Fall 2023 Capstone Project
Final Report

Reinforcement Learning for Generative AI LLMs

Dec 16, 2023

Group 13

Yi Lu (yl5118)
Xiaolin Sima (xs2483)
Yanni Chen (yc4179)
Junyuan Huang (jh4608)
Michelle Sun (ms6514)

Table of Contents

1. Introduction

This capstone project is a collaboration between Columbia University Data Science Institute and Accenture. The motivation of this Capstone Project is to unlock significant insights into the Oil and Gas Industry, emphasizing the critical role of data-driven decision-making through the use of AI such as Large Language Models (LLMs).

One problem that consultants are facing is that company documentation such as annual reports and sustainability reports can range from 50 to 500 pages and contain large amounts of both numeric and textual information. Manually reading through the whole report in this case can be time-consuming. With the help of LLMs, the process of understanding how the company performs can be efficient.

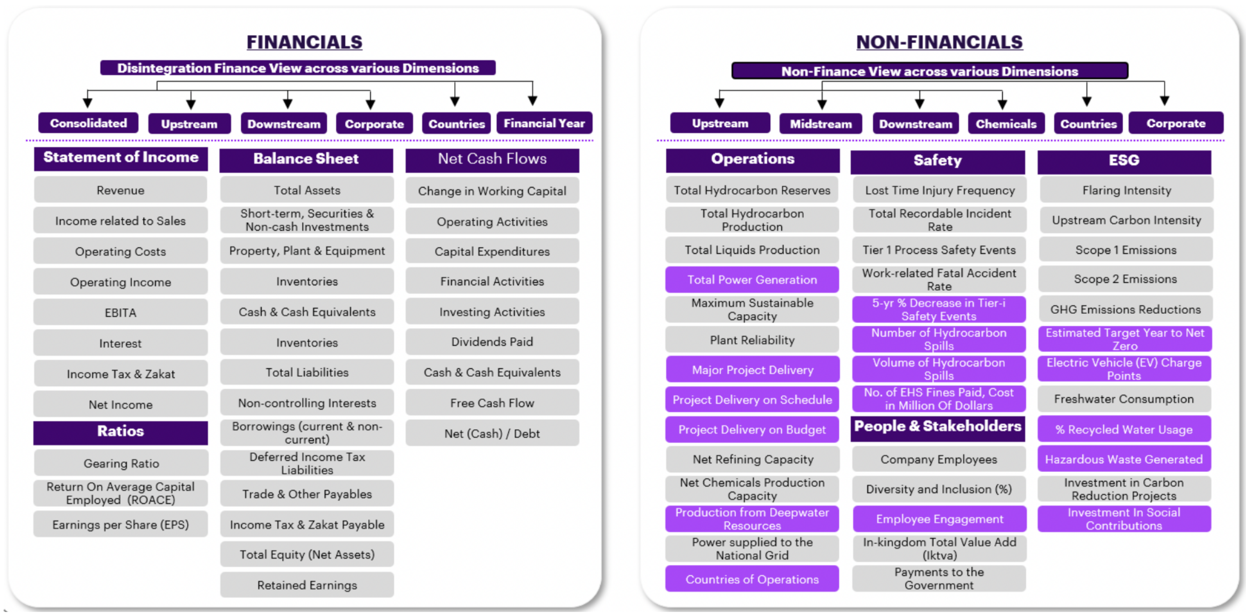
LLMs have gained a lot of attention in recent years for their exceptional capabilities in doing natural language understanding and document analysis. Researchers have explored the application of LLMs in different fields, showcasing their power in both generating text and extracting meaningful information from documentation (Kasneci et al. 2023). Specifically, pre-trained LLMs can be further fine-tuned on a specific task to improve their performance accordingly (Min et al., 2021). With that, we are positive about utilizing and fine-tuning LLMs to help improve the analyzing process.

Our overall approach to tackling this problem is to develop a system that contains two primary models with the use of LLMs: a prompt optimizer that will help the user enhance the quality of their instructions to the report analyzer, and a report analyzer that will read the documentation, and perform tasks such as generating a summary of the report, or answering specific questions that the user cares about, such as KPIs of the company. We will further fine-tune both models to enhance their abilities.

