

Technical Report on Automated Stock Analysis Workflow

Gangadhar Singh Shiva, gshiva@sandiego.edu, Akshobhya Rao BV abv@sandiego.edu,

September 21, 2025

1 Introduction

This report documents the design, execution, and visualization of an automated stock analysis workflow. The system integrates multiple components: data collection (prices, historical trends, and news), natural language processing for sentiment classification, and recommendation analysis. A visualization module compiles results into a structured dashboard in PDF format.

2 System Architecture

The system is composed of three core modules:

1. **Server MCP LangChain (`server_mcp_langchain.py`)**: Implements a stock analysis workflow using LangChain and HuggingFace pipelines. Handles sentiment analysis, draft/critique/final summaries, and structured JSON output.
2. **Server MCP LangGraph (`server_mcp_langgraph.py`)**: A graph-based workflow engine that decomposes analysis into nodes (price fetch, news classification, sentiment, recommendations). Produces detailed step logs.
3. **Visualization (`visualize_workflow_5_code.py`)**: Processes workflow logs, generates multiple plots (timeline, state matrix, sentiment bars, word cloud, recommendation pie chart), and compiles them into a PDF dashboard.

3 Detailed Code Explanation

3.1 Stock Analysis with LangChain (`server_mcp_langchain.py`)

This module performs the stock analysis workflow:

- Imports HuggingFace pipelines for both text generation and sentiment analysis.
- Defines a function `_analyze_stock_impl()` that:

1. Fetches stock price and history using `yfinance`.
 2. Fetches top news headlines.
 3. Runs sentiment classification per headline.
 4. Uses LangChain prompts and LLMChain to create:
 - A **draft summary** of news and sentiment.
 - A **critique** that evaluates the draft.
 - A **final refined summary** integrating critique feedback.
 5. Builds a structured JSON output with all fields.
- Exposes the function as an MCP tool `analyze_stock`.

Listing 1: LangChain Draft-Critique-Final Workflow

```

1 draft_prompt = PromptTemplate(
2     input_variables=["symbol", "headlines"],
3     template="Draft a short summary for {symbol} based on these headlines:\n{headlines}"
4 )
5 draft_chain = LLMChain(llm=generator, prompt=draft_prompt, output_key="draft")
6 draft = draft_chain.run({"symbol": ticker, "headlines": headlines_text})

```

This snippet shows how a draft summary is generated for a stock using LangChain prompt templates.

3.2 Stock Analysis with LangGraph (`server_mcp_langgraph.py`)

This version implements the same workflow but structured as a directed graph:

- Each step (fetch price, fetch history, fetch news, classify sentiment, route analysis, critique, finalize) is modeled as a node.
- State is passed between nodes in a dictionary.
- Logs are written to `workflow_log.jsonl`, allowing later replay and visualization.

Listing 2: LangGraph Node Example

```

1 def classify_step(state):
2     news_items = state.get("news", [])
3     classified = []
4     for item in news_items:
5         result = sentiment_analyzer(item.get("title", ""))[0]
6         classified.append({
7             "title": item.get("title", ""),
8             "sentiment": result["label"].lower(),
9             "score": result["score"]
10        })
11     state["classified"] = classified
12     return state

```

Here, the `classify_step` node processes all news headlines, runs sentiment analysis, and appends results back to the state.

3.3 Visualization Workflow (`visualize_workflow_5_code.py`)

This script generates plots and compiles them into a PDF:

1. Loads the workflow logs from `workflow_log.jsonl`.
2. Produces visualizations:
 - **Timeline:** Execution order of nodes.
 - **State Matrix:** Heatmap showing which states are filled at each node.
 - **Sentiment Bar Chart:** Score per headline.
 - **Recommendation Pie Chart:** Distribution of buy/hold/sell decisions.
 - **Word Cloud:** Built from all news headlines.
 - **Summary Page:** Includes a table of top five headlines with sentiment.

Listing 3: Sentiment Plot Function

```
1 def plot_sentiment(ax, steps):
2     sentiments = []
3     for s in steps:
4         state = s.get("state", {})
5         if "sentiment" in state and isinstance(state["sentiment"], list):
6             for item in state["sentiment"]:
7                 sentiments.append((item.get("title", "N/A"), item.get("score", 0)))
8
9     titles, scores = zip(*sentiments)
10    ax.barh(titles, scores, color="skyblue")
11    ax.set_title("Sentiment Scores per Headline")
```

This function extracts sentiment scores from workflow logs and plots them horizontally for readability.

4 Visualization Report

The complete visualization dashboard has been compiled into a single PDF file. It contains all plots and summary tables for the five companies analyzed.

