prompting:**

Prompt used:

relevance\_prompt\_zero\_shot=('''  
  
You are an LLM expert trained to grade the relevance of a given context to a given question on a scale of 0 to 10. Your task is to evaluate how relevant the provided context is to the given question, considering the following criteria:  
  
- Long and short contexts should be scored equally.  
- The more relevant the context is to the question, the higher the score.  
- The more parts of the question the context is relevant to, the higher the score.  
- Provide a score of 2-4 if the context is relevant to some parts of the question.  
- Provide a score of 5-8 if the context is relevant to most parts of the question.  
- Provide a score of 9-10 if the context is relevant to the entire question.  
- A score of 10 should be given only if the context is relevant and helpful for answering the entire question.  
  
Your response should follow the specified format:  
<Format>  
Supporting Evidence: <Explain your reasoning for the assigned score, including how the context is relevant or irrelevant to the question>  
Score: <The relevance score from 0 to 10 based on the given criteria>  
  
'''

**While Employing Few Shot+ Chain of Thought Prompting:**

relevance\_prompt\_CoT=('''You are an LLM expert trained to grade the relevance of a given context to a given question on a scale of 0 to 10. Follow these steps:  
  
1. Read the provided QUESTION and CONTEXT carefully.  
2. Evaluate how relevant the context is to the question based on the following criteria:  
 - Long and short contexts should be scored equally.  
 - The more relevant the context is to the question, the higher the score.  
 - The more parts of the question the context is relevant to, the higher the score.  
 - Provide a score of 2-4 if the context is relevant to some parts of the question.  
 - Provide a score of 5-8 if the context is relevant to most parts of the question.  
 - Provide a score of 9-10 if the context is relevant to the entire question.  
 - A score of 10 should be given only if the context is relevant and helpful for answering the entire question.  
3. Here are few examples   
Examples:[{ "question": "What can you tell me about albert Albert Einstein?",  
 "context": "Albert Einstein (14 March 1879 – 18 April 1955) was a German-born theoretical physicist, widely held to be one of the greatest and most influential scientists of all time. Best known for developing the theory of relativity, he also made important contributions to quantum mechanics, and was thus a central figure in the revolutionary reshaping of the scientific understanding of nature that modern physics accomplished in the first decades of the twentieth century. His mass–energy equivalence formula E = mc2, which arises from relativity theory, has been called "the world's most famous equation". He received the 1921 Nobel Prize in Physics "for his services to theoretical physics, and especially for his discovery of the law of the photoelectric effect", a pivotal step in the development of quantum theory. His work is also known for its influence on the philosophy of science. In a 1999 poll of 130 leading physicists worldwide by the British journal Physics World, Einstein was ranked the greatest physicist of all time. His intellectual achievements and originality have made Einstein synonymous with genius.",  
 "answer": "Albert Einstein born in 14 March 1879 was German-born theoretical physicist, widely held to be one of the   
 greatest and most influential scientists of all time. He received the 1921 Nobel Prize in Physics for his services to theoretical physics.   
 He published 4 papers in 1905. Einstein moved to Switzerland in 1895",  
 "reason":"The provided context was indeed useful in arriving at the given answer.  
 The context includes key information about Albert Einstein's life and contributions, which are reflected in the answer."  
 "relevancy\_score":10},  
 { "question": "who won 2020 icc world cup?",  
 "context": "The 2022 ICC Men's T20 World Cup, held from October 16 to November 13, 2022, in Australia, was the eighth edition of the tournament. Originally scheduled for 2020, it was postponed due to the COVID-19 pandemic. England emerged victorious, defeating Pakistan by five wickets in the final to clinch their second ICC Men's T20 World Cup title.",  
 "answer": "England",   
 "reason":"the context was useful in clarifying the situation regarding the 2020 ICC World Cup and indicating that England was the winner of the tournament that was intended to be held in 2020 but actually took place in 2022.",}]  
 "relevancy\_score":10}]  
4. Provide your response in the following format:  
  
Supporting Evidence: <Explain your reasoning for the score>  
Score: <The score from 0 to 10 based on the criteria>''')

**Faithfulness:**

The prompt is designed to do the following steps.