2. Research

2.1. KPIs

Below are the illustrations of key KPIs that our LLM model is expected to identify. We will break the KPIs into two big categories: Financial KPIs and Non-Financial KPIs. As for the specific field of the oil industry, we are also interested in upstream, midstream, and downstream productions. For the Financial KPIs, we will mainly extract them from the three financial statements: the Income Statement, the Balance Sheet, and the Statement of Cash Flows. Also, we are interested in the non-financial metrics that break into three fields: Operations, Safety, and ESG (Environmental Social, and Governance) which we can mainly extract from text and small tables.



2.2. Vector Databases

As we are dealing with PDF data, embedding is needed to represent data in a lower-dimensional vector space, making it easier to work with and analyze using machine learning algorithms. After we have our embedded chunks from the raw PDF data, we need to index (store) them somewhere so that we can retrieve them quickly for inference, where the vector database is needed. In this section, we investigate different vector databases, and compare the vector libraries and vector databases, to find an optimal solution to easily store and retrieve the data.

2.2.1. Comparison between Vector Libraries and Vector Database

<u>Vector Libraries</u>

Vector libraries store vector embeddings in in-memory indexes to perform similarity searches. Most vector libraries share the following characteristics:

1. Store vectors only
2. Index data is immutable
3. Query during import limitation

<u>Vector Databases</u>

One of the core features that sets vector databases apart from libraries is the ability to store and update your data. Vector databases have full CRUD (create, read, update, and delete) support that solves the limitations of a vector library. Additionally, databases are more focused on enterprise-level production deployments.

2.2.2. Pros and Cons of Different Vector Database/Libraries

Name	Pros	Cons
ChromaDB	- Easy application and integration - Hands-on experience - Embedding model can be customized	- May struggle with a large-scale dataset - Limited customization for data modeling and querying (may not be the main concern)
Milvus	- Optimized for similarity search with advanced indexing techniques to speed up search operations - Allow large datasets and high query loads - Can leverage GPU resources, leading to significant speed improvements - Specialized in similarity search and handle embedding vectors converted from unstructured data (built to pair with FAISS)	- Setting up and configuring Milvus can be complex - May need to fine-tune Milvus, select appropriate indexing methods, and configure hardware resources like GPUs. This optimization process can be challenging and time-consuming. - Data is often assumed to be static once fed into existing systems, complicating processing for dynamic data

pgvector	- Easy to integrate with PostgreSQL	- Most cloud offerings of PostgreSQL have not yet integrated pgvector - Not that convenient if we are not going to use PostgreSQL
Qdrant	- Fast and reliable even under high load - Supports metadata filtering like ChromaDB, and integrates into technologies like Cohere (embeddings), LangChain, and LlamaIndex	- Open-source vector database written in Rust - Not entirely free
Pinecone	- Offers blazing-fast search capabilities, allowing users to retrieve similar vectors in real-time, making it well-suited for content-based searching - Excellent choice to deal with vast amount of data due to its architecture design - Relatively easy to apply with automatically indexes vectors	- Might lack some advanced querying capabilities that certain projects require - Not open-source but has a free version that search through roughly a million vectors in around 100ms, or through 100K vectors in around 20ms
FAISS	- Allows developers to quickly search for embeddings of multimedia documents that are similar to each other - Contains supporting code for evaluation and parameter tuning	- Not a database, but a library

According to our research, we think ChromaDB and FAISS can be two possible options that are worth trying and comparing when we store and retrieve data with designed queries. They are both easy to implement and can perform essential functions we need.

2.3. Open-source Large Language Models

Name	Introduction	Pros	Cons
T5	The T5 series of models open-sourced by Google is	- A great model to fine-tune at a	- Requires a large amount of

