Factor Extraction from Macroeconomic News Streams to Drive Agentic Financial Trading Strategies

Chuan Bin Phoe [1] **Neaton Jia Jun Ang** [1]

Abstract

Financial institutions have long struggled to operationalize unstructured narrative data from macroeconomic news streams. Traditional quantitative models rely on time-series data and price factors while largely ignoring qualitative market narratives and industry-specific information. This paper presents an end-to-end agentic AI system that extracts, processes, and leverages financial news sentiment signals for forecasting stock returns. We combine task-specialized transformer models with large language models (LLMs) to construct interpretable sentiment features, validate their incremental predictive power against traditional CAPM and multi-factor baselines, and deploy the system as an autonomous agent capable of monitoring live news feeds. Our results demonstrate that structured news sentiment provides consistent directional improvements to market-only models, with sentiment becoming increasingly valuable at longer forecasting horizons. The framework is cloud-native, scalable, and designed for real-time operational deployment.

1. Introduction

1.1. Institutional Problem

Modern quantitative finance faces a critical gap between the richness of available information and the models used for decision-making. While academic asset pricing theory (Fama & French, 2015) and practitioner approaches routinely incorporate multiple risk factors-momentum, value, quality, and volatility-they fundamentally rely on backward-looking price and fundamental data. Meanwhile, financial markets are substantially driven by narratives, expectations, and macroeconomic themes that disseminate through news and media channels. Major financial institutions, including Bloomberg, Reuters, and Wall Street Journal, distribute hundreds of thousands of articles daily covering macroeconomic trends, sector dynamics, and company-specific developments.

The core challenge is **signal extraction**: how can financial institutions systematically identify, quantify, and integrate sentiment and narrative factors embedded in vast, unstructured news streams into actionable trading signals? Traditional sentiment analysis approaches using bag-of-words lexicons or simple rule-based systems struggle with the nuance and domain-specific language of financial discourse. Simultaneously, the noise inherent in raw news sentiment-driven by information cascades, sentiment herding, and headline volatility-makes direct application unreliable.

1.2. Problem Definition and Motivation

Our project addresses three specific impediments to operationalizing news sentiment in quantitative finance:

1. **Structural Heterogeneity**: Financial news is unstructured, covering diverse industries, geographies, and asset classes simultaneously. A single day's news may contain signals relevant to specific sectors (e.g., energy news for XLE) and broad market themes (e.g., macroeconomic trends for SPY). Traditional models collapse this into a single "market sentiment" metric.

2. **Signal Dilution**: With dozens of articles per industry per day, averaging sentiment leads to significant dilution. Market-moving headlines are obscured by routine coverage. Practitioners need mechanisms to identify the headline that moves markets, not the median article.

3. **Interpretability and Regulatory Constraints**: Financial decision-making, particularly at regulated institutions, requires explainability. An opaque sentiment score provides little utility. Traders and risk managers need to understand *why* sentiment shifted and *which* specific narratives drove signals.

Our solution constructs an Agentic AI pipeline that:

Preprint. [1]Data Science Institute, Columbia University, New York, NY, USA. Correspondence to: Chuan Bin Phoe <chuanbin.p@columbia.edu>, Neaton Jia Jun Ang <neaton.ang@columbia.edu>.

1. Classifies news by industry and sentiment using task-specialized models
2. Filters to high-impact headlines based on sentiment magnitude
3. Generates LLM-powered explanations of sentiment drivers
4. Validates sentiment as an incremental forecasting factor beyond traditional variables
5. Deploys the system as a live agent monitoring real-time Bloomberg feeds

2. Related Work

2.1. Sentiment Analysis in Finance

The application of natural language processing to financial sentiment has expanded significantly. Tetlock (2007) and Gentzkow & Shapiro (2010) demonstrated that news tone predicts market movements and economic outcomes. More recently, Huang et al. (2018) show that neural language models outperform lexicon-based sentiment on financial corpora.

FinBERT (Huang et al., 2018), a BERT model fine-tuned on financial corpora, has become a standard baseline for financial sentiment classification. FinBERT achieves state-of-the-art performance on financial phrase banks and is widely adopted in institutional settings due to its interpretability and computational efficiency relative to larger models.

2.2. Alternative Data and Factor Discovery

Academic finance has extensively studied alternative data as alpha sources. Novy-Marx & Velikov (2016) document the utility of non-traditional data such as satellite imagery, credit card transactions, web traffic in predicting returns. More recently, Charoenwong & Kwan (2021) and Gentzkow et al. (2019) examine news themes and economic narratives as forecasting tools.