**Identify Claims:** Carefully read the generated answer and identify all its claims or statements.

**Analyze Context:** Thoroughly analyze the given context and determine which claims from the generated answer can be reasonably inferred or supported by the information in the context.

Count Claims: Count the total number of claims made in the generated answer.

Count Supported Claims: Count the number of claims from the generated answer that are present in the list of claims supported by the context.

Calculate Faithfulness Score: Calculate the faithfulness score using the formula:

**Faithfulness score** = (Number of claims supported by context) / (Total claims in answer)

By following this step-by-step process and providing the supporting evidence, the RAGRate framework can obtain a reliable and interpretable faithfulness score for the generated answers. The supporting evidence, which includes the list of claims inferred from the context and the list of all claims made in the generated answer, enhances the transparency and explainability of the evaluation process.

The faithfulness metric, along with the other evaluation criteria (Context Relevance and Accuracy), provides valuable insights into the factual consistency of the RAG-based LLM application's responses. These insights can guide further improvements in the knowledge retrieval and integration processes, ultimately enhancing the reliability and trustworthiness of the RAG model.

**While Employed Zero Shot Prompting:**

faithfulness\_prompt\_zero\_shot=('''  
You are an evaluator tasked with measuring the faithfulness of a generated answer against a given context. The faithfulness score is calculated by identifying the claims made in the generated answer, determining which of those claims can be inferred from the given context, and then calculating the ratio of the number of claims inferred from the context to the total number of claims made in the answer.  
  
  
  
Calculate the faithfulness score using the formula: (Number of claims in the generated answer that can be inferred from the given context) / (Total number of claims in the generated answer)  
Your response should follow the specified format:  
  
<Format>  
Score: <The faithfulness score ranging from 0 to 1, with higher scores indicating better factual consistency between the generated answer and the given context>  
Supporting Evidence:  
List of Claims inferred from context: <Actual List of claims from the generated answer that can be inferred from the given context>  
List of claims in the given answer: <Actual List of all claims made in the generated answer>  
</Format>  
Please provide your response in the specified format,   
including the faithfulness score, the list of claims inferred from the context, and the list of all claims made in the generated answer.''')

**While Employed Chain Of Thoughts Prompting:**

faithfulness\_prompt\_COT=('''You are an evalution expert and you will given with the QUESTION, CONTEXT, ANSWER  
  
To evaluate the faithfulness of a generated answer against a given context, let's break down the process into steps with an example:  
  
Example:  
Question: What was the Wright brothers' contribution to aviation?  
Context: The Wright brothers, Orville and Wilbur, were American aviation pioneers credited with inventing, building, and flying the world's first successful motor-operated airplane. They made the first controlled, sustained flight of a powered, heavier-than-air aircraft on December 17, 1903, at Kill Devil Hills, North Carolina.  
Answer: The Wright brothers invented the first successful airplane and made the first powered, controlled flight in 1903.  
  
Step 1: Carefully read the generated ANSWER and identify all the claims or statements made in it.   
Claims in the answer:  
1) The Wright brothers invented the first successful airplane.  
2) The Wright brothers made the first powered, controlled flight in 1903.  
  
Step 2: Next, thoroughly analyze the given context and determine which claims from the generated answer can be reasonably inferred or supported by the information in the context.  
Claims supported by context:   
1) The Wright brothers made the first controlled, sustained flight of a powered, heavier-than-air aircraft on December 17, 1903 (supports claim 2)  
2) The Wright brothers invented, built and flew the world's first successful motor-operated airplane (supports claim 1)  
  
Step 3: Count the total number of claims made in the generated answer.  
Total claims in answer = 2  
  
Step 4: Count the number of claims from the generated answer that are present in the list of claims supported by the context.   
Claims supported by context = 2  
  
Step 5: Calculate the faithfulness score using the formula:  
Faithfulness score = (Number of claims supported by context) / (Total claims in answer)  
 = 2 / 2  
 = 1.0  
  
<Format>  
Score: <The faithfulness score ranging from 0 to 1, with higher scores indicating better factual consistency between the generated answer and the given context>  
Supporting Evidence:  
List of Claims inferred from context: <Actual List of claims from the generated answer that can be inferred from the given context>  
List of claims in the given answer: <Actual List of all claims made in the generated answer>  
</Format>  
Please provide your response in the specified format,   
including the faithfulness score, the list of claims inferred from the context, and the list of all claims made in the generated answer.  
  