	available in various sizes of parameters. It is a Sequence to Sequence model that follows an Encoder and Decoder architecture.	relatively low cost - Wide variety of model sizes - Well suited for translation and summarization	computational power and memory - May generate unreliable results with new inputs - Long training time
Falcon	Falcon is a causal decoder-only model built by TII and trained on tokens of RefinedWeb enhanced with curated corpora to use on summarization, text generation, chatbot, etc.	- Strong conversational capabilities and performance optimization, well-suited for conversational experiences	- Falcon contains fewer parameters than GPT so less complexity and capacity to capture and generate human-like text
Llama	Developed by Meta, Llama is an auto-regressive language model that uses an optimized transformer architecture. The tuned versions use supervised fine-tuning and RLHF to align with human preference for helpfulness and safety.	- LLaMA 2 can generate high-quality texts in a matter of seconds. - It uses less computational resources than other LLMs of similar size and complexity	- Fewer parameters, data, context length, and modalities than GPT giving Llama an edge in terms of accuracy, complexity, diversity, and generality of its outputs.
Mistral	The Mistral LLM developed by Mistral AI is a 7-billion-parameter language model engineered for superior performance and efficiency.	- Mistral 7B is better than Llama 2 13B on all benchmarks, has natural coding abilities, and 8k sequence length	- Mistral 7B is a pre-trained base model and therefore does not have any moderation mechanisms
Vicuna	Vicuna is a chat assistant trained by fine-tuning Llama 2 on user-shared conversations collected from ShareGPT by The large model systems organization (LMSYS).	- It achieves more than 90% quality of GhatGPT while outperforming other models like LLaMA in more than 90% of cases according to a non-scientific evaluation	- Vicuna is not good at tasks involving reasoning or mathematics and may have limitations in ensuring the factual accuracy of its outputs

3. Data Preparation

3.1. Source

We gathered our raw pdf data by collecting recent 5-year annual reports, most recent sustainability reports, and 2-3 years quarterly reports directly from the official websites of ten select companies[1]. This data collection process allowed us to access the most up-to-date and accurate data, providing a comprehensive overview of each company's financial performance, strategic initiatives, and commitment to sustainable practices. Through this collection process, we ensured the authenticity and reliability of the information, enabling us to perform in-depth analyses and benefit from further informed decisions in our ongoing business and investment strategies.

3.2. Load and Split

Company reports often contain rich and structured content, including text, images, tables, and graphs. The ability to load and parse PDFs enables the model to extract valuable insights from different data structures. In addition to the loading process, data splitting is equivalently essential for breaking down large documents into smaller, more manageable sections, also called chunks. The splitted chunks are then used as a comprehensive knowledge base for LLM. Different configurations of chunk size, chunk overlap, and separators will form a distinct representation of the original PDF. A successful segmentation makes it easier for the vectorstore to retrieve relevant information quickly and efficiently, improving the overall performance and response time of the system.

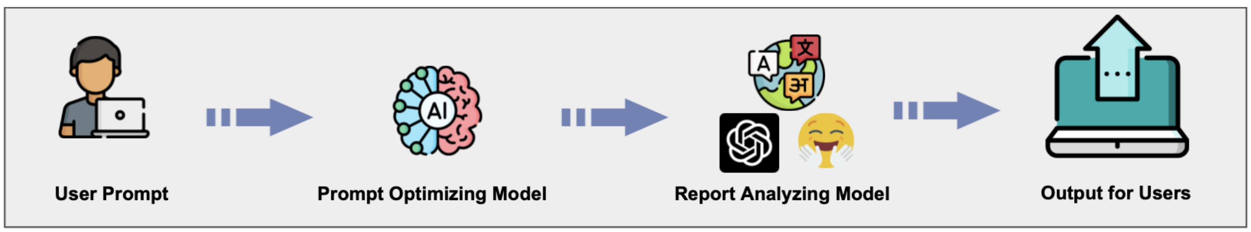
Different LLMs have different input size constraints, which results in different choices on chunk size and number of chunks retrieved per question. Upon testing 21 combinations from 7 PDF loaders, and 3 splitters, we successfully found an efficient setting of loader and splitter. Detailed analysis of loaders and splitters is included in the Q&A section.

---

[1] Shell plc, BP PLC , Saudi Aramco, Chevron, TotalEnergies, Valero Energy, Marathon Petroleum Corporation, Sinopec and PetroChina

4. Modeling and Tuning

The culmination of our project comprises two primary models: the prompt optimization model and the report analysis model. In developing the prompt optimization model, we employ the Black-Box Prompt Optimization method to train a large language model. The primary objective here is to enhance the quality of prompts provided to the report analysis model. Subsequently, the report analysis model processes the report based on the provided prompt, with a primary focus on summarizing the report and extracting key performance indicators of the company. The overall workflow is shown below.