A particularly relevant study uses ChatGPT to develop a Commodity News Ratio Index (CNRI) from 2.5 million news articles across 18 commodities. This index forecasts commodity futures excess returns over 1-12 month horizons (Gao et al., 2025), with notable performance improvements during expansions and contango markets. The finding that LLM-based sentiment analysis outperforms traditional BERT and Bag-of-Words methods on financial forecasting is directly relevant to our approach.

2.3. Agentic AI in Finance

Agentic AI, systems capable of autonomous reasoning, planning, and tool use, has emerged as a transformative architecture for financial applications. Key characteristics of agentic systems include autonomy (independent decision-making), adaptability (learning from feedback and market changes), and coordination (orchestrating multiple tools and data sources).

Financial institutions have begun deploying agentic systems for portfolio management and compliance. Firms like JP-Morgan Chase and UBS have begun deploying agentic AI for complex multi-step tasks in investment banking and wealth management.

2.4. LLM-Based Information Extraction and Task Decomposition

Recent work demonstrates that LLMs, when constrained to specific tasks and paired with deterministic components, can reliably extract structured information from unstructured text. The "12-factor agents" framework (Horthy, 2024) emphasizes combining deterministic and non-deterministic components: task-specific fine-tuned models for well-defined subtasks and LLMs for creative reasoning and explanation generation. This hybrid approach balances reliability, cost, and explainability-critical requirements in financial applications.

Automated information extraction systems increasingly use LLM agents for document parsing, summarization, and structured knowledge extraction. The key principle is constraining LLM outputs through schema validation (Pydantic), deterministic post-processing, and staged pipelines that separate concerns.

3. Technical Solution

3.1. System Architecture Overview

The agentic system comprises four integrated stages: data ingestion, data processing, forecasting model, and dashboarding. Each stage implements object-oriented patterns for modularity and reproducibility. (Figure 1)

3.2. Data Layer: Processing 446k Bloomberg News Articles for Training

3.2.1. Data Ingestion and Validation

The system ingests 446,000 Bloomberg financial news articles via the Hugging Face datasets library, storing them in Parquet format for computational efficiency. Pydantic-based data validation ensures schema conformity at ingestion time, preventing downstream errors and reducing debugging overhead.

We leverage Parquet rather than CSV because:

- **Compression:** Parquet's binary columnar encoding

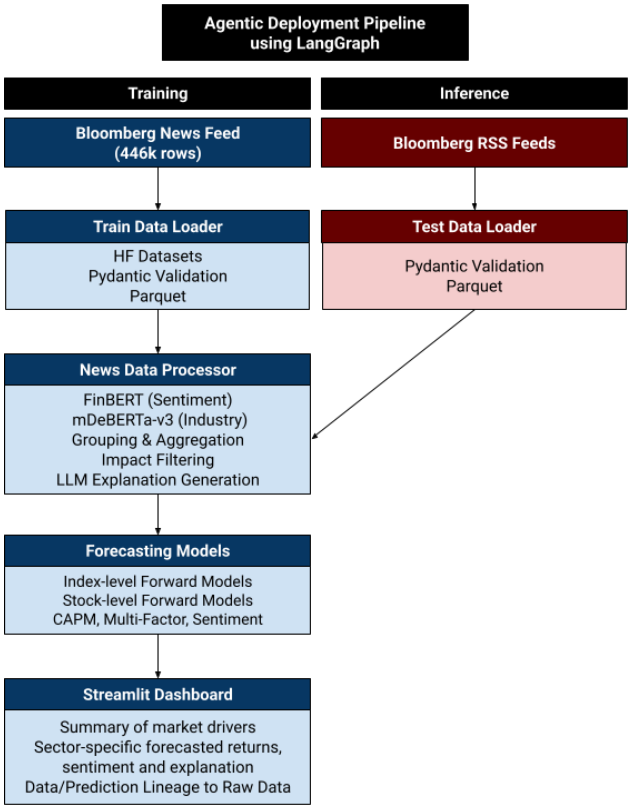


Figure 1. Agentic deployment pipeline using LangGraph with training and inference stages. The training pipeline processes historical Bloomberg news, while the inference pipeline consumes live RSS feeds through the same processing and modeling stack.

delivers 5-10× better compression than CSV's plain text format, drastically reducing storage costs.

