Integrating Large Language Models and Reinforcement Learning for Sentiment-Driven Quantitative Trading

Wo Long, Victor Xiao

May 16, 2025

Abstract

This research develops a sentiment-driven quantitative trading system that leverages a large language model —FinGPT—for sentiment analysis, and explores a novel method for signal integration using a reinforcement learning algorithm, Twin Delayed Deep Deterministic Policy Gradient (TD3). We compare the performance of strategies that integrate sentiment and technical signals using both a conventional rule-based approach and a reinforcement learning framework. The results suggest that sentiment signals generated by FinGPT offer value when combined with traditional technical indicators, and that reinforcement learning algorithm presents a promising approach for effectively integrating heterogeneous signals in dynamic trading environments. The repository can be accessed here: https://github.com/mchlong/GR5293

1 Introduction

The increasing availability of unstructured data has opened new frontiers in quantitative finance. In particular, the integration of sentiment analysis into trading strategies has gained great interest. In contrast to traditional technical indicators, which capture patterns in historical price and volume data, sentiment signals extracted from news articles and other media offer a complementary, forward-looking perspective rooted in investor expectations and market narratives. However, effectively combining these two distinct sources of information, one backward-looking and one anticipatory, remains a significant challenge in systematic investing.

This paper explores an innovative approach to integrating sentiment information with traditional technical indicators in equity market trading. We propose a framework

driven by reinforcement learning to dynamically combine these two categories of signals and evaluate the performance of portfolios constructed based on this method. To provide a benchmark, we also compare the RL-based integration with a conventional rule-based strategy. The rule-based method constructs a composite trading signal by linearly weighting standardized sentiment and technical scores, offering an intuitive and easily adjustable framework. In contrast, the RL agent learns to synthesize and act on these signals through continuous interaction with the market environment, optimizing for long-term and risk-adjusted returns. This comparison enables us to assess the added value of reinforcement learning in the blending of heterogeneous information for portfolio decision making.

The paper is guided by three main objectives. First, his research aims to examine whether sentiment signals extracted from financial news and generated by a large language model have significant predictive power over stock returns. Second, if the sentiment signals generated can enhance the traditional trading strategies based on technical indicators. The third and most innovative contribution of this paper is to evaluate whether reinforcement learning provides a valid and effective approach for integrating sentiment and technical signals into a better-performing trading strategy.

The paper is organized as follows. Section 2 reviews the existing literature on language models, reinforcement learning, and related areas, and discusses the potential contributions of this study. Section 3 describes the data used in this study along with the pre-processing pipeline. Section 4 presents the technical framework of the trading system, detailing the specific language models and reinforcement learning algorithms used, and explaining how they interact dynamically to generate trading decisions. Section 5 outlines the detailed implementation of the trading strategies, including the trading rules and execution assumptions for both the traditional rule-based approach and the reinforcement learning–driven strategy. Section 6 and 7 discusses the results and further implication.

2 Literature Review and Contribution

The integration of Large Language Models into financial decision making has seen significant advancement over the past years, including sectors such as quantitative trading, credit scoring, and even compliance and regulation. In quantitative trading, the application of LLMs can be roughly broken down into categories based on two types of data: text-based application and figures-based application. Text data are often leveraged for sentiment analysis, which has now become more and more crucial in stock picking, tim-

ing, and prediction. Bernard et al. (2023) use a GPT large language model fine-tuned on narrative disclosures and inline XBRL tags to predict what numbers represent based on surrounding text. Lopez-Lira and Tang (2023) document the capability of LLMs to predict stock price movements using news headlines without direct financial training of the models. Glasserman et al. (2023) quantify news novelty, which refers to the changes in the distribution of news text, through an entropy measure. The entropy-based trading strategy generated statistically significant abnormal returns and alpha. Zhou and Mehra (2025) introduce an end-to-end trading system that leverages LLMs for real-time market sentiment analysis by synthesizing data from financial news and social media.