□ Workflow Summary for AAPL

Last Run: 2025-09-21T21:59:49.806132

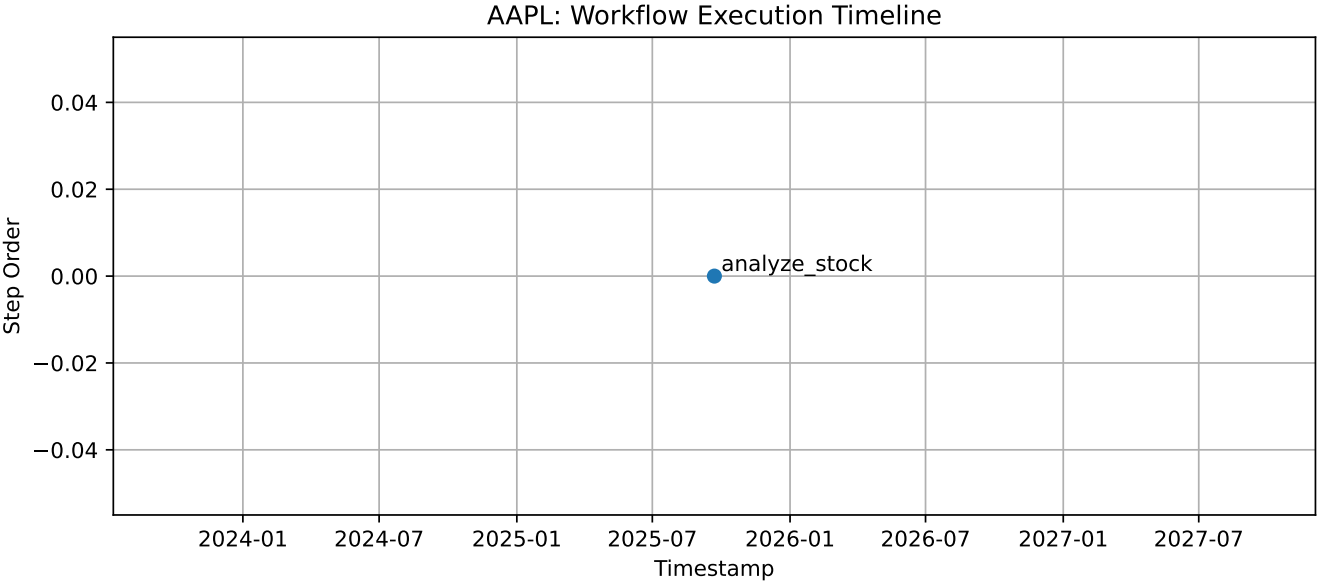
Average Sentiment Score: 0.904

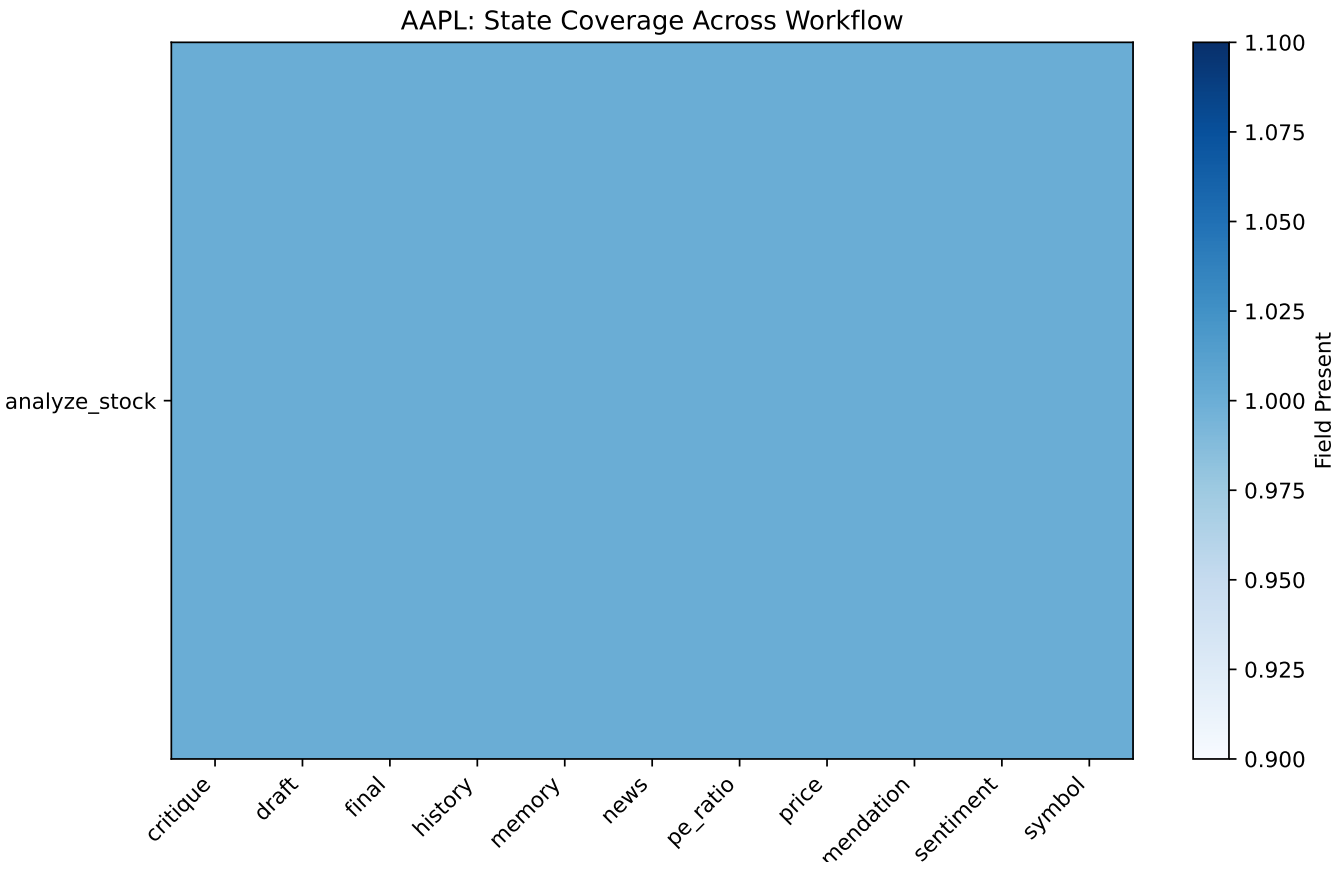
Top Recommendation: Buy - positive sentiment and upward trend

