

CO3: AI-Powered Financial Portfolio Rebalancer Using LangChain with Multiple LLM Integration

Introduction:

This report details the implementation and evaluation of an AI-powered financial portfolio rebalancer using LangChain with multiple Large Language Model integrations. The system is designed to analyze investment portfolios and recommend rebalancing actions to maintain a target allocation strategy. Three different LLMs were integrated and evaluated: OpenAI's GPT-4o, Groq's LLaMA3-8B, and Groq's LLaMA3-70B. This report covers the implementation details, challenges encountered, comparative analysis of the LLMs, and recommendations for their use in financial analysis.

Implementation:

The portfolio rebalancer was built using LangChain's agent framework, which enabled dynamic tool selection for financial analysis tasks. I created a python notebook on colab with the functions set in individual cells, while also including the required pip installs in the first cell. The API keys were assigned to a constant in the beginning of the notebook, and referenced across the file. The core components included:

- **Stock Price Lookup Tool**

Utilizes the Yahoo Finance API to fetch real-time stock data including current price, daily change, percentage change, and company information.

- Handling of different input types (strings, symbols, sets)
- Error recovery for problematic API responses
- Stock information including price, daily change, and company details
- Proper extraction of pricing data and calculation of percentage changes

- **Portfolio Rebalancing Tool**

Analyzes portfolio allocations and suggests rebalancing actions to achieve equal weight distribution among assets.

- Parses portfolio representations from various input formats
- Calculates current weights and compares to target allocations
- Determines specific buy/sell actions with exact share quantities
- Presents recommendations in a clear, structured format with dollar values and percentage targets

- **Market Trend Analysis Tool**

Retrieves and analyzes market indices data to provide context on overall market sentiment.

- Analyzes major market indices (S&P 500, Dow Jones, NASDAQ, Russell 2000)
- Calculates key metrics including weekly performance and volatility
- Determines overall market sentiment based on index performance
- Handles pandas Series data properly to avoid ambiguous value errors

Challenges Encountered:

- **Yahoo Finance API**

The Yahoo Finance API occasionally returned incomplete data or errors, particularly when attempting to retrieve multiple stocks simultaneously. To counter this, I implemented error handling and fallback mechanisms. For multi-symbol requests, I processed individual symbols when batch requests fail.

- **Pandas Series Type Handling**

My initial implementations faced "truth value of a Series is ambiguous" errors and FutureWarnings when working with pandas Series objects. I then modified the code to properly extract scalar values from pandas Series using `.iloc[0]` instead of direct `float()` casting, eliminating warnings while maintaining functionality.

- **Portfolio String Parsing**

LLMs provided portfolio inputs in various formats (Python dictionaries, JSON strings) at times, leading to parsing errors. I implemented a flexible parsing system that handles multiple input formats, including proper JSON parsing, Python eval, and custom string parsing for edge cases.

- **LLM Response Variability**

Different LLMs exhibited varying approaches to tool selection and usage, with some making redundant calls or encountering parsing errors. To fix some of the common errors encountered, I added a lambda wrapper for the Market Trend Analyzer to handle unexpected inputs. And also added try/except blocks to prevent single failures from halting the entire process.

Test Scenarios:

- **Portfolio 1 (Unbalanced)**

- Initial allocation: AAPL (50%), TSLA (30%), GOOGL (20%)
- Target allocation: Equal weight (33.33% each)
- Rebalancing required: Yes

- **Portfolio 2 (Balanced)**

- Initial allocation: MSFT (25%), NVDA (25%), AMZN (25%), META (25%)
- Target allocation: Equal weight (25% each)
- Rebalancing required: No

LLM Performance Analysis:

LLM	Portfolio 1	Portfolio 2	Remarks
OpenAI GPT-4	Correct	Correct	Perfect accuracy and reasoning
LLaMA3-8B	Incorrect	Correct	Incorrectly concluded Portfolio 1 was still out of balance
LLaMA3-70B	Correct	Correct	Added price information

GPT-4 excelled with consistently clear and focused responses that precisely addressed the rebalancing requirements without unnecessary elaboration. Its responses for both portfolios were direct and perfectly aligned with the rebalancing tool's recommendations.

LLaMA3-8B suffered significant issues, including incorrect conclusions about Portfolio 1 still being out of balance despite having the correct rebalancing instructions, revealing potential reasoning deficiencies. It also repeatedly invoked invalid actions (calling "None") and unnecessarily repeated tool calls for Portfolio 2, taking considerably longer than expected although it did finish with the expected result in the end.

LLaMA3-70B produced coherent and accurate responses that correctly identified both the imbalanced state of Portfolio 1 and the already-balanced state of Portfolio 2, demonstrating strong logical reasoning capabilities. Its responses were more verbose than GPT-4's.

LLM Response Time Analysis:

LLM	Portfolio 1	Portfolio 2	Average
OpenAI GPT-4	4.6	4.78	4.69
LLaMA3-8B	4.81	24.18	14.5
LLaMA3-70B	6.31	4.72	5.52

OpenAI's GPT-4 demonstrated the most consistent performance with an average execution time of 4.69 seconds across both portfolio scenarios.