Numerical data has also gained increasing attention in quantitative investing areas. Recent studies have also brought attention to LLMs' potential application in analyzing figures in financial statements to calculate a company's future earnings direction. Kim et al. (2024) tested the predictive power of LLMs on earning's direction using a narrow information set that includes numerical information reported on two financial statements, *i.e.*, balance sheet and income statement. The study also stated that their results were to be the "lower bound" of LLMs' predictive power since only numerical information was incorporated, while textual analysis was the main strength of large language models and most aligned with LLMs' initial purpose. This marks a milestone in LLMs' application in finance and quantitative investment because it has discovered LLMs' potential to outperform fundamental analysts, who make predictions based on a comprehensive understanding of accounting statements.

While these applications highlight the growing role of LLMs in finance, Sarkar and Vafa (2024) raise concerns about look-ahead bias, where pretraining data may unintentionally includes future information, potentially contaminating predictive analyses. Through empirical tests, they demonstrate that LLMs can generate future-dependent insights, such as predicting post-2019 risks (*e.g.*, the COVID-19 pandemic) from earnings calls prior to 2020. They also show that LLMs achieve a 70–80% accuracy in predicting close election outcomes, despite such events being considered unpredictable ex-ante. Their findings emphasize the importance of carefully managing pretraining data cutoffs to ensure robust and reliable financial forecasting. A common approach in prior literature to mitigate look-ahead bias involves using a consistently anonymized format for financial texts, in which specific entities are masked—making it virtually impossible for the model to infer a firm's identity based on the text structure (Kim et al. (2024), Glasserman and Lin (2023)). This method has been empirically shown by Kim et al. (2024) to be effective in structured texts such as financial statements, where the content follows a uniform and predictable format. Kim et al. (2024) also implemented

a robustness check to assess potential information leakage in LLMs by evaluating the model's ability to guess firm-specific entities from masked texts.

Reinforcement learning has emerged as a powerful approach for data-driven decision-making in trading and portfolio allocation, allowing models to dynamically adjust strategies based on evolving market conditions. FinRL Liu et al. (2020) introduces a deep reinforcement learning framework that automates stock trading by training agents using real market environments while incorporating trading constraints such as transaction costs and risk-aversion levels. The FinRL library supports various reinforcement learning algorithms and provides standardized training, validation, and back-testing pipelines. Building on this, FinRL-Meta Liu et al. (2022) improves market simulation quality by introducing hundreds of gym-style market environments with dynamic data updates, addressing common challenges such as low signal-to-noise ratio, survivorship bias, and model overfitting. These frameworks demonstrate that RL-based trading agents can outperform traditional strategies by optimizing risk-adjusted returns and improving execution efficiency. However, while reinforcement learning has shown promise, the challenge remains in ensuring robust generalization to unseen market conditions and mitigating overfitting to historical data.

Despite the expanding body of research on large language models in finance, limited attention has been given to integrating sentiment information with traditional technical signals. This paper contributes to the literature in several ways. First, it introduces sentiment signals generated by a state-of-the-art language model—FinGPT. The long-short strategy based on these sentiment signals exhibits returns that are not explained by conventional factors such as market, size, and value. Second, this paper evaluates whether reinforcement learning algorithms offer a robust framework for translating LLM-generated sentiment signals—when combined with technical indicators—into actionable trades and portfolio allocations. Specifically, it extends the work of Zhou and Mehra (2025) by refining LLM-based signals through an RL-driven decision-making pipeline, and compares the results against traditional buy-and-hold benchmarks. Third, this paper addresses and mitigates the look-ahead bias that often arises in LLM-related research. By incorporating stricter data pre-processing techniques and performing robustness checks, similar to those used in Kim et al. (2024), the study enhances the reliability of the resulting trading strategies. Finally, this paper compares model-free reinforcement learning strategies with conventional rule-based trading approaches, evaluating their effectiveness in dynamic and volatile market environments.

