Programming LLM: Project Proposal

MarketPulseAI: AI-Driven Market Sentiment Extractor and Sector-Based Stock Suggestion System Using LLMs

Submitted by*- Kunal SomendraSingh (UIN: 335007159)*

Problem Statement

Financial markets are highly sensitive to news cycles, macro events, and sector-specific developments. Retail investors often lack the ability to continuously scan global financial news and interpret shifting sentiment in a structured, data-backed way. While professional traders use news terminals and sentiment dashboards, these tools are often expensive, non-personalized, and lack natural language reasoning. With the rapid rise of Large Language Models (LLMs), there is an **opportunity to build an intelligent system** that processes recent financial news, extracts sector-level sentiment, and generates stock suggestions aligned with both historical context and forward-looking narratives. However, **existing LLM-based** finance tools mostly provide static summaries or generic stock lists without justifying recommendations using evidence traceable to recent events. This gap highlights a **need for a system that not only reads news but also reasons over** it to connect sentiment trends to actionable sector-based investment suggestions.

The problem **this project addresses** is the lack of accessible, explainable, and adaptive financial insight generation for sector-focused stock discovery using LLM reasoning.

Project Objectives

The primary objective of this project is to develop an LLM-centric system that ingests financial news from multiple sources, extracts

sentiment at a sector level, and generates **stock suggestions based on a sector chosen by the user**.

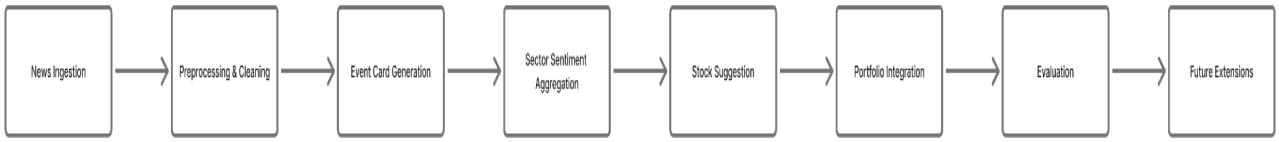
The MVP will focus on **sector-based discovery rather than full portfolio optimization**, gradually evolving into personalized portfolio alignment using CSV inputs. The system will justify each recommendation with event-based reasoning and sentiment context derived from recent market developments. A secondary objective is to ensure the system remains lightweight and cost-efficient by using free-tier APIs and optimized LLM calls. Future extension will include agentic automation for continuous monitoring and memory-based insight refinement.

Methodology

This project will follow a **pipeline-driven LLM-centric approach** optimized for actionable insight generation rather than passive summarization. First, the system retrieves recent financial news using multiple potential sources such as **NewsAPI, Yahoo Finance RSS, Financial Times RSS, and Reuters free feeds**, allowing flexibility and cost control. Instead of storing full articles, only key metadata such as headline, timestamp and a trimmed text snippet are retained to reduce processing cost.

Each news item is passed through an LLM prompt that extracts a **structured event card** containing: a two-line summary, affected sectors, relevant tickers, directional sentiment, and a **short-term or long-term bias tag**. These event cards are then **aggregated by sector** to compute a sentiment score derived from recency-weighted confidence. This approach allows the system to highlight which sectors are currently positioned for potential upside or downside

based on real-world events, not vague narratives.



For the MVP, the user selects a **sector of interest**, and the LLM uses that sector's sentiment profile to suggest a focused list of **candidate stocks or sector ETFs**. Each suggestion must include **a clear catalyst statement**, a **time-horizon classification**, and a **reason derived from event evidence**, ensuring that responses serve as **decision support**, not general commentary. The user can optionally upload a **CSV of tickers and weights** to enable early-stage portfolio alignment.

To maintain accessibility, **all components are built using free-tier APIs and efficient LLM calls**, potentially utilizing lighter open-source models for batch processing like summarization, while reserving high-quality reasoning for the suggestion phase. Evaluation will initially be **qualitative**, checking whether recommendations cite relevant events and present reasoning in a way that aids user decision-making. Additional evaluation options, such as **tracking sector ETF direction against sentiment signals**, will be logged for future analysis.

Related Work

Existing financial news platforms like **SeekingAlpha**, **Yahoo Finance Trending News**, and **Benzinga Newsfeed** provide article aggregation but **do not perform structured sentiment reasoning tied to individual investor intent**. Academic work on financial sentiment

analysis using **FinBERT and news-based trading signals** primarily evaluates accuracy on labeled datasets rather than producing user-directed, event-backed stock suggestions. Similarly, **LLM-based tools like ChatGPT plugins used for finance** often return surface-level summaries without explicit **justification linked to sector positioning or time horizon tagging**.

