**P1 Report**

### **DS5500: Data Science Capstone**

Prof. Kylie A. Bemis

### **1. Title:** FinAdvisor - Financial Advisory Tool for contextually-aware personal finance assistance.

### **2. Authors:**

* Laasya Anantha Prasad
* Krishna Venkatesh
* Rakshak Kunchum

### **3. Summary:**

This project identifies and addresses crucial shortcomings in traditional personal finance management tools, which often fail to meet the complex and rapidly evolving demands of today’s financial landscape. Traditional methods are hampered by static knowledge, requiring scheduling and being limited to business hours, which often results in delayed, inaccurate, or overly generalized advice. Such constraints not only impede efficient decision-making but also limit access to timely financial guidance. The reliance on human expertise, often lacking real-time updates and adaptability, further complicates the effectiveness of traditional financial advisory services. As financial planning grows more complex, the need for more dynamic and responsive tools becomes evident to better serve individuals seeking immediate and personalized financial advice.

### **Limitations of Traditional Methods for Personal Finance**

Traditional methods of managing personal finances often present challenges that can hinder effective financial planning and decision-making. Below are some of the primary limitations associated with traditional approaches:

#### **1. Inconvenience**

Traditional financial advisory often requires clients to schedule appointments and potentially travel to meet with advisors, making the process time-consuming and inefficient. This setup can be especially burdensome for individuals with busy schedules or those located far from financial institutions, resulting in delays and reduced flexibility.

#### **2. Limited Accessibility**

Access to traditional financial services is often restricted to regular business hours, which can be unsuitable for urgent financial needs or for individuals who are unable to attend during standard working hours.

This limitation can result in missed opportunities or delayed decision-making, as clients are unable to access critical financial insights and support when they need them most.

#### **3. Static Knowledge**

Traditional methods rely heavily on the expertise of advisors, which may lack the dynamism needed in today’s fast-paced financial environment. Often, the information provided is not updated in real-time, making it challenging to respond to rapid market changes or access the latest financial insights. This dependency on static knowledge can impact the effectiveness of financial advice and limit clients’ ability to make timely decisions.

In response to these challenges, our project leverages the latest advancements in Artificial Intelligence (AI), specifically through the deployment of advanced Large Language Models (LLMs), to develop an AI-driven financial advisory tool. This innovative tool capitalizes on real-time data and global accessibility, offering tailored financial insights and support across various personal finance areas such as budgeting, investment analysis, and risk assessment. By integrating and fine-tuning cutting-edge LLMs with a meticulously curated and cleaned combination of the FinTalk-19k and Financial Alpaca datasets, our solution ensures that the financial advice provided is both current and highly relevant to individual user needs. This integration significantly enhances the quality and responsiveness of financial planning tools, making sophisticated financial advice more accessible and empowering individuals with the tools to make smarter financial decisions efficiently and effectively.

### **Advantages of Large Language Models in Modern Personal Finance Solutions**

The integration of Large Language Models (LLMs) in personal finance solutions presents several significant advantages that address the inherent limitations of traditional financial advisory methods. Key benefits of LLM-based tools include the following:

#### **1. Universal Accessibility**

LLM-powered financial tools provide continuous access to guidance, available at any time and from any location. This accessibility eliminates the need for in-person consultations, offering flexibility for users with busy schedules or those residing in remote areas.

#### **2. Real-Time Data Insights**

These advanced tools leverage real-time market data to deliver up-to-date financial insights. This capability enables users to respond swiftly to market changes, making well-informed financial decisions without delay.

#### **3. Enhanced Efficiency, Consistency, and Reliability**

Large Language Models provide efficient, consistent, and reliable financial support, surpassing the speed and accuracy of traditional methods. This reliability improves the overall user experience by ensuring that financial queries are addressed quickly and precisely, enhancing the decision-making process.

### **4. Methods and Technical Implementation**

#### **a. Data Collection**

Two primary datasets are being utilized to train and evaluate the model for personal finance insights. These datasets provide extensive data points sourced from various financial discussions, combining both curated and generated data elements to enrich the training quality.

#### **1. Financial Alpaca Dataset**

#### The **Financial Alpaca Dataset** was sourced from Hugging Face and represents a comprehensive blend of the Stanford Alpaca dataset, the Financial Question and Answer (FIQA) dataset, and additional data generated through GPT-3.5. This unique combination of curated real-world financial data and synthetic insights creates a robust training set capable of capturing nuanced financial knowledge and advice. With approximately 68,000 entries, the dataset covers an extensive range of financial topics and questions, including investment advice, market trends, economic outlooks, savings strategies, and personal finance management tips.