3 Data

3.1 Data Description

The primary data used in this research include news articles data and stock price and volume data. We use a universe of 44 stocks selected in S&P 500 from 2018 to 2025, setting aside data from 2024 to 2025 for back-testing and evaluating the out-of-sample performance of the proposed strategies. The news data are sourced from Thomson Reuters, covering the period from 2018 to 2025, and include the 44 S&P 500 stocks selected based on the most active level of news coverage. The 44 companies in the selected universe exhibit relatively consistent news coverage throughout the 2018 to 2025 period. This universe of stocks is intentionally chosen as the objective of this research is to assess the predictive power and incremental value of sentiment information on trading performance, making it logical to concentrate on companies with consistently rich new flow. Each news observation is associated with a specific company identifier and a trading day timestamp. Multiple news articles often available for a single firm on a given trading day. We also dropped news articles released after 4 p.m. on each trading day to ensure the viability of sentiment signals for same-day trading decisions.

The historical price and volume data for the selected universe are daily-level data sourced from CRSP and Bloomberg, spanning also from 2018 to 2025. The technical data include primarily open and close prices, trading volume (measured as number of shares traded per day) obtained from CRSP, and VWAP sourced from Bloomberg.

3.2 Data Processing

The processing of news articles is a critical component of the research. Financial news tends to be lengthy and often contains redundant or irrelevant information, which might jeopardize the accuracy of sentiment classification and substantially increase the computational costs. To enable more efficient analysis, we implement a summarization step. Specifically, we employ LLaMA 3.1 8B to condense raw news articles into concise, company-level daily summaries. As mentioned earlier, multiple news articles are often available for a single firm on a given trading day. After the summarization step, all related articles for a company are condensed into a single summary per trading day. Note that the summarization on trading day t only involves news released before 4 p.m. on the same day. The step leave us with 44 (number of stocks) $\times$ 1848 (trading days from 2018 to January 2025) = 81,312 observations.

To improve scalability and computational efficiency, we adopted vLLM Kwon et al.

(2023), a framework optimized specifically for efficient inference with large language models. By enabling batch processing of queries, vLLM considerably boosts the speed and throughput of our summarization tasks, especially when handling extensive financial news datasets. We integrated vLLM with our LLaMA model to achieve a balance of efficiency and high-quality summaries. During inference, texts are broken into manageable segments to effectively handle large volumes of data while preserving the context necessary for accurate summarization. Sampling methods are thoughtfully adjusted to ensure summaries remain consistent, precise, and focused, resulting in concise and meaningful insights.

Price and volume data are adjusted for dividends, stock splits, and other corporate actions to ensure consistency across time. Price, volume and news data are synchronized at the daily level to generate coherent and meaningful signals for each trading day.

4 Technical Framework

4.1 Overview

The trading system illustrated in Figure 1 leverages cutting-edge large language models and reinforcement learning algorithms to generate trading decisions. This architecture integrates real-time sentiment analysis with historical technical indicators, forming a comprehensive strategy that incorporates both sentiment information and price-volume dynamics.