Total Steps Recorded: 1

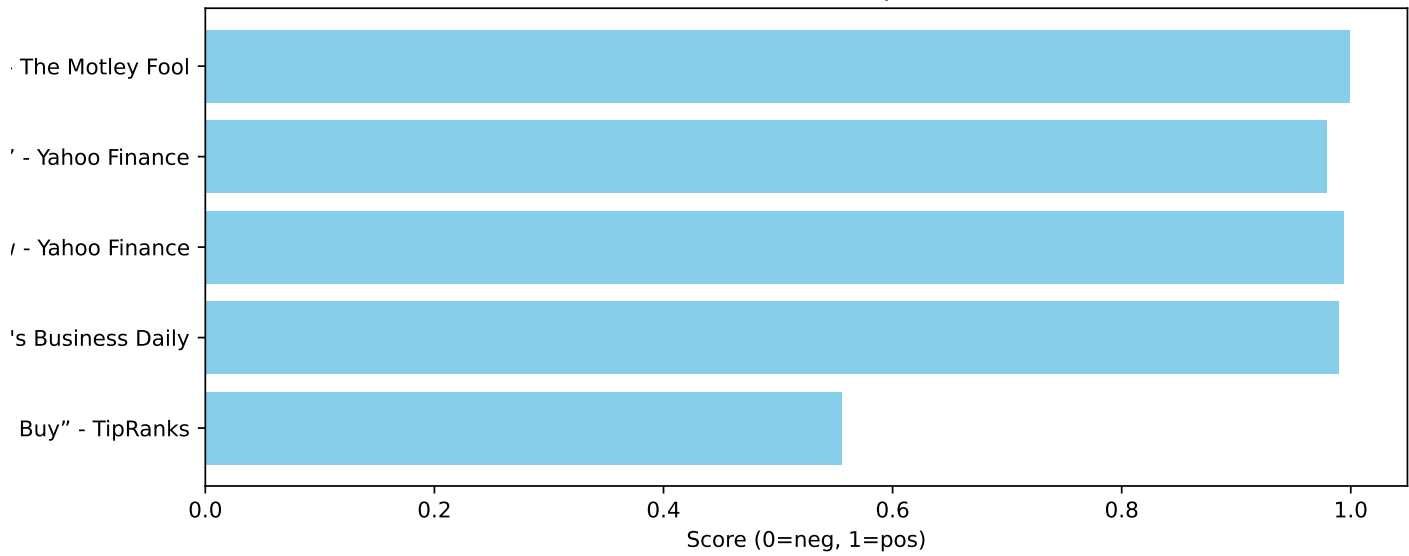
□ Top Headlines Sentiment:

Headline	Sentiment	Score
Apple Stock (AAPL) Is Up 22%, Technical Indicators	negative	0.56
Apple Stock Rises As iPhone 17 Officially Goes On	positive	0.99
Apple (AAPL) Stock Is Up, What You Need To Know -	negative	0.99
Apple Stock (AAPL) Backed by Bernstein as 'Gateway	negative	0.98
Here's Why Everyone Is Talking About Apple Stock -	negative	1.00