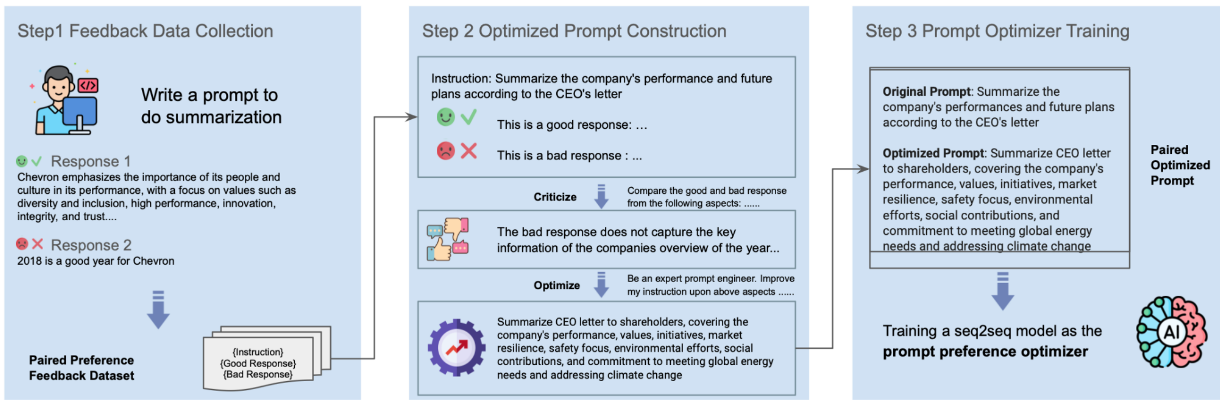


4.1. Prompt Optimizing Model

To ensure an effective analysis of the report, it is crucial to provide a clear and coherent prompt to guide the model in its analysis. This is especially important for the summarization task we expect the report analyzer model to undertake. Given the potential length and diverse content of the report we are dealing with, a well-structured prompt that aligns with the initial focus on key performance indicators (KPIs) and company financial performance is essential to provide precise instructions for the analysis task. In this regard, we intend to develop a prompt optimization model that will enhance the informativeness and effectiveness of the instructions provided to the report analyzing model.

4.1.1. Black-Box Prompt Optimization Method

In order to train the model in a more effective way, we have employed the Black-Box Optimization (BPO) Method. The BPO algorithm operates by enabling the model to optimize the prompt itself by learning what is a good prompt that can produce ideal results. The BPO method we employed is a three-step approach to optimize AI-generated text prompts, particularly for summarization tasks.



Alignment Stage of Black-Box Prompt Optimization

The first step is Feedback Data Collection, where prompts are written to generate summaries. Two types of responses are collected: good responses that align with desired outcomes (like emphasizing a company's values, diversity, performance, innovation, and trust) and bad responses that are irrelevant or inaccurate (like stating a particular year was good without context). The responses are then used to create a feedback dataset that pairs good and bad responses, which serves as training data for prompt optimization.

The second step is Optimized Prompt Construction. In this step, a prompt is refined based on previous feedback data. The process involves critically analyzing both good and bad responses to understand what makes a response high-quality, such as capturing key information about a company's annual overview. An expert prompt engineer then uses this analysis to enhance the instruction of the prompt, ensuring it guides the AI to generate more accurate and informative summaries.

The last step is Prompt Optimizer Training. Here, a sequence-to-sequence (seq2seq) model is trained using the paired optimized prompts from Step 2. The training process involves fine-tuning the model to understand the preferences indicated by the paired prompts, to produce responses that are more aligned with the optimized prompt. The fine-tuning process will be discussed more in 4.1.2. fine-tuning section of the model. The optimized prompt is more detailed, including the company's performance, values, initiatives, market resilience, safety focus, environmental efforts, social contributions, and commitment to meeting global energy needs and addressing climate change.

Overall, this BPO process aims to improve the model's ability to generate summaries that are both relevant and valuable to stakeholders by leveraging feedback data and expert refinement of prompts.

4.1.2. Fine-Tuning of the Model

To enhance the precision of our text generation task, we initiate the fine-tuning process by leveraging a specialized version of the Llama Model, known as the BPO model[2], which has been pre-tuned and available on the Hugging Face platform. Our approach to further refine the model's capabilities involves the introduction of a bespoke dataset, meticulously compiled to include the initial prompt, its enhanced version, and the associated context in the training part of our target reports, ensuring a tailored training experience.