By following this step-by-step process with an example and providing the supporting evidence, you can effectively evaluate the faithfulness of the generated answer against the given context.''')

**Accuracy:**

The accuracy evaluation is performed using a different prompting techniques, where the language model is provided with a detailed prompt that guides its reasoning process. The prompt explains the task of assessing the accuracy of answers generated by an RAG-based LLM application, considering the provided question, answer, context, and ground truth.

The language model is instructed to rate the accuracy of the generated answer on a scale of 1 to 10, based on the following criteria:

A score of 1 indicates that the generated answer is completely inaccurate or unrelated to the question.

A score of 5 suggests that the generated answer is partially accurate or relevant but contains significant errors or omissions.

A score of 10 indicates that the generated answer is highly accurate and comprehensive, addressing all aspects of the question correctly.

The prompt includes examples to illustrate the scoring process, showcasing how the language model should evaluate the accuracy based on the relevance of the provided context and the degree to which the generated answer aligns with the ground truth.

accuracy\_prompt\_zero\_shot=('''  
  
You are an evaluator tasked with assessing the accuracy of answers generated by an RAG-based LLM application. You will be provided with the following information:  
  
- Question: The original question asked.  
- Answer: The answer generated by the RAG-based LLM application.  
- Context: The relevant context information used by the RAG-based LLM application.  
- Ground Truth: The correct or "ground truth" answer.  
  
Your task is to evaluate how accurate the generated answer is compared to the ground truth, considering the relevance of the provided context. You will rate the accuracy of the generated answer on a scale of 1 to 10, where:  
  
1 = Completely inaccurate or unrelated to the question  
5 = Partially accurate or relevant, but with significant errors or omissions  
10 = Highly accurate and comprehensive, addressing all aspects of the question correctly  
  
Your response should follow the specified format:  
  
<Format>  
Score: <The accuracy score from 1 to 10 based on the given criteria>  
Supporting Evidence: <Your explanation for the assigned score, including the evidence from the provided information that supports your evaluation>  
</Format>  
  
Please provide your response in the specified format, including the accuracy score and the supporting evidence for your evaluation.''')

**While Employed Chain of Thoughts Prompting Technique:**

accuracy\_prompt\_COT=("""  
You are an evaluator tasked with assessing the accuracy of answers generated by an RAG-based LLM application. You will be provided with the following information:  
  
- Question: The original question asked.  
- Answer: The answer generated by the RAG-based LLM application.  
- Context: The relevant context information used by the RAG-based LLM application.  
- Ground Truth: The correct or "ground truth" answer.  
Your task is to evaluate how accurate the generated answer is compared to the ground truth, considering the relevance of the provided context.  
You will rate the accuracy of the generated answer on a scale of 1 to 10, where:  
1 = Completely inaccurate or unrelated to the question  
5 = Partially accurate or relevant, but with significant errors or omissions  
10 = Highly accurate and comprehensive, addressing all aspects of the question correctly  
Examples:  
Example 1:  
Question: What is the capital city of France?  
Answer: The capital city of France is Paris.  
Context: France is a country located in Western Europe. Its capital and largest city is Paris, a global center for art, fashion, gastronomy, and culture.  
Ground Truth: The capital city of France is Paris.  
  
Rating: 10  
Explanation: The generated answer is highly accurate and correctly identifies Paris as the capital city of France. The provided context is relevant and supports the answer.  
  
Example 2:  
Question: Who wrote the novel "To Kill a Mockingbird"?  
Answer: The novel "To Kill a Mockingbird" was written by Ernest Hemingway.  
Context: "To Kill a Mockingbird" is a classic novel that explores racial injustice and moral courage in a small Southern town.  
Ground Truth: The novel "To Kill a Mockingbird" was written by Harper Lee.  
  