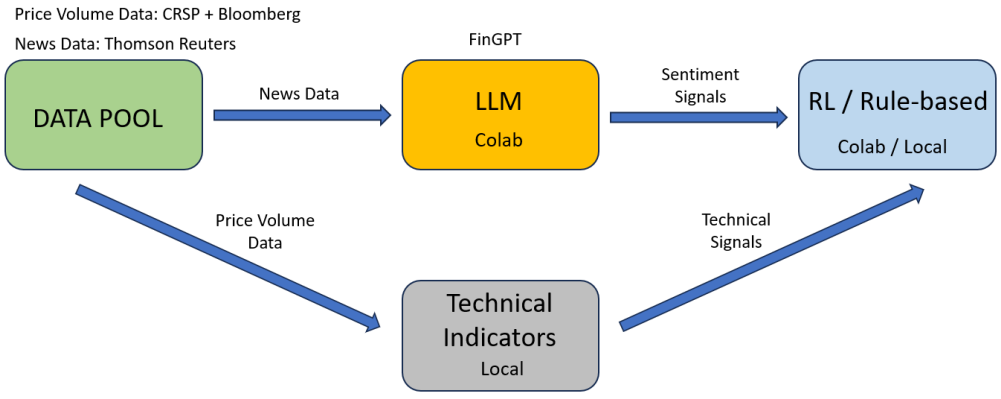


Figure 1: Trading System Illustration

First, news data are processed using a large language model (*i.e.*, FinGPT), which is trained on the Google Colab platform to generate sentiment signals. Simultaneously, a

range of technical indicators, including RSI (Relative Strength Index), VWAP (Volume-Weighted Average Price), MACD (Moving Average Convergence Divergence), *etc.*, is calculated based on historical price and volume data. These sentiment and technical signals are then used as inputs for both reinforcement learning–based and conventional rule-based strategies to generate buy and sell decisions.

4.2 Large Language Model: FinGPT

Although large language models have demonstrated strong capabilities in understanding and evaluating sentiment in textual data, they often underperform in financial applications—particularly in volatile, short-term markets—due to the substantial differences between general-purpose language and domain-specific financial text. Wu et al. (2023) introduced BloombergGPT, a powerful financial large language model trained on a vast range of financial data, which excels in tasks such as named entity recognition and sentiment analysis (Zhou and Mehra (2025)). However, the model is closed-source—implying high costs for implementation—and lacks empirical validation in trading applications. In light of these challenges, Liu et al. (2023) introduced FinGPT, an open-source, domain-specific large language model pretrained on financial texts, which also offers practitioners simple yet effective strategies for fine-tuning on domain-specific financial data. Zhou and Mehra (2025) tested the FinGPT model in live trading environments and demonstrated its strong performance in accurately capturing nuanced market language and translating it into effective trading decisions.

This paper uses a version of the FinGPT model that was pre-trained on data available up to November 2023. The company-level daily news summaries are processed by FinGPT for sentiment classification. Each summary piece is categorized into one of three sentiment classes: [Positive, Neutral, Negative], along with a confidence score (logit). The confidence score is a key input to both rule-based and RL-driven trading strategies. Sentiment signal $sentiment_{i,t}$ for company i on trading day t is generated using the summarized news of company i on trading day t.

4.3 Technical Indicators

Technical signals are computed using price and volume data. These include:

- Relative Strength Index (RSI): Measures the speed and change of price movements, indicating overbought or oversold conditions.
- Volume Weighted Average Price (VWAP): Reflects the average price weighted by volume, useful for intra-day trading analysis and price benchmarking.

- Moving Average Convergence Divergence (MACD): Highlights changes in the strength, direction, and momentum of a trend, assisting in identifying potential buy or sell signals.

- Garman-Klass volatility: Estimator for the true volatility of a financial asset over a given period. It uses high, low, open, and close prices to provide a more efficient estimate than the traditional close-to-close volatility.

All features were z-score normalized and used to construct a technical alpha signal. Technical signals $technical_{i,t}$ for company i on trading day t is generated using the price and volume data of company i from trading day $t-1$.

4.4 Reinforcement Learning Algorithm

We adopt **twin delayed deep deterministic policy gradient** (TD3), a model-free, off-policy actor–critic algorithm designed for continuous action control. TD3 extends the deep deterministic policy gradient (DDPG) to reduce the two principal failure modes of deterministic RL (*i.e.*, over-estimation bias and high target variance) through three modifications:

1. **Twin critics.** Two independent Q-networks $Q_{\theta_1}, Q_{\theta_2}$ are trained and the minimum of their target estimates is used when boot-strapping:

$$y_t = r_t + \gamma \min_{j=1,2} Q_{\theta'_j}\big(s_{t+1},\, \pi_{\phi'}(s_{t+1}) + \epsilon\big).$$

   This simple change halves the over-estimation in practice.