Incorporating the innovative LoRA technique, we target selective portions of the model for adjustment, an efficient strategy that maximizes the fine-tuning impact while minimizing computational demands. This judicious application of LoRA to the BPO model imparts nuanced improvements without the need for exhaustive retraining of the entire neural network structure.

Once the LoRA configuration is integrated, the fine-tuning procedure commences. Through this process, the model assimilates the intricate distinctions between the original and optimized prompts within varied contextual frameworks, thereby expanding its generative acumen, which we will discuss more in the general results section.

---

[2] https://huggingface.co/THUDM/BPO

4.1.3. Evaluation Metrics

We employ ROUGE and BLEU and compare the optimized prompt generated by the prompt optimizing model and the intended good prompt.

For the original BPO model, the average ROUGE score on the testing dataset was approximately 0.251, indicating that there was a certain level of correspondence between the optimized prompt by the model and the reference good prompt, but with substantial room for improvement. After fine-tuning, the new version of the model achieved a higher average ROUGE score of about 0.317. This suggests a more significant overlap with the reference good prompt indicating that the fine-tuning process enhanced the model's ability to replicate the essential points of the reference texts.

Similarly, the original model's average BLEU score was around 0.035, which is relatively low, reflecting that the original model optimized prompt had limited alignment with the reference texts. After fine-tuning, the Tuned BPO Model exhibited an improved average BLEU score of approximately 0.049. While still under the lower end of the spectrum, this increase denotes better precision in terms of the presence of correct n-grams in the optimized prompt in comparison to the references, hence a step forward in the model's translation or summarization capabilities.

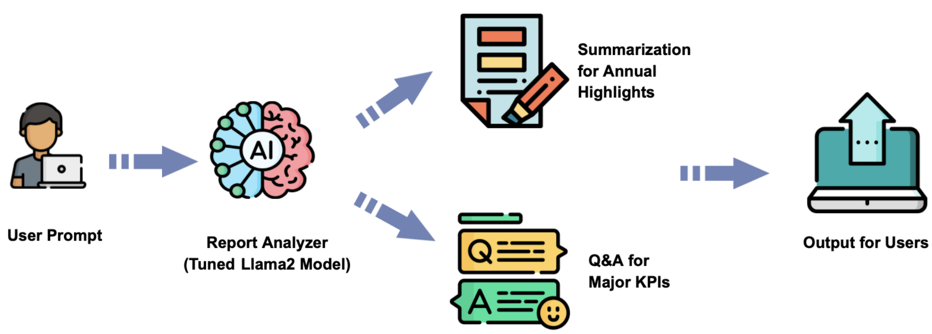
In conclusion, both the ROUGE and BLEU scores indicate that the fine-tuning process has made a positive impact on the model's performance in optimizing prompt that is more in line with the intended focus of the summary.

4.1.4. General Results of the Prompt Optimizing Model

The implementation of fine-tuning on the prompt optimization model has yielded demonstrably beneficial outcomes, as evidenced by the evaluation metrics. By refining the model with a custom dataset, which included a blend of original and optimized prompts in conjunction with their contextual narratives, we have significantly enhanced the model's proficiency in generating an optimized prompt for further analyzing reports and generating summaries.

4.2. Report Analyzing Model

We aim to comprehensively analyze a targeted company's report, focusing on two key aspects: summarizing annual highlights and extracting major KPIs. To accomplish this, we fine-tuned the Llama2 model using a custom dataset that includes informative summarizations and accurate KPI extractions. Specifically, the model adeptly identifies the CEO report—a key source of annual highlights—and produces concise and insightful summarizations. In terms of Q&A, the model can answer the questions provided by users, focusing on KPI.



Report Analyzer Workflow

4.2.1. Similarity Search Method

To efficiently analyze the vast amount of text data extracted from the annual reports of the oil companies, we implemented a sophisticated similarity search mechanism leveraging Facebook AI Similarity Search (FAISS). This step was crucial in identifying and retrieving the most relevant chunks of text for further analysis by our model. The FAISS similarity search mechanism is to identify chunks of text that closely match specific queries. This involved creating embeddings for the text data and organizing them to facilitate fast and accurate similarity searches. The utilization of FAISS in our workflow significantly improved the precision of text retrieval, ensuring that the Language Model received relevant and meaningful chunks for analysis and generated a more targeted and accurate analysis.