Rating: 1  
Explanation: The generated answer is inaccurate. The novel "To Kill a Mockingbird" was written by Harper Lee, not Ernest Hemingway. The provided context is relevant but does not contain enough information to determine the author.  
  
Now, evaluate the next set of provided information using the same format.  
  
- Please answer with this Format:  
  
   
 <Format>  
Score: <The score 0 to 10 based on the given criteria>  
Supporting Evidence:<Actual Evidence>  
  
</Format>  
  
""")

1. **Evaluation Framework Implementation**

The RAGRate framework measures three crucial metrics: Context Relevance, which evaluates the relevance of the retrieved information to the given context; Faithfulness, which assesses the factual accuracy and truthfulness of the generated responses based on the retrieved information; and Accuracy, which measures the overall correctness of the model's outputs.

I have employed a language model itself as a judge to evaluate these metrics, leveraging its natural language understanding capabilities. The RAGRate framework has been applied to test both Few-Shot+CoT (Chain-of-Thought) prompts, which provide the model with additional reasoning guidance, and Zero-Shot prompts, which rely solely on the model's inherent capabilities. By comparing the performance of multiple open-source models across these prompting strategies, this study provides valuable insights into the strengths and limitations of different RAG approaches when handling text data for reasoning and inference tasks.

The actual implementation of the framework is designed and developed using Python.

Working: The framework will take any dataset that contains the question, answer, context, and ground truth as input and automatically evaluate their different metrics for the data as mentioned above. The evaluation is performed using 3 open sourced models and they are

model\_names = ["mixtral-8x7b-32768", "llama3-70b-8192", "gpt-3.5-turbo"]

**Implementation:**

The RAGRate framework is implemented using Python, leveraging its rich ecosystem of libraries and tools for natural language processing and machine learning.

**Working Principle**

The framework is designed to take any dataset containing questions, answers, contexts, and ground truth as input. It then automatically evaluates the Context Relevance, Faithfulness, and Accuracy metrics for the provided data.

The evaluation process is performed using three open-source language models:

**model\_names = ["mixtral-8x7b-32768", "llama3-70b-8192", "gpt-3.5-turbo"]**

The input dataset is tested against all three models using different prompting techniques and model parameters. In this study, the temperature parameter was set to 0.5 to assess the performance under different model configurations.

**Evaluation Process:**

The framework loads the input dataset and preprocesses the data as necessary.

For each question-answer pair in the dataset, the framework retrieves the relevant context information.

The retrieved context is then provided as input to the language models, along with the question and answer, using different prompting techniques (e.g., Few-Shot+CoT, Zero-Shot).

The models generate their respective responses based on the input.

The RAGRate framework evaluates the generated responses against the ground truth using the Context Relevance, Faithfulness, and Accuracy metrics.

The evaluation results are aggregated and analyzed, providing insights into the performance of each model and prompting technique.

The results of this study, as presented in Section 8, showcase the consistent performance of the Llama3-70b model across different prompting techniques and model configurations.

By systematically evaluating multiple language models and prompting strategies, the RAGRate framework aims to provide a comprehensive understanding of the strengths and limitations of RAG models in handling reasoning and inference tasks over text data.

**Data Set used for testing:**

The dataset used is **Opinion-based QA over financial data**  which contains 4 columns and they are Question, answer, context, Ground Truth. More details can be found here about the dataset used.

[**https://sites.google.com/view/fiqa/**](https://sites.google.com/view/fiqa/)

The input dataset is tested against all three models with different prompting techniques and different model parameters by changing the model parameters. The model parameters that I changed are temperature and output token length which are the same across different models. I changed the temperature parameter to 0.5 in this study and tested against these model parameters.  
  
The **temperature parameter** in a language model API controls the randomness or variability of the generated outputs. It typically ranges from 0 to 1, with higher values increasing the randomness and diversity of the outputs, and lower values favouring more predictable and conservative outputs based on the model's training data. A default temperature value, usually around 0.7 or 1.0, aims to balance coherence and diversity. Adjusting the temperature allows fine-tuning the model's behaviour for specific applications, such as prioritizing creativity (higher temperature) or accuracy (lower temperature). However, it's essential to experiment with different temperature values and evaluate the results to find the optimal setting for your use case.