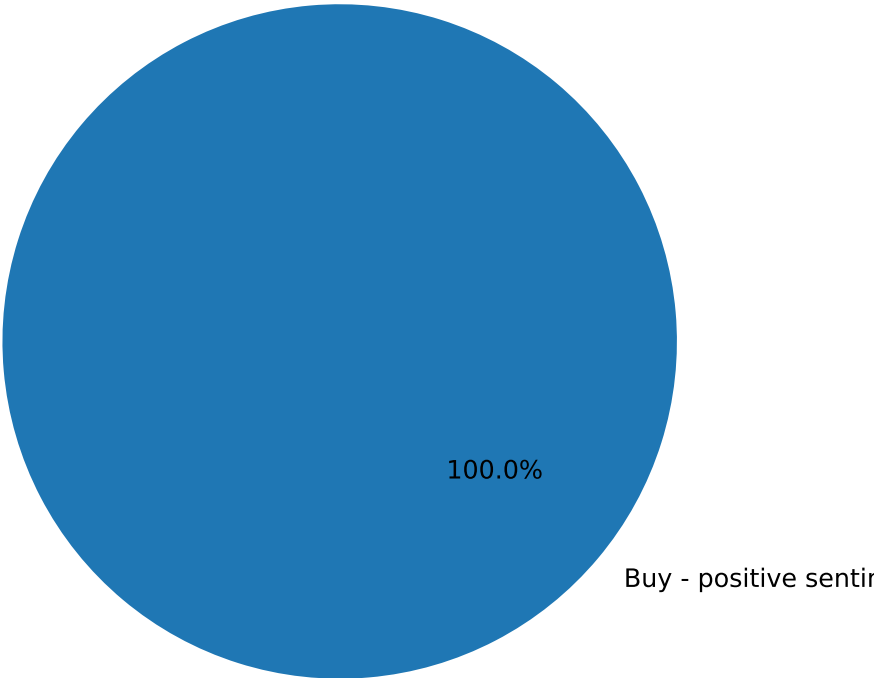




AAPL: Sentiment Scores per Headline



AAPL: Recommendation Distribution



AAPL: News Headlines Word Cloud



□ Workflow Summary for TSLA

Last Run: 2025-09-21T22:00:01.685821

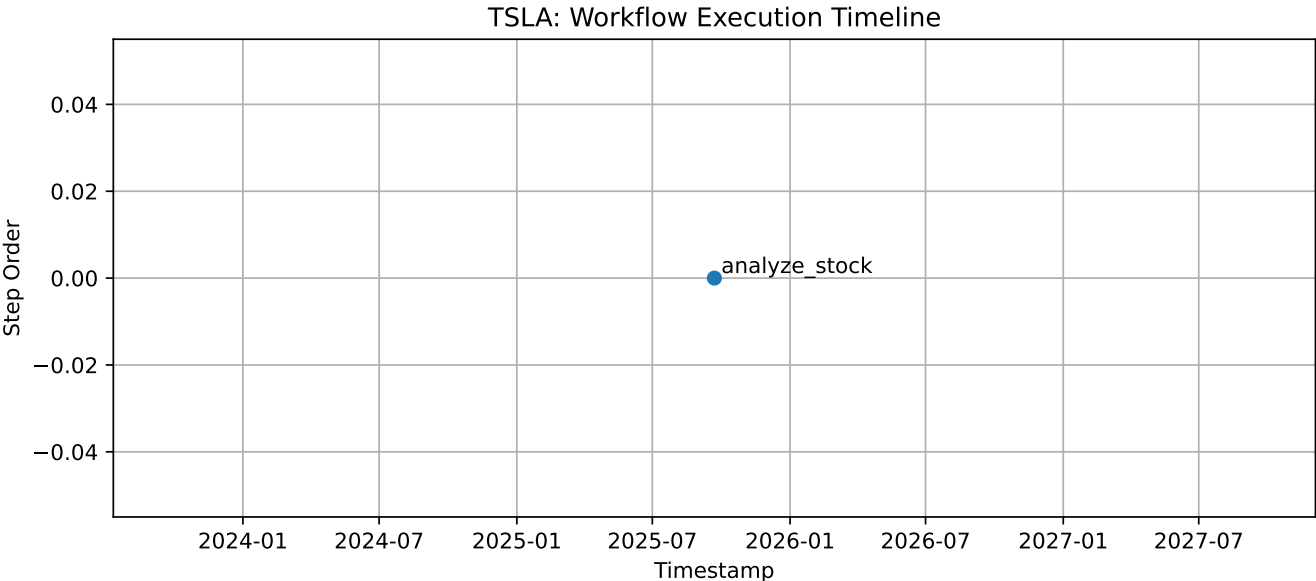
Average Sentiment Score: 0.937

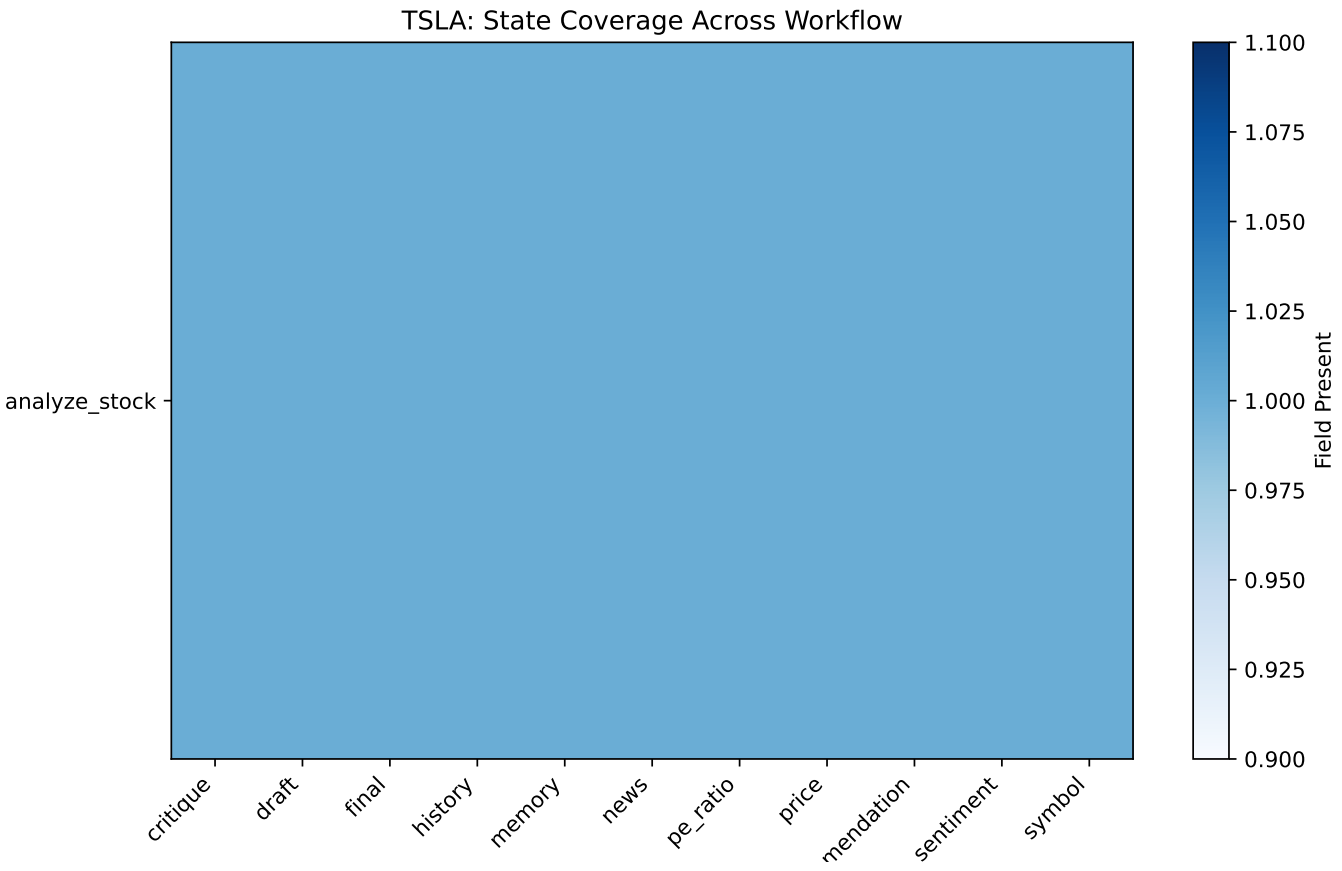
Top Recommendation: Hold - mixed or neutral signals