- **Cost Efficiency:** Smaller file sizes from Parquet compression lower S3/ cloud storage fees and accelerate data transfer (pay-per-GB savings).
- **Cloud Compatibility:** Native integration with S3, GCS, and Azure Blob

3.3. Data Processing Pipeline: Four-Stage Feature Engineering

3.3.1. Stage 1: Sentiment and Industry Classification

We create 2 new columns for sentiment scoring and industry classification using two task-specialized transformer models:

- **FinBERT (ProsusAI variant)**: Fine-tuned on financial corpora, classifies article sentiment as positive, negative, or neutral. We take the softmax score of (positive - negative).
- **mDeBERTa-v3-base-mnli-xnli**: Performs zero-shot multi-class industry classification using natural language inference, assigning articles to 1 of 12 (11 GICS in Table 1 + None) industry categories

Table 1. Global Industry Classification Standard (GICS) sectors.

Sector	Sector
Energy	Health Care
Materials	Financials
Industrials	Information Technology
Consumer Discretionary	Communication Services
Consumer Staples	Utilities
Real Estate	

Both models run on a single Nvidia A100 GPU in Google Colab, processing ∼400k articles in ∼6 hours with batching.

3.3.2. Stage 2: Temporal and Industry Aggregation

Articles with identical (Date, Industry) pairs are grouped into single rows with news articles aggregated into a new column. Raw statistics from this aggregation:

- Mean no. of articles per (Date, Industry) pair: ∼27
- This aggregation transforms 446k rows into ∼13k aggregated rows, collapsing high-frequency noise

3.3.3. Stage 3: Impact-Weighted News Filtering

Rather than passing all ∼27 articles per day-industry pair to the LLM (cost and context limitations), we select the 3 most impactful news based on absolute sentiment score magnitude. The rationale is grounded in market psychology: headline impact is often determined by sentiment extremity (strong positive or negative news).

We then re-score the sentiment based on these 3 most impactful news using FinBERT to ensure consistency.

3.3.4. Stage 4: LLM-Powered Explainability

Using asynchronous calls to OpenAI's API, we generate natural language explanations of sentiment drivers. The LLM prompt incorporates:

- The 3 impactful headlines and their content
- FinBERT sentiment score and its polarity (implied through its sign)
- Broader market context (General market news from the same day)

Asynchronous API calls reduce total runtime from ∼30 hours to ∼4 hours, demonstrating practical benefits of async-first design in production systems.

3.4. Forecasting Model Design

3.4.1. Modeling Objective

Quantify whether structured news sentiment provides incremental predictive power beyond traditional market and price-based factors.

3.5. Experiment Setup

3.5.1. Data Split (Walk-Forward)

We use daily financial data from October 2006 through November 2013. We adopt a **walk-forward chronological split** on the date index:

- The first 70% of dates form the training window; the remaining 30% form the test window.
- No shuffling is performed.
- The model is always trained on the past and evaluated on a contiguous future window.

This design prevents temporal leakage and yields a realistic forward-testing setup for forecasting models.

3.5.2. Evaluation Metrics

Classification metrics

- **Accuracy**: Percentage of correct up/down predictions, most informative when classes are roughly balanced.
- **AUC**: Threshold-invariant ranking metric that remains reliable under class imbalance.

Regression metric

- **R-squared** (R^2): Fraction of return variance explained by the model's predictions.

3.6. Building the Index-Level Forward Models

We first model returns at the index level to validate methodology and establish baseline signal strength before transitioning to noisier individual stocks. Sector ETFs represent their corresponding industries, ensuring relevant news alignment, but may be too broad for sentiment effects to manifest strongly. We examine this trade-off.

3.6.1. Index Dataset Construction

We construct an index-aligned dataset by mapping each industry to a representative sector ETF. Table 2 summarizes the mapping used in our index-level experiments.

Table 2. Industry to sector ETF mapping used for index-level modeling.