#### The inclusion of synthetic data generated by GPT-3.5 enriches the dataset by introducing responses to hypothetical scenarios and diverse financial questions that may not be as prevalent in typical datasets. This synthetic component helps bridge informational gaps, allowing the model to learn from a broader set of financial scenarios, including rare or complex cases. As a result, the Financial Alpaca dataset offers a diverse range of perspectives and examples that make it especially suited for training models that need to generate well-rounded, reliable advice on various financial matters.

#### The dataset’s structure includes not only straightforward financial queries but also multi-faceted financial planning and decision-making scenarios, enabling the model to capture subtleties in financial reasoning and advice formulation. This depth ensures that models trained on the Financial Alpaca dataset can understand both foundational financial principles and more advanced concepts, making them valuable for applications requiring detailed financial analysis or personalized financial advice.

#### 

#### **2. FinTalk-19k Dataset**

The FinTalk-19k Dataset is sourced from Hugging Face and includes data extracted from public Reddit discussions. This dataset is systematically organized with annotations across three primary categories: "Personal Finance," "Financial Information," and "Public Sentiment." These categories are designed to structure the data around relevant themes, allowing the model to differentiate between distinct financial contexts and nuances, which ultimately enhances its ability to provide contextually relevant responses. With over 19,000 entries, the FinTalk-19k dataset offers a comprehensive view of real-world finance-related conversations, capturing not only factual information but also a wide spectrum of public sentiment.

The dataset addresses a range of topics including budgeting strategies, savings practices, investment advice, and general financial literacy queries, reflecting common financial challenges and concerns among users. Its focus on public sentiment allows the model to understand varying attitudes towards financial issues, such as risk tolerance in investments or popular trends in personal finance management. This depth of information equips models trained on FinTalk-19k to interpret user intentions more effectively, provide empathetic responses, and respond to inquiries with a greater awareness of consumer sentiment and behavior.

The FinTalk-19k dataset’s foundation in authentic user-generated content from Reddit provides a unique advantage, as it captures informal and diverse perspectives on financial topics. This makes it an invaluable resource for training language models aimed at personal finance applications, where understanding everyday language and common financial concerns is crucial for generating accessible and relatable advice.