4.2.2. Training and Testing Dataset Generation

4.2.2.1. Summarization Task Dataset

Our refined dataset for the Summarization task is meticulously structured into three distinct segments: the question, the answer, and the context. The question segment is composed of enhanced prompts, which are the refined outputs from our fine-tuned prompt optimizing model. These are carefully crafted to encapsulate the specific summarization objectives and guide the model's focus. The answer segment contains the target summaries, which serve as the gold standard for the information that our model aims to condense. Lastly, the context is derived from a curated collection of CEO reports, which have been selected and located in the data preparation by similarity search to act as the primary source for extracting annual highlights. This tripartite arrangement ensures a comprehensive framework for training our model, facilitating a robust understanding and generation of concise, relevant summaries that accurately reflect the essence of the source material.

4.2.2.2. Q&A Task Dataset

One of the distinctive advantages of our project lies in the fine-tuning training dataset, where we depart from conventional approaches that often rely on open-source datasets. Instead, we crafted a custom dataset tailored to the intricacies of the oil and gas industry. This dataset, comprising Question-Answer pairs, serves as the foundation for fine-tuning our Language Model. The key innovation in our methodology is the utilization of Knowledge Distillation, where we leverage the capabilities of GPT3.5-turbo to impart a refined understanding of question-answering dynamics to our model.

We categorized the questions in our custom dataset into three distinct types, each serving a specific purpose in enhancing the adaptability and proficiency of our LLM:

- Data Query Questions (50%):
  Examples: "What is the net profit of a certain company in 2021?"
  Purpose: Focused on extracting specific data points from the annual reports.

- Data Statistical Analysis Questions (30%):
  Examples: "What is the growth rate of a company’s greenhouse gas emissions in 2021?"
  Purpose: Geared towards probing statistical insights and trends within the data.
- Open Questions (20%):
  Examples: "Tell us about a company’s business situation in 2022?"
  Purpose: Encourages a more narrative and comprehensive response, fostering contextual understanding.

The incorporation of Knowledge Distillation in our dataset generation process empowered our LLM to not only answer questions accurately but also to respond with a stylistic finesse akin to GPT3.5-turbo. This approach sets our project apart by delivering a model that not only understands the intricacies of the oil and gas industry but also responds in a manner reflective of state-of-the-art language models.

4.2.3. Fine-Tuning of Report Analyzing Models

To optimize the effectiveness of our question-answer pipeline, we leverage a combination of our proprietary dataset and a quantized llama model. This strategic pairing enhances the model's performance by tailoring it to the intricacies of the oil and gas industry. The utilization of a proprietary dataset ensures that the model is finely tuned to handle specific queries within this domain, while the incorporation of a quantized llama model introduces a resource-efficient approach to computation.

In the technical implementation, we load the quantized llama model with 4-bit precision, a key decision aimed at maximizing efficiency. This not only minimizes the computational burden but also conserves random-access memory (RAM). The adoption of a 4-bit qlora approach proves particularly advantageous for our pipeline, allowing us to strike a balance between model performance and resource usage. This precision choice facilitates faster processing times and optimal memory utilization, contributing to an overall streamlined question-answer pipeline.

The adaptability of our strategy is highlighted by the ability to fine-tune a substantial model like llama within a simplified Colab notebook environment. Despite the potential limitations of this platform, our qlora approach demonstrates its versatility, showcasing how efficient and effective fine-tuning can be achieved in less resource-intensive settings. This combination of a domain-specific dataset, a quantized model, and an accessible platform like Colab positions our approach as both sophisticated and practical for addressing the unique challenges of the oil and gas industry.

4.2.4. Reinforcement Learning Initial Approach

The implementation of Reinforcement Learning with Human Feedback (RLHF) played a pivotal role in refining the performance of our fine-tuned Language Model. RLHF involves the manual labeling of data, training a reward model, and subsequently utilizing Proximal Policy Optimization (PPO) to further enhance the capabilities of our model.

- Manual Labeling Process: To generate labeled data for RLHF, we manually ranked the answers produced by our fine-tuned LLM. The ranking process involved assessing answers based on multiple criteria such as accuracy, readability, and other relevant factors.
- Training the Reward Model: The labeled data served as the training set for our reward model. We utilized GPT2 as our based reward model. Through a supervised learning approach, the model learned to assign rewards to different responses based on the human-generated rankings.
- Proximal Policy Optimization (PPO): We conducted prompt completion experiments, ranked the responses using an average criterion, and leveraged backpropagation to assess and fine-tune our instructive Language Model. Our reference model is the fine-tuned LLama, with frozen weights serving as a benchmark for our aligned model. To address deviations, we incorporated a KL divergence penalty into the reward system, ensuring that the model remains close to the reference model even during hallucinations. The Parameter-Efficient Fine-Tuning (PEFT) adapter was employed to progressively align the PPO model

through multiple rollouts, enhancing its responsiveness and aligning it more closely with our refined LLama.