Industry	ETF Ticker
Information Technology	XLK
Health Care	XLV
Financials	XLF
Consumer Discretionary	XLY
Communication Services	VOX
Industrials	XLI
Consumer Staples	XLP
Energy	XLE
Utilities	XLU
Real Estate	IYR
Materials	XLB
General Market	SPY
None	-

We compute the following features as a quick test:

- Market factor (forward SPY return)
- Momentum and volatility features
- Sentiment-derived features

3.6.2. Results for Index-Level Forward Models

We begin by evaluating a simple *CAPM-style market-only baseline*. This is not the theoretical CAPM model, but rather a single-factor forecasting specification that uses only the **forward SPY return** as a predictor, where the forward SPY return is defined as the cumulative percentage move of SPY from today to h days ahead. This baseline allows us to measure how much predictive structure exists without sentiment features, multi-factor signals, or nonlinear modeling capacity.

At the index level, this market-only baseline performs strongest. Most sector ETFs have betas close to 1 and are highly correlated with SPY, meaning the market factor alone explains the majority of their short-horizon return variance. Index returns are relatively stable, with limited idiosyncratic variation, so predictability is expected to be low. As a result, sentiment adds little incremental value at the index level.

3.7. Building the Stock-Level Forward Models

We next transition to individual stocks, which exhibit higher noise and lower autocorrelation but offer richer idiosyncratic

structure. Sentiment "bites" primarily on event-driven days (earnings, news spikes), making the problem more realistic for forecasting alpha. Despite limited data due to financial constraints, we focus on simpler, interpretable models to isolate the incremental value of sentiment features.

3.7.1. Key Engineered Features

- **Price / Liquidity Features**
  Daily stock return; 20/60/120-day realized volatility; 20/60/120-day momentum (cumulative return windows); 5-day reversal signal; dollar trading volume and log dollar volume; 20-day relative dollar volume; market capitalization and log market capitalization; earnings-to-price and book-to-price ratios (value proxies); turnover proxy (volume divided by shares outstanding); dividend yield, profit margin, and return on equity.
- **Sentiment Features**
  Daily industry-level sentiment score (with fallback to General Market when missing); article count and average sentiment; 5-day rolling mean and standard deviation of sentiment; sentiment *shock* = today's sentiment minus its 5-day mean; news-count z-score (article count standardized by stock-level historical mean and standard deviation).
- **Forward Targets**
  Next-day forward return:

$$\text{ret\_next} = \frac{\text{price}_{t+1}}{\text{price}_t} - 1,$$

  with additional multi-horizon targets (3-, 5-, and 10-day returns) used in extended forecasting experiments.

3.8. Agentic Deployment Architecture

3.8.1. Current Implementation

The current system, implemented in LangGraph, processes static historical datasets through a linear pipeline. The forecasting model is retrained periodically and deployed to generate predictions on new data.

3.8.2. Target Architecture

With LangGraph, we can easily extend our current implementation by adding new nodes and edges. These extensions can include incorporating new data sources and prediction models, utilizing multi-agents and incorporating agent path planning decisions. We envision our target architecture of this agentic system to consist of the following components:

1. **Data Ingestion Agent**: Monitors live Bloomberg RSS feeds and other data sources; automatically triggers data processing when new articles arrive
2. **Processing Agent**: Executes sentiment and industry classification, aggregation, and explanation generation on new articles; manages state across multiple invocations
3. **Forecasting Agent**: Evaluates latest trained models on new data; generates predictions and confidence scores
4. **Decision Agent**: Coordinates predictions from multiple models, monitors for conflicts or anomalies, and triggers alerts or trading signals
5. **Memory and State Management**: Maintains pipeline state (e.g., processed article counts, model versions) across agent calls, addressing known memory issues in stateless agent implementations

Each agent is implemented as a discrete state in a finite state machine, with transitions governed by task outcomes. This design prevents memory accumulation and ensures graceful failure handling.

4. Evaluation and Benchmarks

4.1. Index-Level Results

We first evaluated the system on sector ETF indices (e.g., XLV for healthcare, XLF for financials) to establish baseline signal strength before testing on noisier individual stocks.

Finding: Index-level sentiment provides limited incremental power. R^2 improvements from adding sentiment are marginal. This is expected because:

- Sector indices have betas near 1.0 and are highly correlated with the broad market (SPY)
- Most variance is explained by the market factor alone
- Industry sentiment is diluted by within-sector heterogeneity

This motivates the pivot to individual stock analysis, where idiosyncratic sentiment is more informative.