Total Steps Recorded: 1

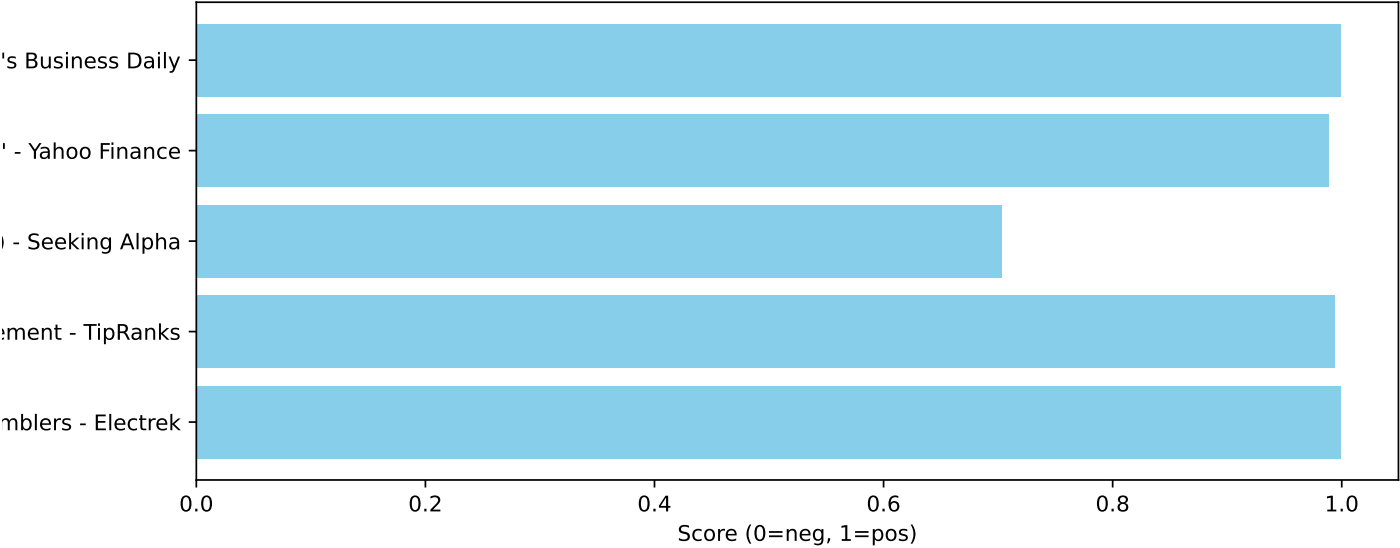
□ Top Headlines Sentiment:

Headline	Sentiment	Score
Tesla loses another Optimus robot leader, and upse	negative	1.00
AAPL, NVDA, TSLA: Wall Street Regulator Moves to E	negative	0.99
Why Tesla Stock Is A Strong Buy (Rating Upgrade) (negative	0.70
Tesla Stock To Hit \$3,000 In 2035? Analyst Says 'R	negative	0.99
Tesla Loses To Meta AI Executive Integral To '80%	negative	1.00