The RLHF process significantly contributed to the enhancement of our LLM's performance. By incorporating human-generated rankings into the training process, we fine-tuned the model to produce responses that not only align with factual accuracy but also with the correct values, more helpfulness, truthfulness, and harmlessness.

4.2.5. Evaluation Metrics

In the evaluation process of the report analyzing model, we adhere to methodologies akin to those applied to the BPO model. Two crucial metrics, the BLEU score and Rouge score, serve as indispensable criteria for assessment. Notably, the BLEU score registers a positive shift, rising from 0.215 to 0.262, indicative of improved performance. Additionally, the Rouge score witnesses advancement, climbing from 0.451 to 0.461, further affirming the model's effectiveness.

4.2.6. General Results of Report Analyzing Models

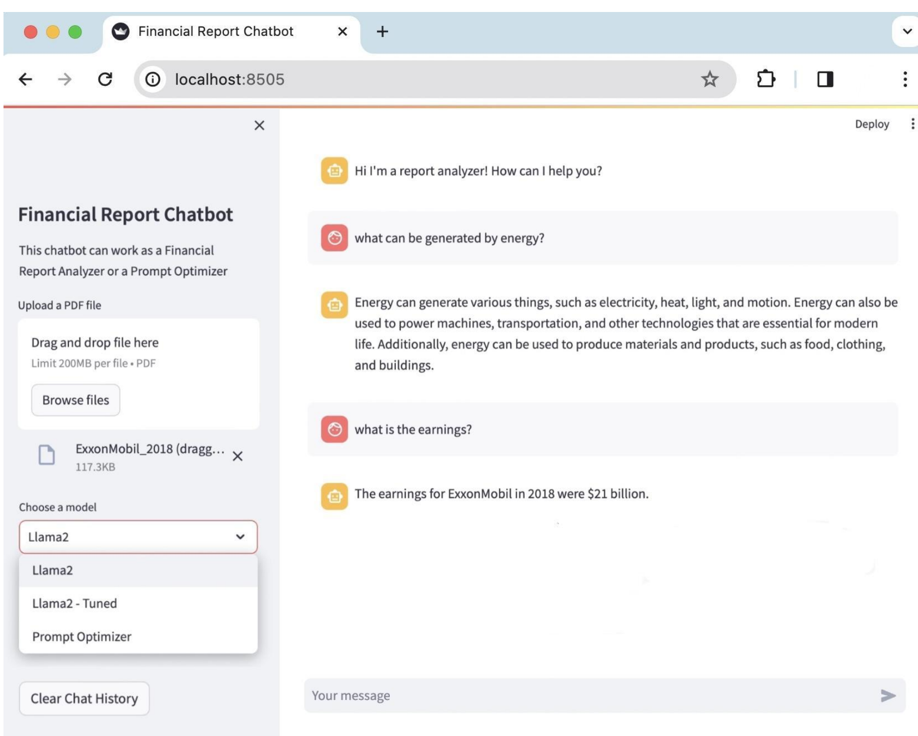
Here we outline the modeling and tuning strategies for two key models: the prompt optimization model and the report analyzing model. The prompt optimization model utilizes the Black-Box Prompt Optimization (BPO) method in a three-step process. Feedback Data Collection gathers good and bad responses to create a training dataset, Optimized Prompt Construction refines prompts based on feedback, and Prompt Optimizer Training fine-tunes a sequence-to-sequence model. This approach enhances the model's ability to generate valuable summaries by leveraging expert-refined prompts and feedback data, as reflected in improved ROUGE and BLEU scores.

The report analyzing model focuses on summarizing annual highlights and extracting key performance indicators. Fine-tuning the Llama2 model involves a custom dataset and a similarity search method using Facebook AI Similarity Search (FAISS). The dataset for summarization tasks is carefully structured, incorporating refined prompts and context from CEO reports. For the Q&A task, a custom dataset tailored to the oil and gas industry is crafted, using Knowledge Distillation to refine question-answering dynamics. The inclusion of a quantized llama model with 4-bit precision optimizes the question-answer pipeline, ensuring resource efficiency.