This project differentiates itself by **positioning LLM reasoning as a decision-support layer**, not just a summarization layer. Instead of generic sentiment scores, it generates **sector-aware catalysts tied to investor-selected themes**, making the system adaptive to **individual intent rather than static market overviews**. While some open-source projects explore **LLM + financial scraping agents**, they lack **personalized sector filtering and evidence-driven stock proposals**, which this work treats as core functionality.

The novelty lies in designing the **event card representation**, **sector aggregation logic**, and **evidence-citing suggestion output**, forming a structured reasoning loop rather than one-shot prompting. This creates a foundation for future **agentic extension**, where the system could continuously monitor sentiment drift and update stock lists autonomously, but without introducing that complexity at the MVP stage.

Timeline

Phase 1 (Weeks 1–4): Build the MVP pipeline — news ingestion, event card extraction using an LLM, and sector-level sentiment aggregation. Implement sector-based stock suggestion prompts and test accuracy of extracted catalysts and time-horizon tags.
Phase 2 (Weeks 5–8): Extend the system to accept a CSV-based portfolio input, integrate sector weighting logic, and refine the suggestion mechanism to include reasoning citations and cross-sector

comparisons.

Milestones:

1. News ingestion and preprocessing complete (Week 1)
2. Event-card LLM pipeline operational (Week 2)
3. Stock suggestion module with reasoning (Week 4)
4. Portfolio CSV integration and dashboard output (Week 5)
5. Evaluation and final documentation (Week 7)
   This phased plan keeps progress measurable while enabling iterative models and prompt refinement.
6. Final performance testing and debug issues. Finalize model functionality. (Week 8)

Challenges and Risks *(100–150 words)*

The main challenge is maintaining **accuracy and relevance** when LLMs interpret fast-changing financial narratives. News tone can conflict across sources, leading to inconsistent sentiment outputs. To mitigate this, the project will average multiple events per sector and enforce reasoning citation checks. Another risk is **limited free-tier API quotas**, which could restrict daily data pulls; this will be managed by caching five-day windows and trimming article text. Computational cost of repeated LLM calls will be reduced by batching summaries and using lightweight open-source models for preprocessing. Evaluation bias is another concern, since success is partially subjective; thus, multiple metrics—qualitative coherence, sector-direction alignment, and user usefulness scores—will be tracked. Collectively, these mitigations keep the system feasible and stable under resource constraints.

Resources Needed

Hardware / Software:

- Standard laptop or cloud CPU instance or GPU as per need.
- Python 3.11 +, libraries: requests, BeautifulSoup, pandas, openai / transformers, sqlite3 / Chroma.
- Streamlit / FastAPI for lightweight interface.

Data Requirements:

- Multiple free news feeds (NewsAPI, Yahoo Finance RSS, Reuters, Investing.com)
- Historical price data via yfinance.
- Optional FinBERT model for sentiment cross-check.

These resources ensure low-cost experimentation and easy local deployment.

Expected Deliverables

Code:
A modular Python repository with four main components — data ingestion, event card generation, sector sentiment aggregation, and stock suggestion logic.

Final Report:
Comprehensive documentation of methodology, pipeline design, evaluation metrics, and analysis of sector-based results.

Additional Artifacts:

- Streamlit dashboard displaying top sectors, suggested stocks, and reasoning citations.
- Example CSV portfolio demonstrating future integration.

- Visualization of sentiment distribution across sectors and time horizons.

These deliverables demonstrate a functional, personalized insight engine ready for future agentic automation and portfolio optimization phases.

Ideas to be Explored (Future Extensions)

- **RAG-Based Contextual Updates:** Implement a lightweight Retrieval-Augmented Generation (RAG) system to keep the LLM grounded in recent market data. This will enable the model to dynamically pull and reference the latest financial news snippets, reducing hallucinations and ensuring every recommendation is supported by timely, verifiable evidence.

- **Agentic Automation for Continuous Monitoring:** Introduce autonomous agents that periodically ingest and analyze market data to refresh sector sentiment, identify emerging trends, and update recommendations. This would transition the system from manual prompts to an always-on financial insight assistant.

- **Adaptive Learning Through Historical Feedback:** Log user actions and outcome data (e.g., sector or stock performance after recommendations) to gradually refine the model's prompting logic and weighting functions. This creates a feedback loop where system suggestions become more precise and personalized over time.