The results showcase Llama3-70b show great consistency across different models as shown in the Section 7.

**7. Testing against a Custom RAG**

I tested the framework with a publicly available dataset stated above and a private Q&A over some documents related to Form 10-Q of different companies containing the financial results of Q1 2023. So, the total number of questions tested is 40.

Example of a sample answer from RAGRate while using Llama-3-70b. In this example, I have used the CoT prompting technique for evaluation.

**Question, answer, Answer, Context are inputs to RAGRate**

**Question**: What do you know about Microsoft income in q1 2023

**Answer**: In Q1 2023, Microsoft's income saw a decrease of 11% in the More Personal Computing segment. This decline was primarily driven by decreases in Windows revenue, Devices revenue, and Gaming revenue. Windows revenue specifically decreased by 18%, mainly due to a decline in Windows OEM revenue by 28% as a result of continued weakness in the PC market. Additionally, Devices revenue dropped by 24% due to challenges in executing new product launches and ongoing weakness in the PC market. Gaming revenue also saw a decrease of 7%, driven by declines in Xbox content and services as well as Xbox hardware.

**Context:**

This context is retrieved from the RAG application which is extracted from the retriever component of the RAG application. (Retriever component is explained above in the RAG )

|  |
| --- |
| Highlights from the second quarter of fiscal year 2023 compared with the second quarter of fiscal year 2022 included:  Microsoft Cloud revenue increased 22% to $27.1 billion.  Office Commercial products and cloud services revenue increased 7% driven by Office 365 Commercial growth of 11%.  Office Consumer products and cloud services revenue decreased 2% and Microsoft 365 Consumer subscribers increased to  63.2 million.  LinkedIn revenue increased 10%.  Dynamics products and cloud services revenue increased 13% driven by Dynamics 365 growth of 21%.  Server products and cloud services revenue increased 20% driven by Azure and other cloud services growth of 31%.  Windows original equipment manufacturer licensing (Windows OEM) revenue decreased 39%. Highlights from the third quarter of fiscal year 2023 compared with the third quarter of fiscal year 2022 included:  Microsoft Cloud revenue increased 22% to $28.5 billion.  Office Commercial products and cloud services revenue increased 13% driven by Office 365 Commercial growth of 14%.  Office Consumer products and cloud services revenue increased 1% and Microsoft 365 Consumer subscribers increased to   65.4 million.  LinkedIn revenue increased 8%.  Dynamics products and cloud services revenue increased 17% driven by Dynamics 365 growth of 25%.  Server products and cloud services revenue increased 17% driven by Azure and other cloud services growth of 27%.  Windows original equipment manufacturer licensing (Windows OEM) revenue decreased 28%. Three Months Ended December 31, 2022 Compared with Three Months Ended December 31, 2021  Revenue increased $1.0 billion or 2% driven by growth in Intelligent Cloud and Productivity and Business Processes, offset in part by a  decline in More Personal Computing. Intelligent Cloud revenue increased driven by Azure and other cloud services. Productivity and  Business Processes revenue increased driven by Office 365 Commercial. More Personal Computing revenue decreased driven by declines  in Windows, Devices, and Gaming.  Cost of revenue increased $528 million or 3% driven by growth in Microsoft Cloud, offset in part by a reduction in depreciation expense due  to the change in accounting estimate for the useful lives of our server and network equipment. Six Months Ended December 31, 2022 Compared with Six Months Ended December 31, 2021  Revenue increased $5.8 billion or 6% driven by growth in Intelligent Cloud and Productivity and Business Processes, offset in part by a  decline in More Personal Computing. Intelligent Cloud revenue increased driven by Azure and other cloud services. Productivity and  Business Processes revenue increased driven by Office 365 Commercial and LinkedIn. More Personal Computing revenue decreased driven  by declines in Windows, Devices, and Gaming, offset in part by growth in Search and news advertising.  Cost of revenue increased $2.3 billion or 8% driven by growth in Microsoft Cloud, offset in part by the change in accounting estimate. Operating  expenses included a favorable foreign currency impact of 3%.  More Personal Computing  Revenue decreased $3.3 billion 11%.  Windows revenue decreased $2.1 billion or 18% driven by a decrease in Windows OEM. Windows OEM revenue decreased  28% driven by continued PC market weakness. Windows Commercial products and cloud services revenue increased 2%  driven by demand for Microsoft 365, offset in part by a decline in standalone product sales.  Devices revenue decreased $893 million or 24% driven by continued PC market weakness and execution challenges on new  product launches.  Gaming revenue decreased $667 million or 7% driven by declines in Xbox content and services and Xbox hardware. 399 $ 22,247 (8)% $ 41,917 $ 42,485 (1)%       NOTE 13 LEASES  We have operating and finance leases for datacenters, corporate offices, research and development facilities, Microsoft Experience Centers,  and certain equipment. Our leases have remaining lease terms of less than 1 year to 19 years, some of which include options to extend the  leases for up to 5 years, and some of which include options to terminate the leases within 1 year.  The components of lease expense were as follows:    (In millions) Three Months Ended  December 31, Operating expenses increased $554 million or 12% driven by investment in LinkedIn and employee severance expenses.  Revenue, gross margin, and operating income included an unfavorable foreign currency impact of 6%, 8%, and 11%, respectively. Operating  expenses included a favorable foreign currency impact of 2%.  Intelligent Cloud  Revenue increased $3.2 billion or 18%.  Server products and cloud services revenue increased $3.2 billion or 20% driven by Azure and other cloud services. Azure and  other cloud services revenue grew 31% driven by growth in our consumption-based services. Server products revenue  decreased 2%. Three Months Ended March 31, 2023 Compared with Three Months Ended March 31, 2022  Revenue increased $3.5 billion or 7% driven by growth in Intelligent Cloud and Productivity and Business Processes, offset in part by a   decline in More Personal Computing. Intelligent Cloud revenue increased driven by Azure and other cloud services. Productivity and   Business Processes revenue increased driven by Office 365 Commercial. More Personal Computing revenue decreased driven by Windows   and Devices.  Cost of revenue increased $513 million or 3% driven by growth in Microsoft Cloud, offset in part by a reduction in depreciation expense due   to the change in accounting estimate for the useful lives of our server and network equipment. |