Reinforcement Learning with Human Feedback (RLHF) plays a pivotal role in refining the report analyzing model. Manual labeling, reward model training, and Proximal Policy Optimization (PPO) contribute to improved precision and summarization capabilities. Evaluation metrics, including BLEU and Rouge scores, validate the positive impact of these modeling and tuning efforts on the model's performance, emphasizing their tailored adaptability to the oil and gas industry's complexities.

5. UI interface design

As a final deliverable, we deploy our Prompt Optimizer and Report Analyzer into a web app through Streamlit. The Streamlit web application provides an intuitive and user-friendly interface for leveraging the capabilities of our Prompt Optimizer and Report Analyzer.



Financial Report Chatbot Web App interface through Streamlit

The layout is cleanly divided into two sections, catering to different aspects of user interaction. On the left-hand side, the sidebar serves as the operational dashboard where users can seamlessly upload their financial reports by dragging and dropping PDF files, adhering to a size limit of 200MB per file. Additionally, this sidebar allows users to select their preferred analytical model from a dropdown menu, offering options of original 'Llama2', 'Llama2 - Tuned', and 'Prompt Optimizer' to generate a powerful prompt for summarization tasks.

The right-hand section of the interface mimics a conversational chatbot, where users can type in their queries. The chatbot, designed to analyze financial reports, can answer questions posed by the users, as demonstrated by the displayed messages regarding energy generation and company earnings. The dialogue box is straightforward, encouraging users to engage in a text-based interaction with the system to obtain insights or summaries from the uploaded financial documents.

6. Discussion

6.1. Summary

Our project leverages LLMs to streamline the analysis of voluminous Oil and Gas industry reports. This initiative addresses the inefficiency of manual report analysis by employing LLMs to extract crucial information more efficiently, enhancing data-driven decision-making.

Our group crafted a system comprising a Prompt Optimizer and a Report Analyzer. The former refines user queries, specifically making the summarization prompt focus on the company's financial performance, while the latter produces summaries and identifies KPIs asked by the user. Both models are fine-tuned to improve accuracy. Performance after fine-tuning is confirmed by elevated ROUGE and BLEU scores, signifying better prompt generation and report analysis. Our final milestone is the system's integration into a user-friendly Streamlit web app, featuring a dual interface for uploading reports and interactive querying via a chatbot.

Overall, this project showcases how LLMs can significantly transform sector-specific analytical tasks, opening the door for further progress and ethical considerations in the realm of AI applications. The thorough research and model development outlined in this report underscore the potential of AI to bring innovation to industry analysis.

6.2. Take Home Messages

With the right approach and tools, AI can be a powerful ally in distilling vast amounts of information into strategic knowledge and business value.

6.3. Future Work

For future work, utilizing a larger dataset for fine-tuning the model and designing a more comprehensive rewarding system for reinforcement learning during the training process can be considered.

6.4. Ethical Considerations

Ethical considerations for this project encompass the responsible handling of sensitive financial data, and ensuring that all analyses uphold the strictest data privacy standards. As AI models have the potential to influence business decisions, it's imperative to consider the accuracy and fairness of AI-generated summaries and KPI identification.

References

Cheng, Jiale, et al. Black-Box Prompt Optimization: Aligning Large Language Models without Model Training. *arXiv:2311.04155, arXiv,* 7 Nov. 2023. arXiv.org, http://arxiv.org/abs/2311.04155.

Kasneci, Enkelejda, et al. “ChatGPT for Good? On Opportunities and Challenges of Large Language Models for Education.” *Learning and Individual Differences*, vol. 103, Apr. 2023, p. 102274. https://doi.org/10.1016/j.lindif.2023.102274.

Min, Bonan, et al. “Recent Advances in Natural Language Processing via Large Pre-Trained Language Models: A Survey.” *arXiv (Cornell University)*, Nov. 2021, https://doi.org/10.48550/arxiv.2111.01243.

Nantasenamat, Chanin. “How to Build a Llama 2 Chatbot.” *Streamlit*, 31 Aug. 2023, blog.streamlit.io/how-to-build-a-llama-2-chatbot.