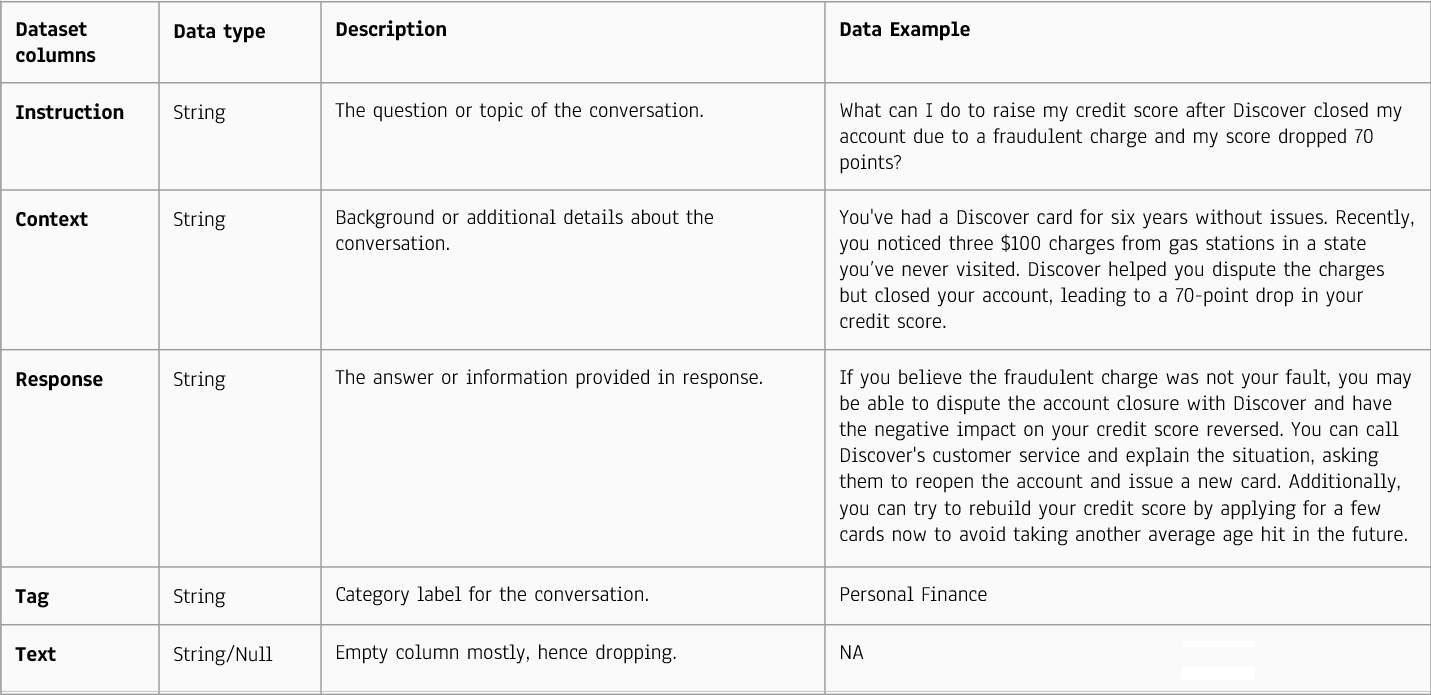


Figure 1. Sample Data

#### **b. Data Processing**

**Cleaning and Preprocessing:** Our initial data cleaning involves meticulous efforts to remove extraneous whitespace, dashes, and bullets, ensuring textual consistency across our financial datasets. Additionally, non-ASCII characters are stripped from the text to standardize encoding and prevent processing errors, crucial for maintaining high data integrity.

**Text Normalization and Tokenization:** The text is normalized by converting all characters to lowercase and standardizing punctuation across the dataset. This standardization aids in maintaining consistency, making it easier for our AI models to process and learn from the data. Subsequently, the normalized text is tokenized, breaking it down into smaller units (tokens) that serve as the basic input for our AI models.

**Embedding and Clustering:** After tokenization, the instructions are converted into embeddings using the sentenceTransformer model (specifically, the all-MiniLM-L6-v2 model). These embeddings, which are high-dimensional vector representations of the text, capture the semantic meaning of the instructions. We then apply K-means clustering (with k=5) to these embeddings to group similar instructions, as demonstrated in the clustering analysis.

**Dimensionality Reduction and Visualization:** To facilitate a clearer understanding and interpretation of these clusters, we utilize t-SNE to reduce the dimensionality of the data to 2D. This reduction allows us to visually represent the clustering of data, as shown in the figure. **Figure 2** provides a t-SNE visualization of clusters for instructions, where each color represents a different cluster, illustrating the grouping based on semantic similarity. This visualization helps validate the effectiveness of our clustering approach and offers intuitive insights into the data's underlying structure, enhancing our model's training and responsiveness to varied financial queries.

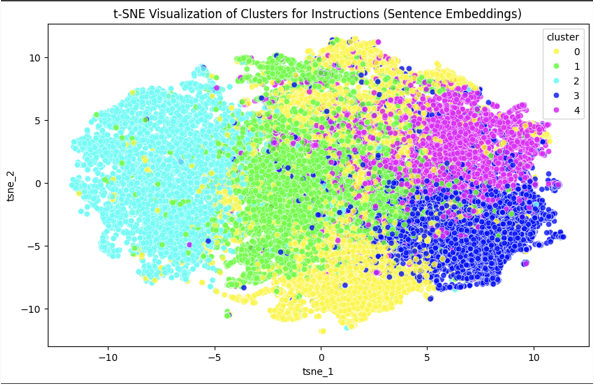


Figure 2. Cluster Analysis

#### **c. Exploratory Data Analysis**

#### **Word Cloud Analysis:** The word cloud generated from the instructions in our dataset reveals the most frequent terms such as "credit," "investment," and "market." These terms are pivotal as they highlight the primary areas of concern and interest among users seeking financial advice. This analysis ensures our model's focus is finely attuned to the core themes within the financial queries it will address. The prominence of such terms in the dataset's vocabulary is depicted in the word cloud visualization as shown in the accompanying Figure 3.

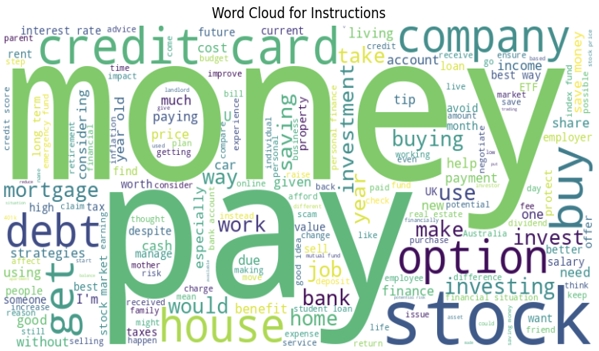


Figure 3. Word Cloud Analysis

#### **Distribution Analysis:** The distribution of instruction lengths within our dataset is critical for determining the optimal context window size for our Large Language Models (LLMs). The histogram displays a concentration of queries predominantly ranging between 40 to 60 words. This insight helps in configuring the model to handle typical user inputs efficiently, ensuring that the model is both effective and resource-efficient in processing financial queries. The specifics of this distribution are outlined in the histogram of instruction lengths.

#### **Token Distribution Analysis:** Analyzing the token distribution assists in fine-tuning the model by identifying the most common response lengths, which primarily fall within a 200-300 token range. A context window size of 256 tokens is selected based on this analysis, optimizing the model’s ability to process the majority of responses without excessive computation. This strategic choice enhances the model's responsiveness and is critical for maintaining computational efficiency, as detailed in the token distribution analysis.