**Ground Truth:**

In the publicly available dataset, the ground truth column is already present and in some of the cases, I manually created the ground truth by studying and analysing the document.

|  |
| --- |
| Q1 2023 income for Microsoft is $23 B |

**Answers from RAGRate:**

**Context Relevance Score and Reasoning:**

**4**The context provides detailed information about Microsoft's revenue and growth in different sectors for various quarters, including the second and third quarters of fiscal year 2023. However, it does not provide direct information about Microsoft's income in Q1 2023. The context does provide information about revenue growth and changes in different sectors, but it is not specifically about Q1 2023.   
  
Score: 4

**Accuracy and Reasoning:**

The generated answer provides some accurate information about the decline in Microsoft's income in the More Personal Computing segment, specifically mentioning the decreases in Windows revenue, Devices revenue, and Gaming revenue. However, it does not directly address the question about Microsoft's Q1 2023 income.  
  
The provided context is relevant and contains information about Microsoft's revenue and income in different segments, but it does not explicitly mention the company's Q1 2023 income. The ground truth answer is $23 billion, which is not directly addressed in the generated answer.  
  
Overall, the generated answer is partially accurate and relevant to the topic, but it does not fully address the question, resulting in a score of 5.

**Faithfulness and Reasoning Score:**

List of Claims inferred from context:  
1) Microsoft's income saw a decrease of 11% in the More Personal Computing segment.  
2) The decline was primarily driven by decreases in Windows revenue, Devices revenue, and Gaming revenue.  
3) Windows revenue decreased by 18%, mainly due to a decline in Windows OEM revenue by 28% as a result of continued weakness in the PC market.  
4) Devices revenue dropped by 24% due to challenges in executing new product launches and ongoing weakness in the PC market.  
5) Gaming revenue decreased by 7%, driven by declines in Xbox content and services as well as Xbox hardware.  
  
