

Building an AI Investment Advisor: A Multi-Agent System for Real-Time Stock Analysis and Portfolio Recommendations



University of
New Hampshire

Nicolas Siagian
Department of Computer Science, University of New Hampshire

Introduction

- Retail investors struggle with:
 - Interpreting market data
 - Managing risk
 - Making timely decisions
- Existing tools:
 - Require manual analysis
 - Do not personalize recommendations
- This project introduces an AI-powered investment advisor that:
 - Uses real-time data
 - Adapts to user goals
 - Generates actionable recommendations

Approach

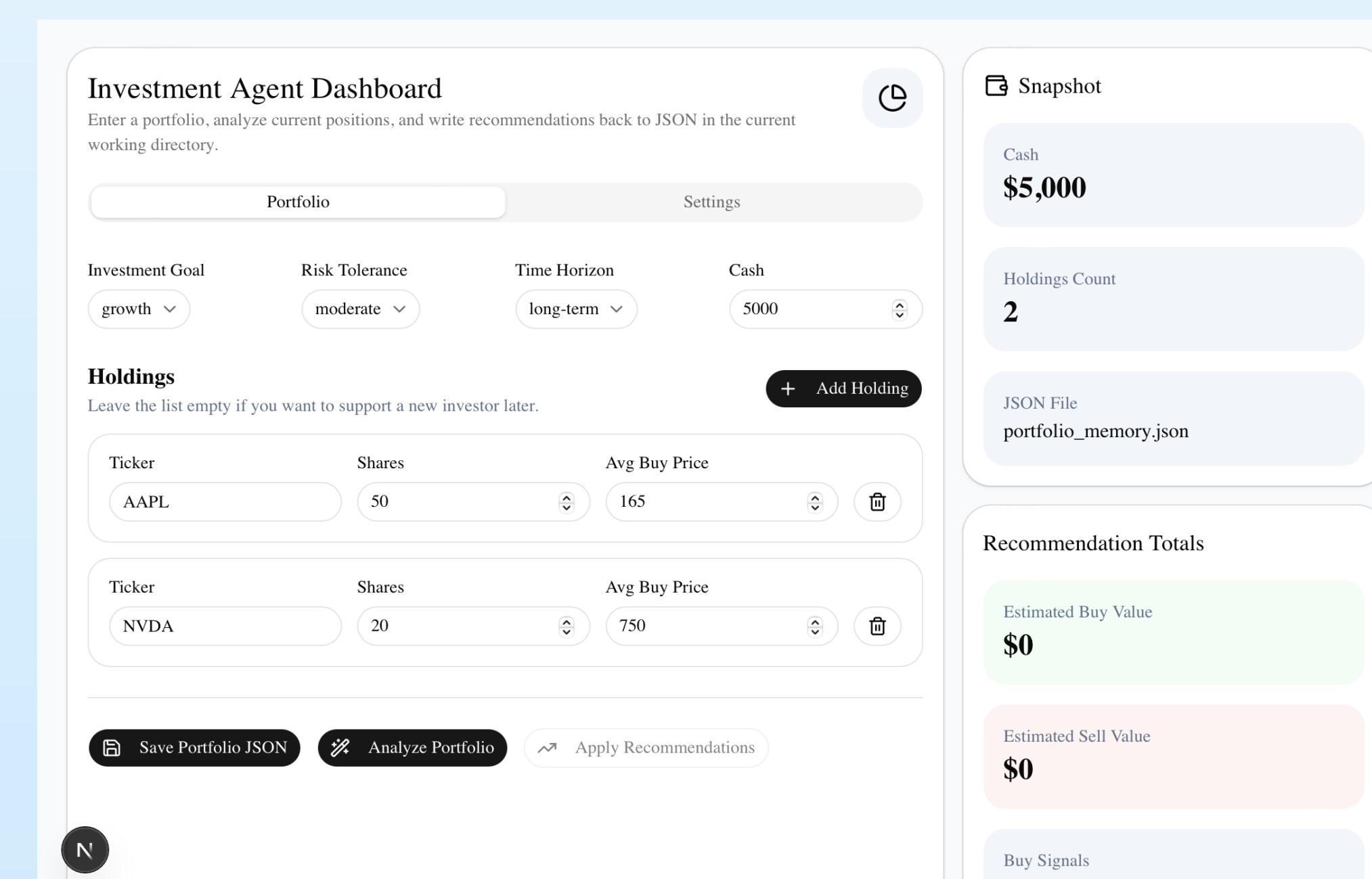
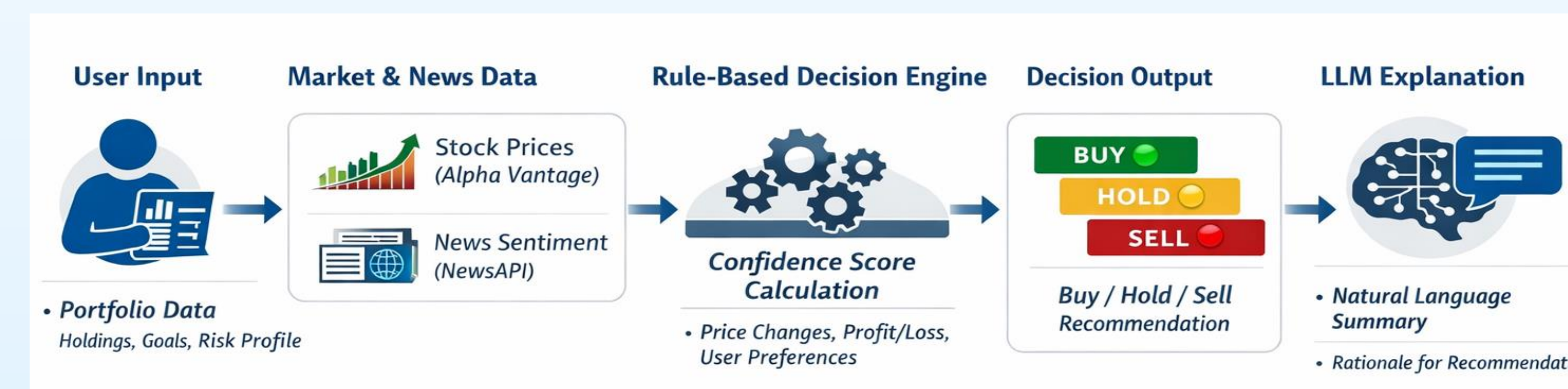
- Developed a multi-agent system that integrates news, market data, and user input to create a recommendation
- Rule-based logic agent builds a confidence score which factors in daily percent change, portfolio profit/loss, and user preferences (investment goals and risk tolerance)
- Uses an LLM to create reasoning for the confidence score