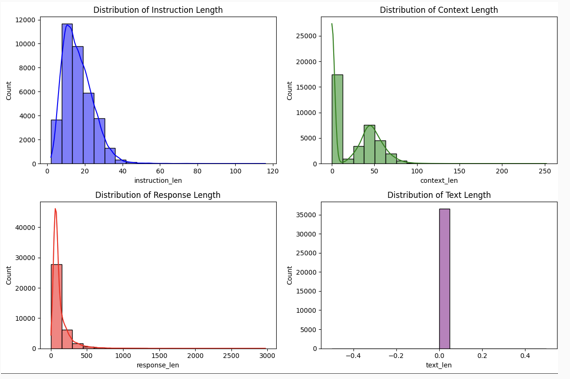
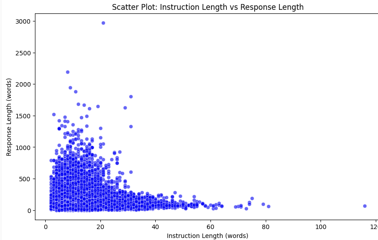
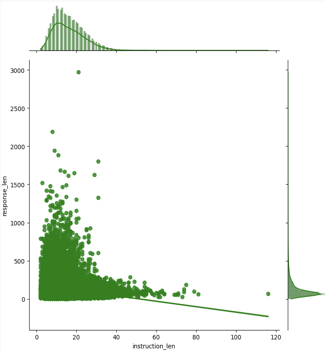


Figure 4. Token Distribution Analysis

#### **Correlation and Pairwise Relationships:** Our analysis extends to exploring the relationships between the lengths of instructions and responses. A scatter plot reveals a mild positive correlation, suggesting that longer queries generally produce more detailed responses. This relationship underpins our approach to model training, emphasizing the need to adjust the model's sensitivity to the length of user inputs. The Pearson Correlation Coefficient further quantifies this relationship, with the scatter plot and correlation analysis providing a clear visual and statistical representation of how query lengths influence response characteristics as illustrated in the scatter plot and correlation coefficient Figure 5.

=

#### **Insights Using Pairwise Relationships**

Figure 5. Correlation Relationships

### **Pearson Correlation Coefficient:** To test the relationship between instruction length and response length, the Pearson correlation coefficient was calculated. This coefficient, denoted as ff, measures the strength and direction of a linear relationship between two variables. The calculated Pearson correlation coefficient was **-0.17**, indicating a weak negative correlation between instruction length and response length. This value suggests that there is no meaningful relationship between the two variables, as the correlation is both weak and negative. Consequently, this analysis does not support the hypothesis that "longer instruction lengths lead to longer response lengths." Instead, response length appears to be largely independent of instruction length, implying that other factors may play a more significant role in determining response length. This insight is valuable for model optimization, as it suggests that focusing on instruction length alone may not be effective for predicting response length within the model.

**Pairwise Relationships:** The pairwise relationship plots offer a deeper understanding of how different textual features relate to each other. These plots reveal non-linear relationships between features, suggesting complex interactions that are not immediately obvious through linear analysis alone. Such insights indicate the presence of nuanced patterns in the data, which could inform more sophisticated modeling strategies. The detailed insights from these relationships are captured in the pairwise plots as demonstrated in the pairwise relationship Figure 6.

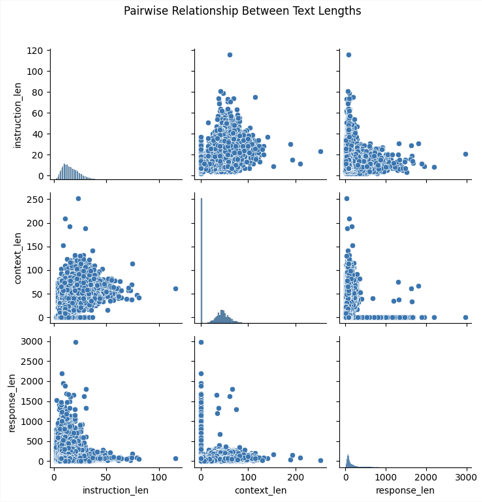


Figure 6. Pairwise Relationships

**Instruction vs. Context Length** (top row, middle column): This shows a **slight positive correlation**, meaning longer instructions tend to have somewhat longer contexts. However, the spread indicates that many short instructions still have longer contexts.

**Instruction vs. Response Length** (top row, right column): This shows a **weak correlation**; longer instructions don’t necessarily mean longer responses. The response length varies widely even for short instructions.

**Context vs. Response Length** (middle row, right column): This scatterplot shows a **clearer trend**. Longer contexts tend to be associated with longer responses, though the correlation is not perfect.