List of claims in the given answer:  
1) In Q1 2023, Microsoft's income saw a decrease of 11% in the More Personal Computing segment.  
2) This decline was primarily driven by decreases in Windows revenue, Devices revenue, and Gaming revenue.  
3) Windows revenue specifically decreased by 18%, mainly due to a decline in Windows OEM revenue by 28% as a result of continued weakness in the PC market.  
4) Devices revenue dropped by 24% due to challenges in executing new product launches and ongoing weakness in the PC market.  
5) Gaming revenue also saw a decrease of 7%, driven by declines in Xbox content and services as well as Xbox hardware.

I have attached a sample dataset and results from RAGRate for different models below.

Some of the test datasets are acquired from Hugging Face.



**Code can be found here:** **https://github.com/patukuri/RAGRate/tree/master**

**8.ReAct Prompting**

**ReAct** prompting is a framework that allows language models (LLMs) to engage in reasoning and interact with external sources to gather information, leading to more reliable and factual responses. This approach combines the strengths of LLMs with the ability to retrieve and utilize external knowledge, enabling them to handle complex tasks and decision-making scenarios.

In the ReAct prompting approach, the LLM generates reasoning traces, which are step-by-step thought processes or action plans. These reasoning traces allow the model to induce, track, and update its plans, as well as handle exceptions or unexpected situations. Additionally, the **action step in the ReAct framework enables the LLM to interface with and gather information from external sources such as knowledge bases, search engines, or simulated environments.**

By incorporating external information sources, ReAct prompting aims to enhance the reliability and factual accuracy of LLM responses, as the model can draw upon a broader knowledge base beyond its initial training data. This approach has shown promising results, outperforming several state-of-the-art baselines on language and decision-making tasks.

However, testing and evaluating applications built using ReAct prompting can be **challenging for several reasons:**

**Complexity of reasoning traces:** Evaluating the reasoning traces generated by the LLM can be intricate, as it involves assessing the logical coherence, relevance, and completeness of the thought processes. This requires a deep understanding of the task domain and the ability to evaluate the reasoning steps objectively.

**External knowledge integration:** The integration of external knowledge sources adds another layer of complexity to the evaluation process. The quality, relevance, and reliability of the retrieved information can significantly impact the overall performance of the application. Ensuring the consistency and accuracy of the external knowledge sources is crucial but can be a daunting task.

**Dynamic environment interactions:** In cases where the ReAct framework interacts with simulated or real-world environments, the evaluation process becomes even more challenging. These environments can be dynamic and unpredictable, making it difficult to create comprehensive test cases or scenarios that cover all possible situations.

**Human evaluation:** While automated metrics can provide quantitative evaluations, human evaluation is often necessary to assess the quality, coherence, and trustworthiness of the LLM's responses. This can be time-consuming and resource-intensive, particularly for complex tasks or domains.

To address these challenges, a combination of techniques may be employed, such as:

* Developing specialized evaluation metrics that capture the nuances of reasoning, knowledge retrieval, and integration.
* Creating diverse and representative evaluation datasets that cover a wide range of scenarios and edge cases.
* Incorporating human evaluation through crowdsourcing or expert panels.
* Conducting extensive testing in simulated environments or controlled settings before deploying the application in real-world scenarios.
* Continuously monitoring and updating the external knowledge sources to ensure their relevance and accuracy.

Additionally, the authors of the ReAct framework recommend combining ReAct prompting with the chain-of-thought (CoT) approach, which allows the LLM to leverage both its internal knowledge and the external information obtained during reasoning. This combination has shown improved human interpretability and trustworthiness of LLM responses.

Overall, while ReAct prompting offers a promising approach to enhancing the capabilities of LLMs, the complexity of reasoning traces, external knowledge integration, and dynamic environment interactions make testing and evaluating applications built using this framework a significant challenge that requires careful consideration and a multifaceted approach.