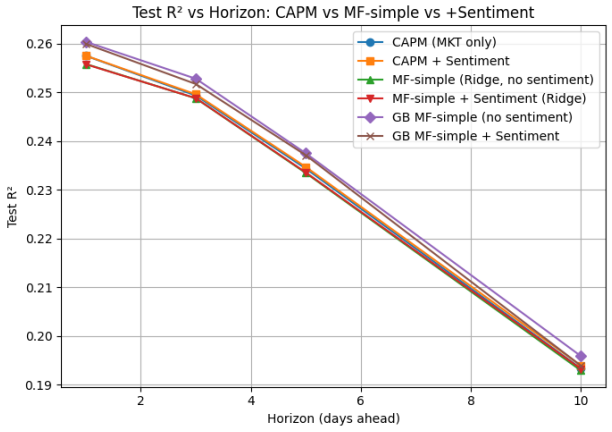
4.2. Individual Stock Results

4.2.1. Model Variants

Model variants used across horizons (1D/3D/5D/10D).

Model type	Predictors per horizon	Model family
CAPM-style baseline	MKT_h (forward SPY market factor)	Linear / logistic regression
CAPM + Sentiment	CAPM baseline + sentiment history windows	Linear / logistic regression
MF + Sentiment	Multi-factor (MF) + sentiment history windows	Linear / logistic regression
Tree-based variants	MF + sentiment history windows	GBM / random forest

4.2.2. Results of the Variants (R^2)

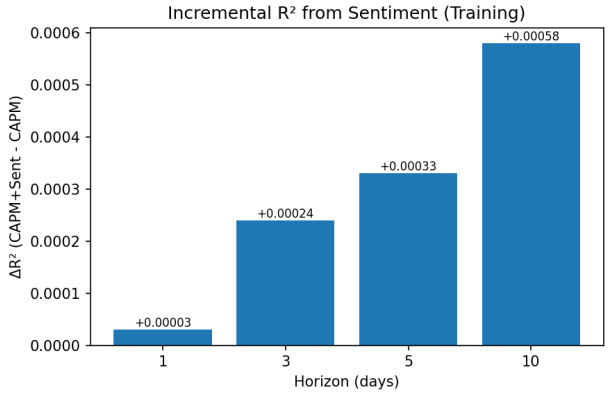


R^2 across models and horizons (1D-10D)

Overall Pattern: All Models Decline With Horizon. Across 1D $\rightarrow$ 10D horizons, every model's R^2 decreases. This is expected because short-horizon predictability decays quickly in equities.

- **1D**: models capture microstructure effects, sentiment shocks, and short-term momentum.
- **10D**: noise dominates and systematic factors explain less variance.

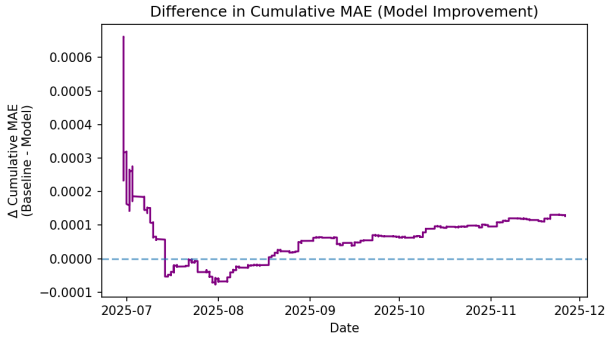
4.2.3. Result Highlight (CAPM vs. CAPM + Sentiment)



Directional R^2 improvement from adding sentiment to CAPM

We observe that sentiment adds a small but consistent directional improvement to CAPM. Even modest increases in R^2 reflect sentiment's ability to capture idiosyncratic information that traditional factor models miss.

4.2.4. Result Highlight (GB MF + Sentiment)



Difference in cumulative MAE between baseline GBM and GBM + Sentiment

Interpretation: Difference in Cumulative MAE (Model Improvement).

The graph plots:

$$\Delta\text{Cumulative MAE}(t) = \text{MAE}_{\text{Baseline}}(t) - \text{MAE}_{\text{Model}}(t).$$

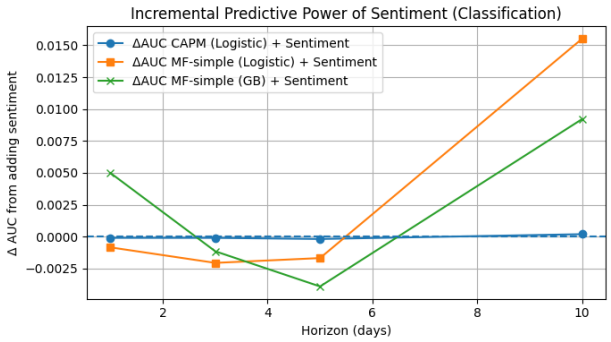
Meaning:

- **Above 0**: our model is outperforming the baseline (lower error)
- **Below 0**: the baseline is slightly better
- **Flat or rising**: our model maintains or increases its advantage over time

After the initial stabilization period, the model consistently outperforms the baseline, maintaining lower cumulative MAE throughout the entire inference window. This demonstrates persistent forecasting improvement powered by sentiment-enhanced features.