#### **d. Modeling**

**Fine-Tuning with LoRA (Low-Rank Adaptation):**

**LoRA Overview:** LoRA modifies a minimal set of model parameters, enabling rapid adaptation of our pre-trained language model to specialized financial contexts without the need for extensive retraining. The structure of this process is illustrated in Figure 9, highlighting how data flows from the training pipeline into the fine-tuning stages.

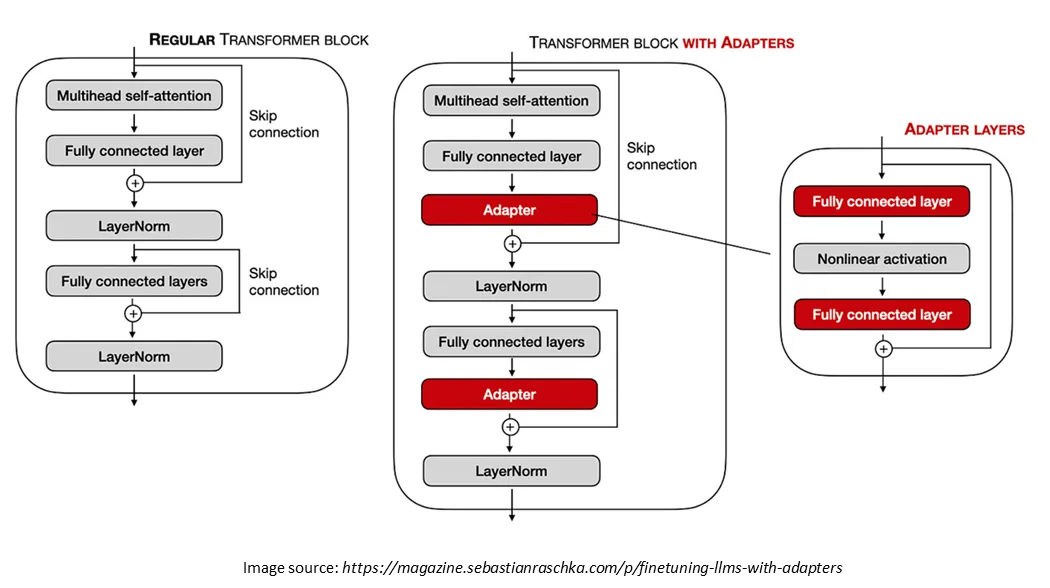


Figure 7: Parameter Efficient Fine Tuning

Parameter-efficient Fine-tuning overcomes the problems of consumer hardware, storage costs by fine tuning only a small subset of model’s parameters significantly reducing the computational expenses while freezing the weights of original pretrained LLM. In the above **Figure 7**, we can see that adapter layers are added after multi-head attention and feed-forward layers in the transformer architecture. The parameters of these added layers are only updated during fine-tuning while keeping the rest of the parameters frozen.

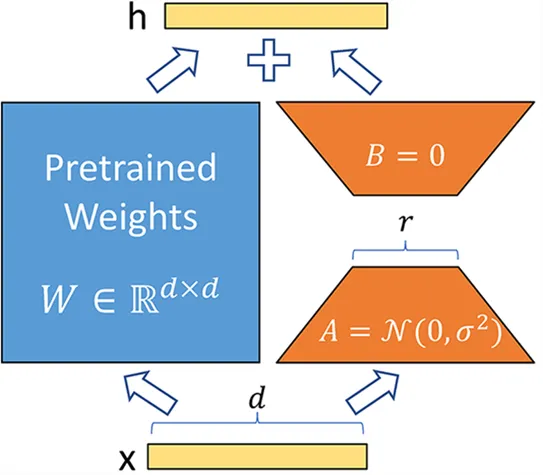


Figure 8: Low Rank Adaptaion Matrix Manipulation

LoRA papers says that this method can minimize the number of trainable parameters by up to 10,000 times and the GPU memory necessity by 3 times while still performing on par or better than fine-tuning model quality on various tasks. As observed in **Figure 8**, Rather than altering the weight matrix W of a layer in all of its components, LoRA creates two smaller matrices, A and B, whose product roughly represents the modifications to W. The adaptation can be expressed mathematically as , where A and B are the low-rank matrices. If W is an matrix A might be and B is where r is rank and much smaller than m,n. During fine tuning only A and B are adjusted enabling the model to learn task specific features.

**Training:** Through LoRA adaptation, the model fine-tunes its understanding of financial data, significantly improving its ability to generate context-sensitive and accurate financial advice.In our training, we are using PEFT with QLoRA configuration with 8-bit precision to adaptively fine-tune a causal language model by applying low-rank matrix adaptations on the **'q\_proj'** and **'v\_proj'** modules, optimizing memory and computational efficiency for training on limited VRAM.