Results & Limitations

- System successfully generates real-time Buy/Hold/Sell recommendations
- Produces confidence scores (0–1 range) for each decision
- Generates natural language explanations using LLM
- Tested on multiple portfolios (AAPL, NVDA, etc.)
- Recommendations aligned with short-term price trends in most cases
- System consistently produces actionable outputs within seconds
- Automated portfolio updates based on suggested actions
- API limits caused minor delays but did not affect results
- Enabled persistent portfolio tracking through JSON-based memory
- Limitations:
 - Recommendations were rule-based but not predictive
 - Relies heavily on external data
 - You must execute trades manually
 - Explore advanced memory systems (e.g., vector databases) for adaptive learning

Workflow

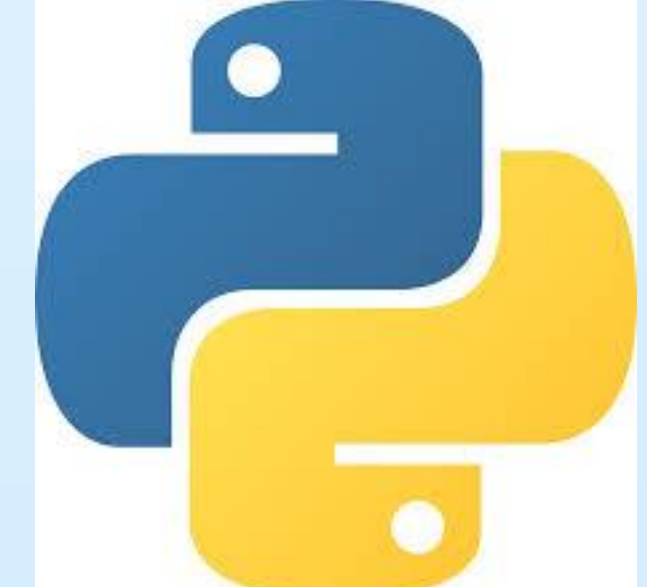
1. User inputs portfolio (tickers, shares, average buy price, goals, risk tolerance)
2. Frontend sends data to FastAPI backend
3. Backend retrieves stock prices (Alpha Vantage) and news data (NewsAPI)
4. Rule-based agent calculates: Profit/loss, Confidence score, Buy / Sell / Hold recommendation
5. LLM generates explanation using portfolio + market context
6. Results returned to frontend and displayed to user
7. Portfolio can be updated and saved to JSON memory



Success	Buy Signals						
Analysis complete.	2						
Analysis Results							
Each row can map directly to the response your Python service returns.							
Ticker	Recommendation	Confidence	Held Shares	PL %	Suggested Action	Est. Value	LLM Reasoning
AAPL	HOLD	65%	50	55.10%	BUY 10 shares	\$2,859.2	Given your existing 50 shares of AAPL with a significant profit of 55.1% and an average buy price of \$165.0, buying more shares at the current price of \$57.92 aligns with your growth investment goal. Despite recent market volatility sparked by geopolitical tensions, including Iran's retaliatory strike threats, AAPL's daily change of 0.11% suggests resilience. Additionally, positive news such as the promotion of Apple-compatible products like Anker's USB-C rapid chargers may contribute to the stock's potential for further growth, supporting the recommendation to buy.
NVDA	HOLD	65%	20	-76.35%	BUY 4 shares	\$709.56	Given your existing position in NVDA with 20 shares held at an average buy price of \$750.0, the current price of \$177.29 presents an opportunity to average down and potentially reduce your overall cost per share. Recent market volatility, as seen in the decline of the Dow, S&P 500, and Nasdaq, may have created a buying opportunity for growth-oriented investors like yourself. The recommendation to buy more NVDA shares is supported by the potential for long-term growth in the AI optics sector, despite current challenges in the Chinese market and short-term market fluctuations.

Methodology & Tools

- Frontend: React/Next.js for user interface and inputs
- Backend: FastAPI for API handling
- Data Sources: Alpha Vantage API provides real-time market prices and News API for recent news sentiment
- Agent Logic: Python-based rule agent to compute confidence score for each stock
- LLM integration: Groq accesses the Llama LLM via LangChain to generate explanations for the recommendations
- Storage: JSON file used as a persistent portfolio
- Security: Only accepts stock ticker symbols and number of shares
 - Does not store personal identity data or account credentials
 - API keys stored in backend .env and never exposed to frontend



Conclusion

- Agent-based AI systems can support financial decision-making based on user input
- The combination of rule-based logic and LLM reasoning provides both structured decision-making and explainability
- A full-stack implementation shows the feasibility of deploying such systems in real-world applications
- Next Steps:
 - Incorporate predictive analytics
 - Generate recommendations for users without a portfolio
 - Rely more on LLMs to create a recommendation