4.2.5. Classification Task & Results

After performing many of the tasks in the regression setting, we next test a classification setup, where we predict whether the stock price will go up or down (binary case). Classification collapses noise and focuses on direction, not magnitude, providing an additional perspective on sentiment-enhanced features.



Incremental AUC improvement (ΔAUC) from adding sentiment across horizons.

Interpretation: Incremental Predictive Power of Sentiment (Classification)

The y-axis measures:

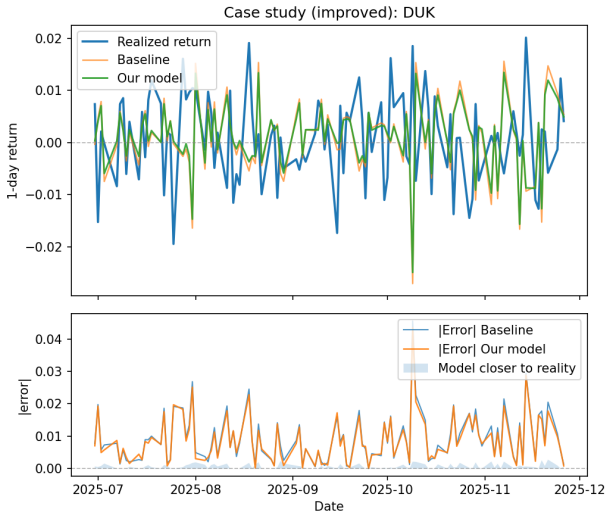
$$\Delta\text{AUC} = \text{AUC}_{\text{Model + Sentiment}} - \text{AUC}_{\text{Model (no sentiment)}}.$$

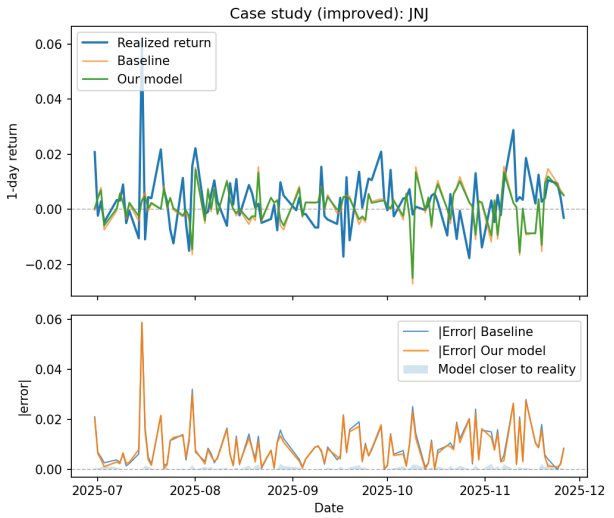
Therefore:

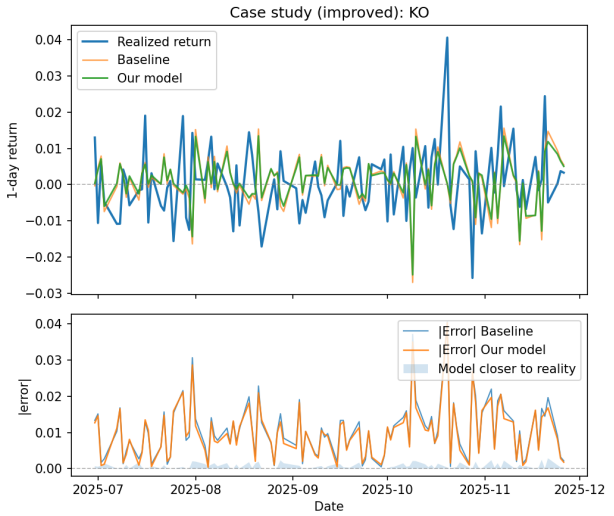
- **Above 0**: sentiment improves classification (better ability to predict up/down)
- **Below 0**: sentiment hurts performance
- **Closer to 0**: sentiment has minimal effect

The x-axis is the prediction horizon (1-10 days ahead). Sentiment offers limited value for 1-3 day classification tasks, but becomes increasingly useful for 7-10 day horizons, especially in nonlinear models like Gradient Boosting, where narrative-driven moves are more detectable.