**Performance Tracking:**

We use **Weights & Biases (wandb)** for tracking model performance and metrics during experiments, ensuring continuous improvement and validation of the model’s capabilities. As depicted in Figure 3, this tool allows us to visualize training progress and compare different model versions effectively.

**Weights & Biases** also enables us to optimize model performance based on real-time data and collaborate seamlessly by sharing results and insights across the team, enhancing our project's development cycle.

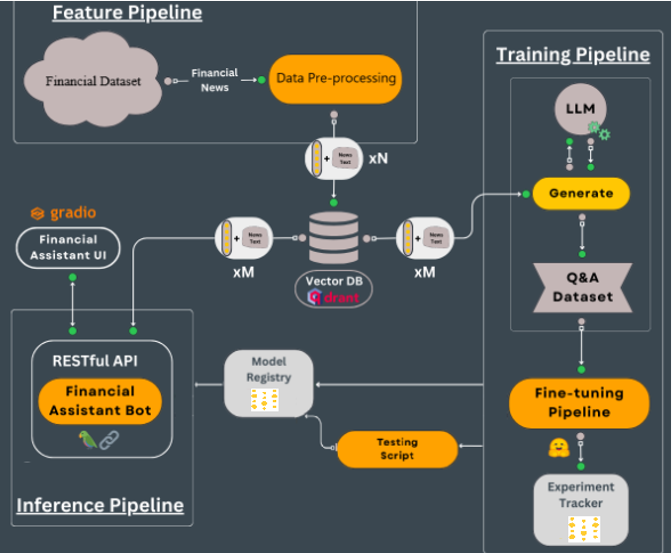


Figure 9. Model Outline [(Iusztin P. The framework to deploy LLM systems to do RAG | Decoding ML. Medium.)](https://medium.com/decodingml/the-llms-kit-build-a-production-ready-real-time-financial-advisor-system-using-streaming-ffdcb2b50714)

**e. Model Deployment: RESTful API and Gradio Interface**

The fine-tuned model is deployed through a RESTful API, which is designed to handle requests in real-time.

**Gradio** serves as the interactive front end where users can pose financial questions and receive advice directly. This setup not only facilitates easy user interaction but also ensures that the advice provided is based on the latest financial data processed by the model.

**f. Inference Pipeline:** The inference pipeline is crafted to support real-time data processing and efficient model inference.

Integration of the retrieval-augmented generation framework allows the model to access the most relevant and recent financial data quickly.

This setup guarantees that each response generated by the model is both up-to-date and contextually informed, providing users with reliable and actionable financial advice.

### **5. Results**

### **Current Performance Metrics**

Our model training is rigorously monitored through various metrics to ensure its effectiveness and efficiency, particularly focusing on the LLaMA 7B model's adaptation process. The training loss and perplexity score, which are key indicators of model performance, have shown significant improvement over several iterations.

**Training Loss Reduction for LLaMA 7B:** The training loss for the LLaMA 7B model decreased steadily from 2.2643 at step 50 to 1.527531 at step 250. This trend is clearly illustrated in the provided Figure 10. This reduction in loss indicates that the LLaMA 7B model is effectively learning from the training data, continuously improving its ability to generate accurate responses. Each training step enhances the model's proficiency in handling complex financial queries, which is essential for delivering reliable financial advice.

**Perplexity Score to assess Llama 7B’s accuracy in predicting sequences:** A lower perplexity score indicates that the model is better at understanding the context and predicting subsequent tokens, which is essential in ensuring coherent and contextually relevant responses in financial advisory contexts. While **SOTA (state-of-the-art) models typically achieve perplexity scores in the range of 4 to 6**, our model currently achieves a perplexity score of **7.12**. Given the **limited computational resources**, this result is fair and demonstrates that our model is performing close to industry-leading standards, particularly considering the constraints. Monitoring perplexity allows us to measure the model's capability to generate linguistically and semantically accurate sequences, which is vital for practical applications in real-time financial guidance.

**Other Metrics for Future Integration:** Looking ahead, we plan to integrate additional metrics such as the ROUGE Score, to measure the overlap between the model-generated responses and the reference texts. This metric will provide further insights into the model's linguistic capabilities and its effectiveness in simulating human-like financial advice. The integration of these metrics will further solidify our understanding of the model's performance across different linguistic and contextual challenges.