In contrast, LLaMA3-8B showed highly variable performance, with a 4.81 seconds for the simpler Portfolio 1 with the incorrect result but a concerning 24.18 seconds for Portfolio 2, yielding an inefficient average of 14.50 seconds. This inconsistency makes LLaMA3-8B unpredictable for time-sensitive financial applications.

LLaMA3-70B maintained reasonable execution times averaging 5.52 seconds, placing it between the other models but closer to GPT-4's performance.

LLM Response Tool Use Analysis:

Aspect	GPT-4	LLaMA3-8B	LLaMA3-70B
Portfolio 1 Tool Calls	1 (PortfolioRebalancer)	4 (3× PortfolioRebalancer, 1× MarketTrendAnalyzer)	4 (1× PortfolioRebalancer, 3× StockPriceLookup)
Portfolio 2 Tool Calls	1 (PortfolioRebalancer)	3 (1× PortfolioRebalancer, 1× StockPriceLookup, 1× MarketTrendAnalyzer) + 3 invalid "None" attempts	2 (1× PortfolioRebalancer, 1× MarketTrendAnalyzer)
Total Valid Tool Calls	2	7	6
Invalid Action Attempts	0	3	0
Necessary Calls	2	2	2
Redundant Calls	0	2	0
Efficiency	High	Low	Medium

GPT-4 was extremely efficient with its tool use by immediately identifying and utilizing the PortfolioRebalancer tool for both scenarios without unnecessary tool calls. For Portfolio 1, it precisely recognized the need for rebalancing, and for Portfolio 2, it correctly determined no action was required.

LLaMA3-8B struggled significantly with tool selection, repeatedly calling the same PortfolioRebalancer tool three times in succession for Portfolio 1 despite getting identical results each time, and making redundant calls to StockPriceLookup and MarketTrendAnalyzer for Portfolio 2 when no rebalancing was needed, causing it to take significantly longer.

LLaMA3-70B matched GPT-4's perfect efficiency for Portfolio 2 but made additional StockPriceLookup calls for Portfolio 1 beyond what was strictly necessary for the rebalancing recommendation, slightly reducing its overall efficiency while still maintaining strong performance.

Quality of Financial Advice:

GPT-4 provided great and correct financial advice, with its recommendations concise and complete, providing exact share quantities to trade for Portfolio 1 and correctly identifying no action was needed for Portfolio 2.

LLaMA3-8B's advice was concerningly subpar, with its final assessment for Portfolio 1 incorrectly stating that the portfolio is still out of balance and may require further analysis despite having already generated the correct rebalancing instructions earlier in its reasoning chain, demonstrating a failure to maintain consistent financial analysis. For Portfolio 2, it unnecessarily complicated the analysis by looking up stock prices and market trends when the portfolio was already perfectly balanced.

LLaMA3-70B also provided quality financial advice comparable to GPT-4, correctly calculating the exact shares to buy and sell for Portfolio 1 while also appropriately verifying current stock prices. For Portfolio 2, it correctly identified no rebalancing was needed while also providing valuable context from market trend analysis that reinforced the decision to maintain the current allocation.

Overall LLM Analysis:

OpenAI GPT-4 demonstrated exceptional capabilities for financial tasks, with the best reasoning, consistent performance, and efficient information processing that made it ideal for complex portfolio rebalancing. However, it showed limited initiative in seeking contextual market data and its higher API costs could be prohibitive for routine analyses that don't require its advanced reasoning capabilities.

LLaMA3-8B offered reasonable processing speed for simple financial queries and represented the most economical option for high-volume, non-critical tasks. However, it suffered from serious deficiencies in financial reasoning, exhibited redundant tool usage patterns, showed inconsistent execution times, and provided indecisive recommendations that would be problematic in real-world financial scenarios.

LLaMA3-70B achieved a strong balance between computational efficiency and reasoning capability while proactively investigating stock prices and market trends to provide valuable financial context. Its primary limitations included occasional verbosity in processing and a slight performance gap compared to GPT-4's precision.

LLM Recommendations:

For high-stakes investment decisions where accuracy and reliability are paramount regardless of cost, GPT-4 is the optimal choice due to its superior reasoning consistency and performance, though human verification remains essential for critical transactions.

For routine portfolio monitoring and standard rebalancing operations, LLaMA3-70B offers the ideal balance between reliability and cost-efficiency, with its proactive market analysis providing valuable context.

LLaMA3-8B's cost benefits could be a factor in its adoption, but its subpar reasoning capabilities means that it should be avoided in financial settings where even small mistakes have major ramifications.

A multi-tiered approach combining models could optimize costs while maintaining appropriate quality controls. LLaMA3-70B could be used for standard screening and analysis, and GPT-4 for complex or high-value decisions with clear differentiation criteria based on portfolio size, complexity, and risk profile.