2. **Target-policy smoothing.** Small clipped Gaussian noise $\epsilon \sim \mathcal{N}(0, \sigma^2)$ is added to the next-state action so that the critic learns a value integrated over a region of the action space, yielding smoother gradients and improving robustness of noisy market rewards.

3. **Delayed policy updates.** The actor π_ϕ is updated less frequently, that is, every $d = 2$ critic steps, giving the critics time to converge and producing more stable policy learning.

The TD3 algorithm is suitable for portfolio allocation primarily for three reasons. First, it supports continuous action spaces, which handles naturally the structures of assets weights, whether constrained to a simplex (*i.e.*, long-only portfolio) or hyper-cube (*i.e.*, long-short strategies with leverage constraints). Second, TD3 is well-equipped to deal

with the inherent noise in financial data. Financial rewards often exhibit a low signal-to-noise ratio. TD3's conservative value estimation helps reduce bias caused by this characteristic. Third, the algorithm is sample-efficient, as it employs off-policy learning with a replay buffer, which allows the model to repeatedly learn from historical market transitions. This feature is particularly critical when training dataset is limited, with only around 1,800 trading days available.

Network architecture & hyper-parameters. Both actor and critic networks use two fully-connected layers of 256 units with ReLU activations. Key training settings:

Parameter	Value	Notes
Learning rate	10^{-4}	Adam optimiser
Discount factor γ	0.99	Daily compounding
Replay buffer size	$25\,(T_{\text{train}} - 1)$	$\approx 2.0 \times 10^5$
Batch size	$T_{\text{train}} - 1$	One episode-length mini-batch
Policy noise σ	0.2	Clipped to ± 0.5
Delay d	2	Actor update every 2 critic steps
Total updates	5.1×10^7	512 epochs on the 2018–2024 sample

4.5 Mitigating Look-ahead Bias

As previously mentioned, the FinGPT model used in this study is pre-trained on data up to November 2023. Therefore, the sentiment scores generated for the out-of-sample testing period from 2024 to 2025 are not subject to look-ahead bias. To further mitigate potential look-ahead bias during sentiment analysis, we follow the methodology in Kim et al. (2024), in which all company names, product names, and dates in news articles published before 2024 are masked prior to being processed by FinGPT.

5 Methodology and Implementation

5.1 Rule-based Trading Strategy

Standardized technical indicators are aggregated into a **composite technical alpha**. The final signal used for portfolio construction is a weighted linear combination of the technical and sentiment signals. The technical signal and sentiment signal are constructed as follows:

$$Technical\ Signal_{i,t} = \frac{Volume_{i,t-1}}{MA20_{i,t-1}} + MACD_{i,t-1} \tag{1}$$

$$Sentiment\ Signal_{i,t} = Confidence_{i,t} \cdot I_{\{1:Positive;\ -1:Negative;\ 0:Neutral\}} \tag{2}$$

Both signals are z-score normalized after construction and then integrated using the following formula:

$$Combined\ Signal_{i,t} = w_t \cdot Technical\ Signal_{i,t} + (1 - w_t) \cdot Sentiment\ Signal_{i,t} \tag{3}$$

The default weight is set to $w_t = 0.5$, which assigns equal importance to both the technical and sentiment components. This parameter can be adjusted according to market conditions or strategic preferences. The resulting combined score serves as the basis for the construction of the rule-based portfolio, where stocks are ranked daily and sorted into quintiles based on their scores. We use a universe of 44 stocks selected in S&P 500 from 2018 to 2025, setting aside data from 2024 to 2025 for back-testing and evaluating the out-of-sample performance of the proposed strategies.

Execution Assumption:

- Orders are executed at the same day's closing price (*i.e.*, trades filled at the closing auction ($\approx$ 4 p.m. ET).)
- Long-Short Strategy with initial investment of 0.
- We assume zero transaction costs for the rule-based strategy by default. As a robustness check, we also evaluate the strategy's performance under a transaction cost of 5 basis points per trade.