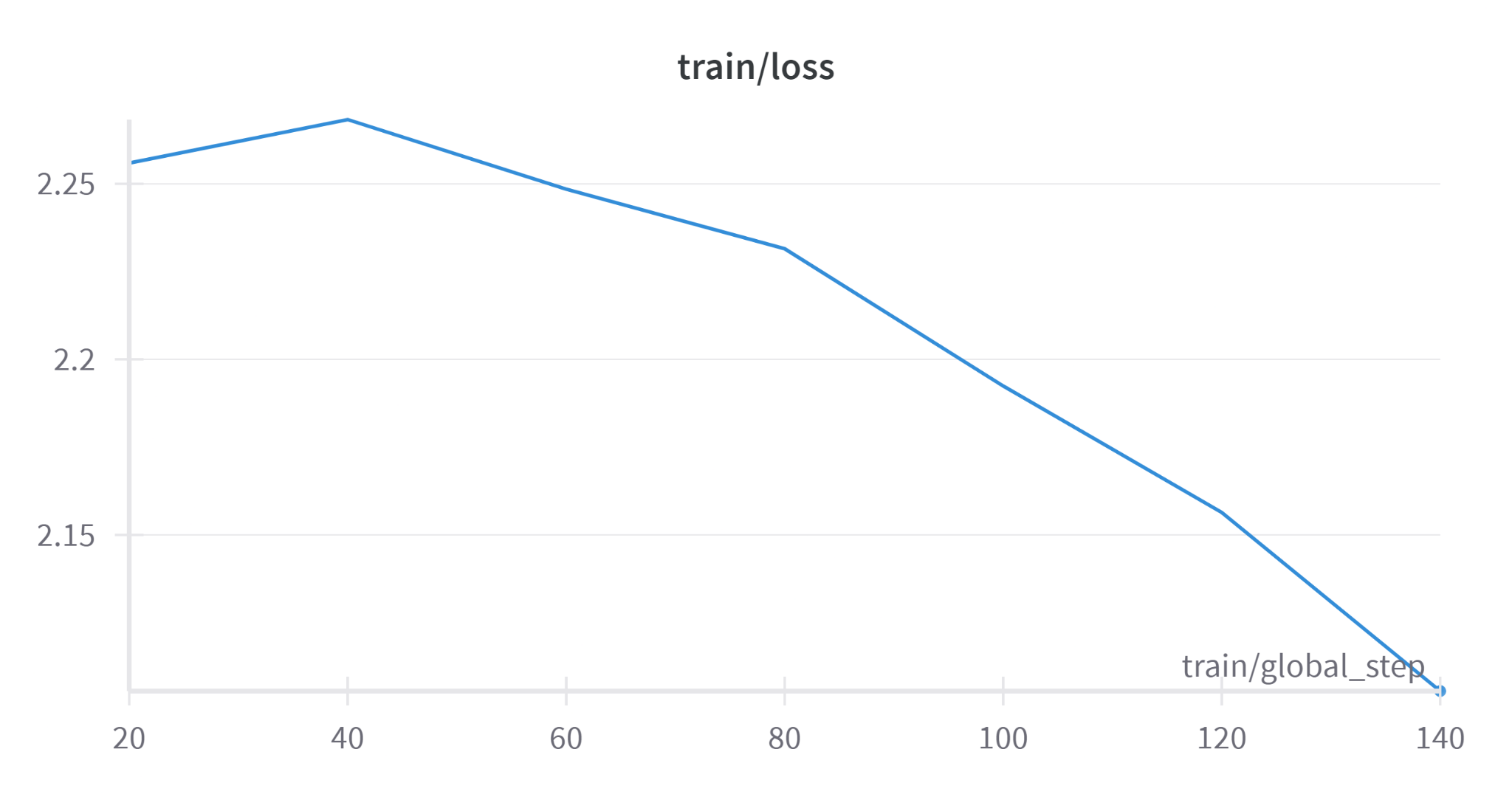


Figure 10:. Fine Tuning Training Loss

#### 

#### **Challenges Impacting Model Training and Performance**

Despite the progress, we face several challenges that affect the model’s training efficiency and scalability:

**VRAM Limitations:** Our current setup is constrained by insufficient VRAM, which limits the model’s capacity to handle large batches of data. This restriction has direct implications on the training efficiency, as it prevents the model from processing larger data sets in parallel, thus extending the training duration and impacting the speed of model iterations.

**GPU Constraints:** The limited availability of powerful GPUs slows down the model training process and increases processing time. This not only affects the model’s ability to scale to more extensive training regimes but also hampers our ability to quickly iterate and improve the model based on continuous feedback.

### **6. Next Steps for Phase 2**

### As the project progresses into Phase P2, several key components have been outlined to advance the functionality and scalability of the financial advisory tool:

### **a. Feature Pipeline**

### The **feature pipeline** has been completed, encompassing data acquisition and preprocessing from financial datasets. This stage involved transforming raw financial data into a structured format suitable for embedding and subsequent model training.

### **b. Training Pipeline and Model Fine-Tuning**

### The training pipeline involves the **fine-tuning of large language models (LLMs)** on financial data to enhance accuracy in responding to financial queries. **LLaMA Fine-Tuning** has already been completed, while **Mistral Fine-Tuning** is currently in progress. Should time allow, additional LLMs may be explored and fine-tuned to further improve the system's robustness and adaptability to various financial contexts.