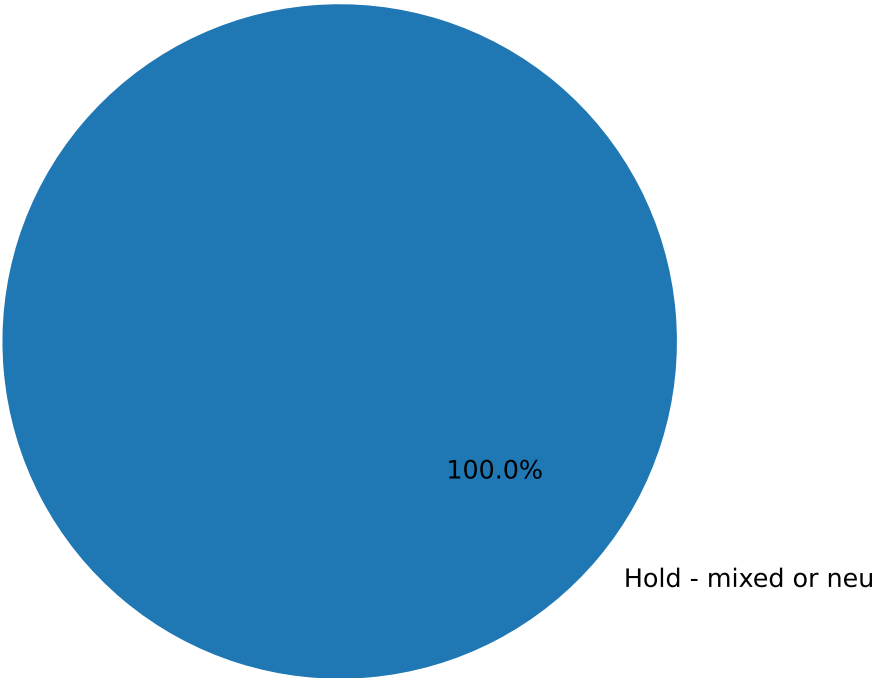




TSLA: Sentiment Scores per Headline



TSLA: Recommendation Distribution



[illegible]