1. **Results:**

The results of our evaluation using the RAGRate framework are presented across three key metrics: Accuracy, Faithfulness, and Context Relevance. We compared the performance of three language models (mixtral-8x7b-32768, llama3-70b-8192, and gpt-3.5-turbo) under different prompting techniques: Zero Shot, CoT (Chain-of-Thought), and CoT with Randomness 0.5.

**Accuracy**

The accuracy scores as shown in Fig (2), provide insights into the correctness of the generated responses. Under the Zero Shot prompting condition, the mixtral-8x7b-32768 model achieved the highest average accuracy score of approximately 7.8, followed by llama3-70b-8192 at around 6.8, and gpt-3.5-turbo at approximately 5.2. With the CoT prompting technique, the scores across all three models were relatively similar, ranging from 6.2 to 6.4. When introducing randomness (CoT with Randomness 0.5), the models exhibited slightly lower accuracy scores, with llama3-70b-8192 performing the best at around 6.2.

**Faithfulness**

Fig 3, depicts the faithfulness scores, which measure the factual consistency of the generated responses against the provided context. The CoT with Randomness prompting technique yielded the highest faithfulness scores, with mixtral-8x7b-32768 leading at approximately 0.74, followed by llama3-70b-8192 at around 0.66, and gpt-3.5-turbo at approximately 0.56. Under the Zero Shot condition, the faithfulness scores were relatively lower, ranging from 0.52 to 0.56 across the three models. The CoT prompting technique resulted in slightly higher faithfulness scores compared to Zero Shot, with llama3-70b-8192 achieving the highest score of around 0.66.

**Context Relevance**

As shown in Fig 1, the context relevance scores evaluate how relevant the retrieved context is to the given question. The CoT with Randomness 0.5 prompting technique demonstrated the highest context relevance scores, with mixtral-8x7b-32768 and llama3-70b-8192 models scoring around 5.6, while gpt-3.5-turbo scored slightly lower at approximately 4.8. Under the Zero Shot condition, the context relevance scores were relatively lower, ranging from 3.2 to 3.6 across the models. The CoT prompting technique yielded context relevance scores of around 5.0 for both mixtral-8x7b-32768 and llama3-70b-8192 models.

Overall, the results suggest that the prompting technique and the choice of language model can significantly impact the performance across the evaluated metrics. **The llama3-70b-8192 model consistently demonstrated strong performance, particularly in terms of faithfulness and context relevance,** while the mixtral-8x7b-32768 model excelled in accuracy and context relevance under certain prompting conditions. The CoT with Randomness 0.5 prompting technique generally yielded higher faithfulness and context relevance scores, although it may have compromised accuracy to some extent.

These findings highlight the importance of carefully selecting the appropriate language model and prompting strategy based on the specific requirements and trade-offs between accuracy, faithfulness, and context relevance for the target application.

A graph of different sizes and colors

Description automatically generated with medium confidence

Fig 1

A graph of blue bars

Description automatically generated with medium confidence

Fig 2

A screenshot of a graph

Description automatically generated

Fig 3

## **10.Future Work and Conclusion:**

1. **Evaluate ReAct+ RAG Prompting**:
   * Implement ReAct+ RAG prompting and assess result consistency across models and randomness values.
2. **Compare ReAcT and ReAct+RAG**:
   * Conduct comparative studies between ReAcT and ReAct+RAG across different domains.
   * Evaluate strengths, weaknesses, and performance in terms of accuracy, relevance, and coherence.
3. **Semantic Search and Cosine Similarity**:
   * Integrate semantic search and cosine similarity for knowledge retrieval.
   * Explore similarity metrics and ranking algorithms for optimal retrieval.
4. **Scalability and Optimization**:
   * Investigate techniques for scaling to larger datasets and complex prompts.
   * Explore distributed computing, caching, and indexing strategies for performance optimization.

**11.References:**

* Llama-index
* Langchain
* Groq
* Chat GPT
* AWS Bedrock
* IBM Watson Studio
* <https://arxiv.org/abs/2210.03629>
* Truelens
* ( RAGAS, Truelens, Databrickslabs, FastChat)