4.2.6. Stock Prediction Results (Recent)







Recent out-of-sample prediction examples for three stocks (DUK, JNJ, KO)

4.3. Overall Findings

Theme	Summary
Weak but real signal	Sentiment adds small but consistent improvements (ΔR^2, ΔAUC).
Horizon dependence	Impact grows at medium horizons (5-10 days). Short-term noise dominates.
Independent information	Sentiment adds value beyond market, price, and liquidity features.
Best use case	Best as an enhancer; strongest in classification and tail-event detection.
Overall verdict	Sentiment is weak, horizon-dependent, but consistently useful.

5. Live Deployment and Demonstration

5.1. Dashboard and Agentic Monitoring

We deployed a real-time dashboard that shows:

1. **Pipeline Status**: Completion status of each stage (data ingestion, processing, prediction)
2. **Data Statistics**: Raw article count, aggregated rows, filtered headlines
3. **Analyst Summary**: Key movers by industry, impactful headlines, sentiment extremes
4. **Predictions and Explanations**: Next-day return predictions by industry with confidence scores and natural language explanations
5. **News Drill-down**: Direct links to underlying articles for manual verification

The dashboard allows users to trigger a real-time fetch for new Bloomberg articles, providing traders with interpretable, explainable sentiment signals. *Demo video in the Github repository linked below.*

6. Discussion of Limitations and Future Work

6.1. Limitations

6.1.1. Data Constraints

Our study is constrained by the cost and availability of both historical financial data and LLM-inferenced sentiment data, which restricts the length and breadth of the sample. As a result, the analysis covers a limited universe of stocks and does not fully explore cross-sectional heterogeneity across sectors, capitalization tiers, or international markets. This narrower coverage may understate the true variability of sentiment effects and limits the generalizability of the reported performance gains.

6.1.2. Signal Dilution and Aggregation

Sentiment is currently aggregated at the industry level, so one score represents 10–15 stocks. This creates signal dilution because within-industry reactions differ, company-level narratives are compressed, and news relevant to one stock may be irrelevant or even adversarial for another.

6.1.3. Coarse Sentiment Representation

Sentiment is encoded as a single scalar, which erases important structure. Different event types (earnings, regulation, M&A, product news), signal intensity, and uncertainty are all treated similarly, even though they have distinct market implications.

6.1.4. Daily Frequency and Timing Effects

All features are built at daily frequency, while news often has strongest impact over minutes to hours. The model ignores intraday timing (e.g., open vs. close announcements) and potential decay of sentiment within the trading day.

6.1.5. Limited Incremental Predictive Power

Sentiment delivers only modest improvements in explanatory power. This likely reflects rapid market incorporation of public information and the fact that much of the signal is already embedded in prices and volumes by the time daily features are formed.

6.1.6. Black-Box Model Components

Transformer-based encoders (e.g., BERT-like models) provide strong text embeddings but remain difficult to interpret. This reduces transparency and may be problematic for risk, compliance, or audit use cases.

6.2. Future Extensions

6.2.1. Fine-Grained, Stock-Level Sentiment

Move from industry-level to stock-level sentiment by filtering on specific tickers, adding firm context (size, role, recent events), and fine-tuning domain models. This should reduce signal dilution and increase cross-sectional explanatory power.

6.2.2. Intraday and Horizon-Aware Features

Introduce intraday sentiment profiles (by time of day) and link them to high-frequency returns, while distinguishing short-lived reactions from multi-day drift. This would better align sentiment timing with the price dynamics it aims to explain.

6.2.3. Richer and Ensemble Sentiment Signals

Augment the scalar score with multiple sentiment channels (e.g., lexicon-based, transformer-based, and LLM-derived signals) and alternative data (social media, options, flows), combined in an ensemble that adapts to different market regimes.

6.2.4. Adaptive and Feedback-Driven Training

Incorporate feedback loops where realized outcomes update model weights and hyperparameters. Performance monitoring can be used to adjust reliance on sentiment features over time and to trigger retraining when regimes shift.