### **c. Vector Data Pipeline**

### The **vector data pipeline** is in progress, using a vector database (Qdrant) to store high-dimensional embeddings. Currently, the data is being sourced directly and this component is essential for efficient similarity searches and retrieval of relevant financial context, enabling the model to provide precise and contextually accurate responses.

### **d. Inference Pipeline**

### In the inference pipeline, the focus is on building the **API and user interface**. The RESTful API will serve as the backbone for the financial assistant bot, allowing users to interact with the model seamlessly. Additionally, the **Financial Advisor UI** will be developed to provide users with a streamlined, intuitive platform for accessing personalized financial guidance. The user interface will be integrated with Gradio, allowing easy deployment and testing.

### 

### **7. Discussion**

The preliminary findings from FinAdvisor, an AI-driven financial advisory tool, demonstrate significant enhancements in decision-making and efficiency. Leveraging the LLaMA 7B model, FinAdvisor automates complex financial tasks, which not only speeds up financial decisions but also increases accuracy and reduces errors. This progress is evident from the consistent reduction in training loss observed during the model's iterative training, which ensures that the tool is becoming increasingly proficient at managing complex financial scenarios.

FinAdvisor has also shown its capacity to offer personalized financial advice, aligning closely with essential personal finance topics such as "investment," "risk," and "credit." This alignment, confirmed through exploratory data analysis, suggests that the model is well-equipped to handle key areas of financial management, enhancing its ability to provide tailored advice. The upcoming integration of additional metrics like the ROUGE Score will further refine its ability to predict sequences and assess the overlap between generated responses and reference texts, enhancing its linguistic accuracy and contextual relevance.

Furthermore, by utilizing advanced LLM technology, FinAdvisor makes expert financial advice more accessible and affordable, democratizing financial expertise for widespread use. However, the tool faces computational challenges, including VRAM and GPU limitations, which currently restrict its ability to scale and extend to more extensive datasets without compromising performance. Addressing these challenges will be crucial for improving the scalability and accessibility of the services.

Looking forward, expanding the dataset to cover a broader range of financial scenarios will enhance the model's versatility, and refining the tokenization and data retrieval processes will improve its efficiency. Early incorporation of user feedback and a broader array of performance metrics will enable FinAdvisor to align more closely with user expectations and adapt more responsively to various financial contexts. By tackling these areas, FinAdvisor aims to support a diverse range of user needs, providing personalized and actionable financial advice that adapts to the complexities of modern financial landscapes.

### **8. Statement of Contributions**

The following outlines the contributions of each team member, who equally participated in brainstorming, researching, and implementing the project:

**Krishna Venkatesh:**

* Led data collection from the Alpaca News API, ensuring the data was relevant and robust for financial analysis.
* Managed text normalization processes, including removing special characters and standardizing text to ensure consistency across datasets.
* Contributed significantly to exploratory data analysis, creating visualizations such as word clouds to highlight prevalent financial terms.

**Laasya Anantha Prasad:**

* Handled text tokenization and data embedding, optimizing the efficiency of data storage and retrieval.
* Actively participated in EDA, identifying key financial terms to refine data preprocessing.
* Assisted in modeling adjustments to improve the accuracy and responsiveness of the financial advice generated.

**Rakshak Kunchum:**

* Supported high-quality data collection from the Hugging Face finance-alpaca dataset, ensuring data suitability for advanced model training.
* Researched and implemented advanced data storage techniques to facilitate efficient and scalable model performance.
* Enhanced the implementation of the LLaMA 7B model, contributing to improvements in real-time data handling and response accuracy.

### **9. References**

1. Alpaca Real-Time News API. *Alpaca Market News API Documentation*. Available at: <https://docs.alpaca.markets/docs/streaming-real-time-news>
2. Gaurav Bharti. "Finance Alpaca Dataset." *Hugging Face Datasets*. Available at: <https://huggingface.co/datasets/gbharti/finance-alpaca>
3. Hu, E. J., Shen, Y., Wallis, P., Allen-Zhu, Z., Li, Y., Wang, S., & Chen, W. "LoRA: Low-Rank Adaptation of Large Language Models." *arXiv preprint arXiv:2106.09685*. Available at: <https://arxiv.org/abs/2106.09685>
4. Izacard, G., & Grave, E. "Leveraging Passage Retrieval with Generative Models for Qdrant. "Qdrant Vector Database Documentation." Available at: <https://qdrant.tech/>Open Domain Question Answering." *arXiv preprint arXiv:2007.01282*. Available at: <https://arxiv.org/abs/2007.01282>
5. Gradio. "Gradio Documentation." Available at: https://gradio.app/docs/
6. Comet. "Comet Machine Learning Experiment Management." Available at: <https://www.comet.ml/>

### **10. Appendix**

Github Link : <https://github.com/sriksven/Financial_Advisor_LLM>