### **1. Title:** FinAdvisor

### **2. Authors:** Krishna Venkatesh, Laasya Anantha Prasad, Rakshak Kunchum

### **3. Github:** https://github.com/sriksven/FinAdvisor\_LLM

### **4. Summary**

In the financial sector, rapid and precise analysis of expansive datasets is critical for making informed decisions about investments, risk management, and personalized financial advice. Traditional methods often struggle to keep up with the volume and velocity of data generated, such as market reports and real-time financial news, leading to potential gaps in decision-making. The Financial LLM Advisor project is designed to address these challenges by leveraging the capabilities of large language models (LLMs) fine-tuned for the financial domain. These AI models can process and interpret complex financial data effectively, offering insights that are beyond the scope of conventional tools.

The dataset for this project consists of a variety of financial documents, including news articles, regulatory filings, and other timely financial data from trusted sources.

### **5. Proposed Plan:**

The Financial LLM Advisor project employs a multi-stage methodology to enhance decision-making in the financial sector by leveraging AI, particularly large language models (LLMs) fine-tuned for financial applications. Initially, data collection encompasses diverse financial sources—ranging from market reports to real-time financial news—which is then cleaned to remove irrelevant information and transformed into a structured format for model training. This prepared data is stored in a vector database, ensuring efficient retrieval for real-time analysis.

In the second phase, the project utilizes Low-Rank Adaptation (LoRA) to fine-tune pre-trained LLMs on a Q&A dataset derived from financial texts. This method optimizes the model to understand complex financial language and context accurately. Integration of Retrieval-Augmented Generation (RAG) further refines the system's responsiveness, pulling relevant data during inquiries to provide precise financial guidance and reduce errors typically associated with AI "hallucinations."

The deployment of the system via a RESTful API facilitates real-time interaction with users, offering personalized financial advice based on dynamic data inputs. Tools such as Comet are employed for continuous monitoring and refinement of the model's performance based on actual user interactions. This ongoing feedback loop enhances the system’s accuracy and reliability, ensuring the financial advice remains relevant to current market conditions and user-specific scenarios. Through this advanced AI application, the project aims to transform traditional financial advisory services by providing faster, data-driven insights that are scalable and adaptable to individual needs.

### **6. Preliminary results:**

**Word Cloud** visualization showcases common terms related to investment, such as "investing," "market," and "growth," with larger words indicating higher frequency in the dataset.



Fig 1 : Word Cloud

**Prompt Length Distribution**, the histogram shows the distribution of prompt lengths, highlighting that most samples have lengths between 120 and 140 tokens, with clear upper and lower token limits marked in red and blue.

### 

Fig 2 : Prompt Length distribution

### **7. Project Milestones**

1. **Weeks 1-3**: Data collection, cleaning, embedding, and exploratory analysis.
2. **Weeks 4-6**: Fine-tuning the LLM with LoRA, and tracking model performance.
3. **Weeks 7-8**: Model evaluation, performance metrics, and refinement.
4. **Weeks 9-10**: Integration of RAG for real-time data retrieval and testing.
5. **Weeks 11-12**: API development and testing for the financial assistant bot.

**Mid-Phase Goals**:

1. Finalize the data cleaning and ensure that all collected datasets are ready for training, stored efficiently in a **Vector Database (Qdrant)** for future retrieval.
2. Ensure that the fine-tuning process has reached acceptable levels of performance on financial text tasks, with progress tracked through **Comet**.
3. The model successfully integrates with **Qdrant**, pulling accurate and relevant data in real-time during user interactions, reducing risks of hallucinations.

**End-Phase Goals:**

1. The model should demonstrate stable and improved performance based on predefined financial-specific benchmarks, showing clear improvements after fine-tuning using LoRA and tracked metrics via Comet.
2. The Retrieval-Augmented Generation (RAG) integration should work seamlessly, accurately retrieving and integrating real-time data from Qdrant into the model’s responses, with minimal hallucination errors.

### **8. References**

1. **Financial News Articles and Blogs**. A collection of current and historical articles sourced from financial media outlets such as Reuters, Bloomberg, and The Financial Times.
2. **Hugging Face**. Pre-trained Large Language Models (LLMs). Available at: [https://huggingface.co](https://huggingface.co/)
3. **Qdrant Vector Database**. High-performance vector database for storing embeddings. Available at: [https://qdrant.tech](https://qdrant.tech/)
4. Hu, E. J., Shen, Y., Wallis, P., Allen-Zhu, Z., Li, Y., Wang, S., Wang, L., & Chen, W. (2022). *LoRA: Low-Rank Adaptation of Large Language Models*. Available at: <https://arxiv.org/abs/2106.09685>
5. Lewis, P., Perez, E., Piktus, A., Petroni, F., Karpukhin, V., Goyal, N., Küttler, H., Lewis, M., Yih, W., Rocktäschel, T., Riedel, S., & Kiela, D. (2020). *Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks*. Available at: <https://arxiv.org/abs/2005.11401>
6. **Comet**. Machine Learning Experiment Tracking Tool. Available at: [https://www.comet.ml](https://www.comet.ml/)
7. **Flask/ FastAPI**. RESTful API frameworks for Python. Available at: https://flask.palletsprojects.com/ and https://fastapi.tiangolo.com/