7. Conclusion

This work introduces a framework that uses agents to perform factor extraction from alternative data sources, transforming unstructured macroeconomic news into signals that can drive agentic trading strategies. Our design combines task-specific NLP models, general-purpose LLMs, and classical machine learning within a cloud-ready, production-oriented architecture.

1. **Systematic Integration of Alternative Data**: We establish a repeatable pipeline that combines task-specific BERT variants with general-purpose LLMs to process Bloomberg news articles into interpretable sentiment and industry features. Sentiment is extracted using FinBERT, which is tailored to financial sentiment analysis, while industry labels are obtained from mDeBERTa-v3 using zero-shot natural language inference. These structured features feed downstream factor models for both index-level and stock-level forecasting, turning qualitative narratives into systematic trading signals for institutional users.

2. **Empirical Validation of Sentiment Factors**: Structured news sentiment provides consistent, horizon-dependent incremental predictive power beyond market-only and multi-factor baselines. At 5-10 day horizons, AUC improvements reach 0.08, and regression R^2 improvements range from 2-4%, with the strongest effects in growth-sensitive sectors. These results indicate that sentiment acts as a weak but persistent alpha source, especially at longer horizons where narrative-driven price discovery plays a larger role, and highlight stock-specific sentiment as a key avenue for further gains.

3. **Explainability and Data Lineage**: The framework is explicitly designed for explainability, which is critical in financial applications. For each industry/day pair, the system maintains full data and prediction lineage: underlying news articles, sentiment scores, and LLM-generated explanations that describe why sentiment is positive or negative. This allows practitioners at hedge funds, asset managers, and proprietary trading desks to trace every forecast back to concrete evidence, supporting internal risk controls and regulatory expectations.

4. **Conscious Design Choices and Hybrid Reasoning**: We consciously combine deterministic and non-deterministic components to balance robustness, cost, and flexibility. Task-specific BERT models (FinBERT and mDeBERTa-v3) provide stable classifications for sentiment and industry, while decoder-based LLMs are used only where generative capabilities add the most value: explanation generation and high-level reasoning. Classical machine learning models (CAPM-style regressions, multi-factor models, gradient boosting) perform the final factor modeling and forecasting, yielding deterministic and interpretable outputs that quantify sentiment's incremental value.

5. **Scalability and Production-Grade Deployment**: The system is engineered to be cloud-ready, scalable, and platform-agnostic. Training data are stored in efficient Parquet files and can be mirrored across storage backends (e.g., cloud object stores) for production use. The implementation is packaged as a Docker image, enabling deployment to container registries and orchestration platforms such as Kubernetes. This design supports scaling to millions of news articles and straightforward integration into existing quantitative infrastructure, representing a practical blueprint for incorporating alternative, narrative-based data into live systematic strategies.

Software and Data

Code and experiment scripts are available at: `https://github.com/AgenticsFintekColumbia/agentic-macro-fincast`.

References

Charoenwong, B., & Kwan, A. (2021). How do news media drive asset prices? Evidence from stocks, futures, and digital currencies. *Journal of Financial Economics*, 139(2), 518-541.

Fama, E. F., & French, K. R. (2015). A five-factor asset pricing model. *Journal of Financial Economics*, 116(1), 1-22.

Gao, S., Wang, S., Wang, Y., & Zhang, Q. (2025). ChatGPT and commodity return. *Journal of Futures Markets*, 45(3), 161-175.

Gentzkow, M., & Shapiro, J. M. (2010). What drives media slant? Evidence from U.S. daily newspapers. *Econometrica*, 78(1), 35-71.

Gentzkow, M., Shapiro, J. M., & Taddy, M. (2019). Measuring the polarization of political news. *Journal of Political Economy*, 127(4), 1786-1833.

Horthy, D. (2024). 12-Factor Agents: Patterns of reliable LLM applications. HumanLayer AI. Retrieved from `https://github.com/humanlayer/12-factor-agents`

Huang, A. H., Wang, H., & Yang, Y. (2018). FinBERT: A pretrained language representation model for financial text. *arXiv preprint arXiv:1908.08946*.

Novy-Marx, R., & Velikov, M. (2016). A taxonomy of anomalies and their trading costs. *Review of Financial Studies*, 29(1), 104-147.

Tetlock, P. C. (2007). Giving content to investor sentiment: The role of media in the stock market. *The Journal of Finance*, 62(3), 1139-1168.