□ Workflow Summary for MSFT

Last Run: 2025-09-21T22:00:05.665339

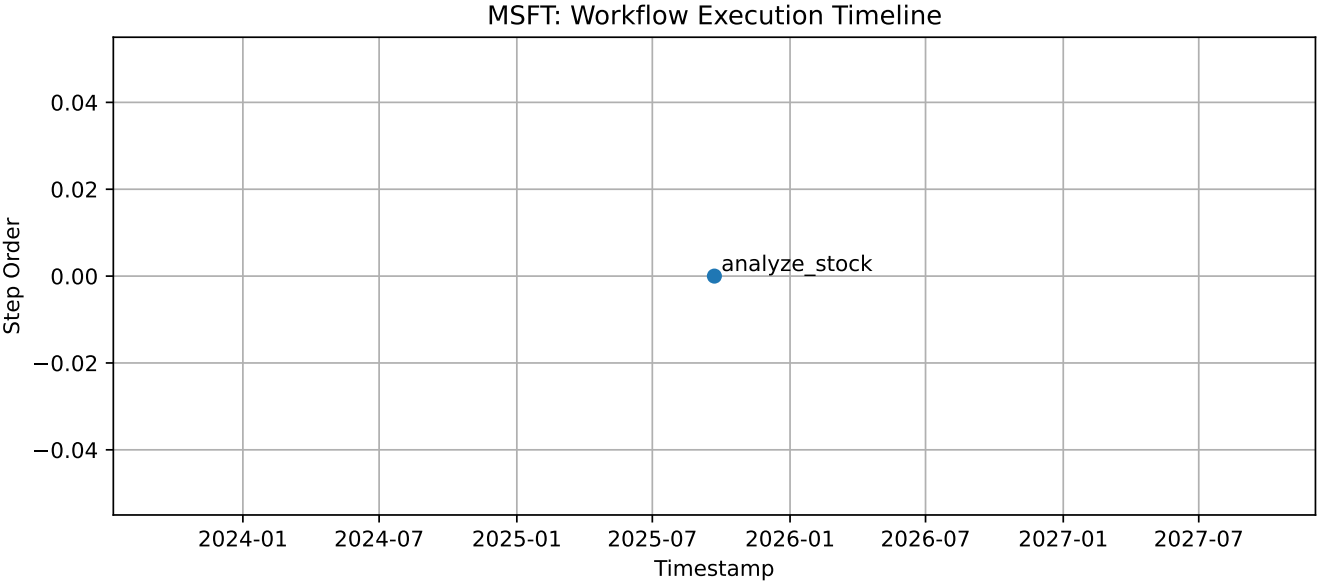
Average Sentiment Score: 0.902

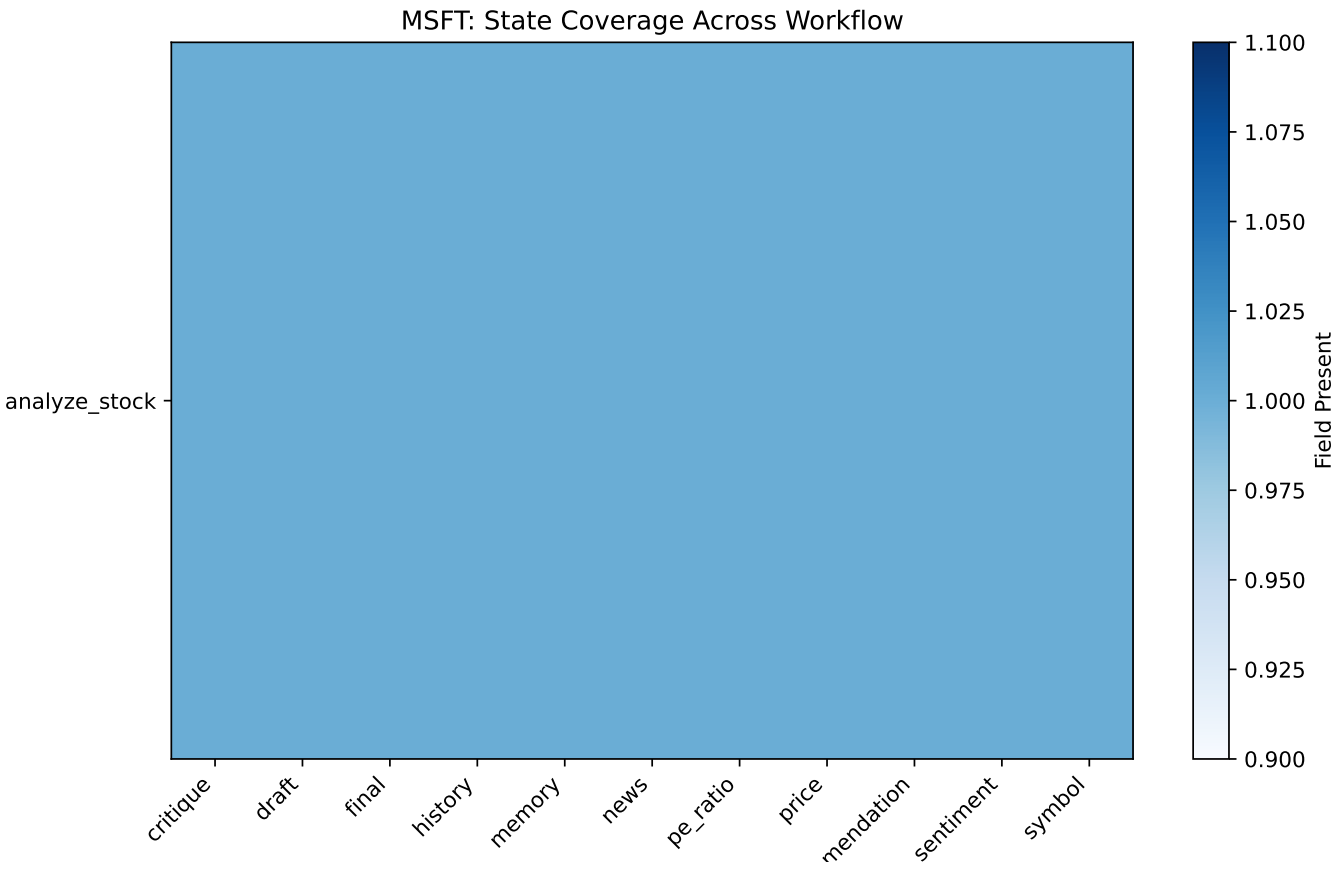
Top Recommendation: Buy - positive sentiment and upward trend

Total Steps Recorded: 1

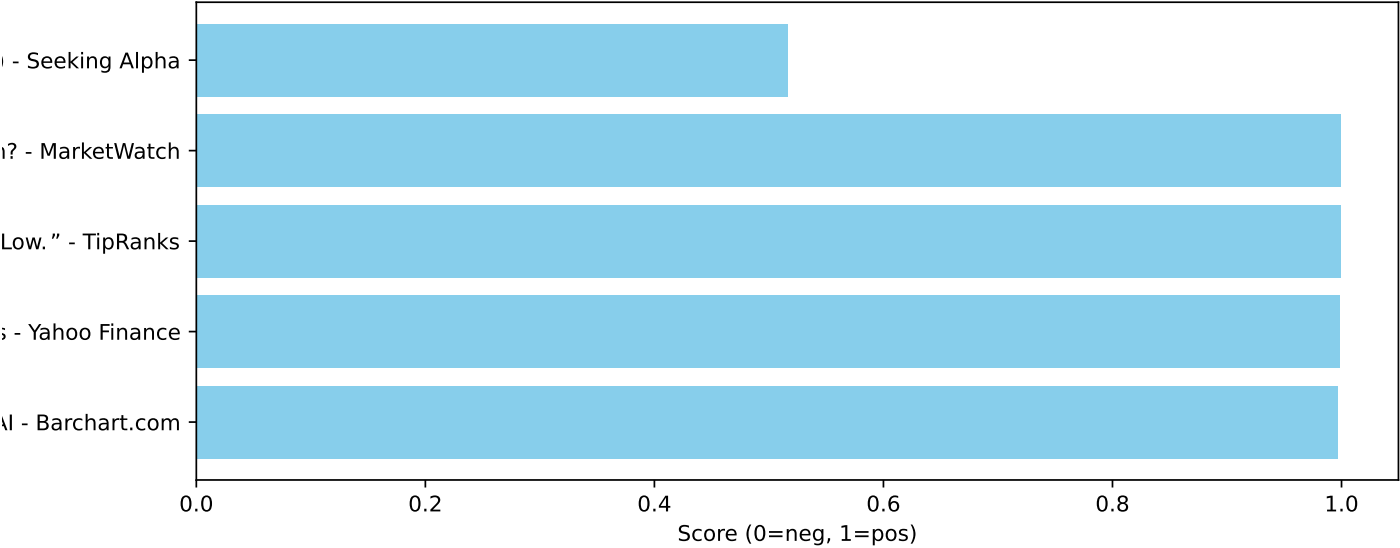
□ Top Headlines Sentiment:

Headline	Sentiment	Score
MSFT Stock Looks Set to Rejoin the \$4 Trillion Clu	negative	1.00
Microsoft Stock (MSFT) Seen Delivering Strong Tota	positive	1.00
Microsoft Stock (NASDAQ:MSFT) Gains: Morale at an	negative	1.00
Microsoft's stock has been in a rut since earnings	negative	1.00
Microsoft Could Define The Next Era Of Wealth Crea	negative	0.52

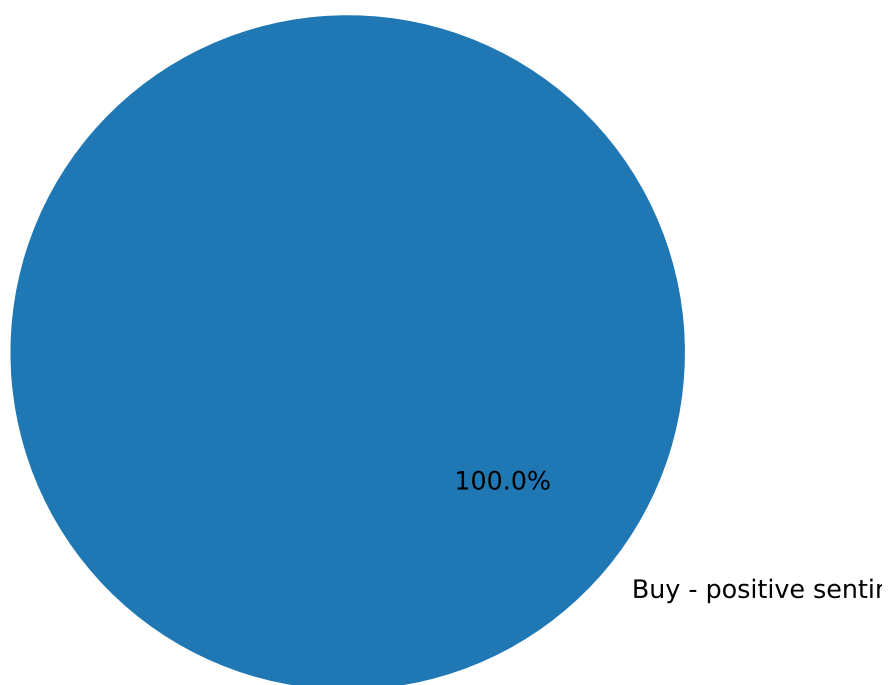




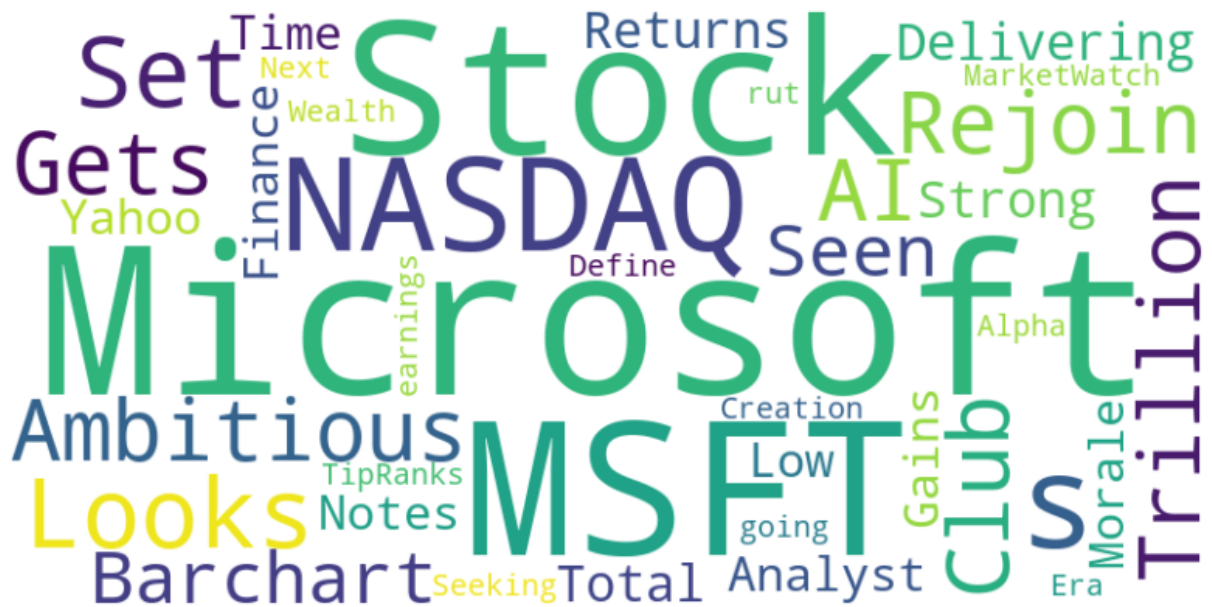
MSFT: Sentiment Scores per Headline



MSFT: Recommendation Distribution



MSFT: News Headlines Word Cloud



5 Conclusion

The automated pipeline demonstrates how stock data can be systematically collected, analyzed, and visualized in a reproducible manner. Integrating sentiment analysis with financial metrics provides an enriched perspective for decision-making.