Caveats:

- Trades are executed on trading day t using the close price $close_{i,t}$.

Trading Rules: The $Combined\ Signal_{i,t}$ serves as the ranking basis for long-short portfolio construction. Stocks are ranked by the signal each day and assigned to 5 quintiles. The following portfolio longs the top 20% stocks and shorts the bottom 20% stocks.

5.2 RL-Driven Trading Strategy

The feature vector fed to the agent takes into account various factors, including historical returns, price-volume characteristics, volatility, historical assets weights, and most importantly, the sentiment signals generated by FinGPT in previous step. Specifically, observation s_t for trading day t concatenates seven asset-specific characteristics and the previous day portfolio weights:

- **Lagged return** r_{t-1}: close-to-close log-return of the previous trading day.
- **Momentum / overbought-oversold**:
  - 14-day relative strength index (RSI_{14}).
  - MACD signal line value (12-26-9 convention).
- **Price-volume microstructure**:
  - VWAP gap: $\dfrac{\mathrm{VWAP}_t}{\mathrm{Close}_t} - 1$.
  - Volume pressure: $\dfrac{\mathrm{Volume}_t}{\overline{\mathrm{Volume}}_{20}}$.
- **Volatility**:
  - Realized volatility ratio: $\dfrac{\mathrm{RV}_t}{\overline{\mathrm{RV}}_{20}}$, where RV is a 5-day close-to-close standard deviation.
  - Garman–Klass volatility.
- **Sentiment:** $\mathit{Sentiment\ Signal}_{i,t}$ from FinGPT. The signal is processed similarly as in rule-base strategy (*i.e.*, scaled to $[-1, 1]$ and multiplied by the confidence scores).
- **Previous portfolio weights** $\mathbf{w}_{t-1}^{\top}$ are also incorporated into the RL framework, allowing the actor to account for transaction costs and portfolio turnover in its decision-making process.

Environment design: We extend the `gymnasium` API with a custom `PortfolioEnv`. Specifically, at each trading day t :

- the **state** s_t concatenates flattened feature tensors $\mathbf{x}_t \in R^{n_{\text{stocks}} \times n_{\text{feat}}}$ and current portfolio weights $\mathbf{w}_t \in \Delta^{n_{\text{stocks}}}$:

$$s_t = \big[\mathrm{vec}(\mathbf{x}_t),\, \mathbf{w}_t\big];$$

  the $n_{\text{feat}} = 7$ features listed on Slide 2 capture return, momentum, sentiment, liquidity and volatility.
- the **action** $a_t \in R^{n_{\text{stocks}}+1}$ is a set of raw logits that are *projected* into valid long-only weights via softmax.

- the **reward function** constructed is:

$$r_t = \underbrace{\frac{V_{t+1}}{V_t} - 1}_{\text{gross portfolio return}} - \underbrace{\text{turnover}_t \, c_{\text{tcost}}}_{\substack{\text{proportional} \\ \text{transaction cost}}} - \underbrace{\text{short\_exposure}_t \, c_{\text{borrow}}}_{\text{borrow cost}},$$

Training-testing protocol: The training period of the reinforcement learning agent spans from 2018 to 2023. TD3 algorithm is fitted for 512 epochs using the 5-year in-sample data. Upon completion of training, the learned policy is frozen and evaluated on an out-of-sample period from January 2024 to January 2025. During this back-testing period, the trained actor is re-played in a separate environment with identical cost parameters to collect out-of-sample trading trajectories for performance assessment.

Execution Assumption:

- Orders are executed at the same day's closing price (*i.e.*, trades filled at the closing auction ($\approx$ 4 p.m. ET).)
- Long-only Strategy with initial investment of $1 million, a level considered small enough to avoid exerting significant market impact.
- Transaction cost is set conservatively at 10bps.

The caveats are the same as that applied in the rule-base strategy, except that the transaction cost for RL-driven strategy is set at a more conservative 10bps per trade.

6 Results

6.1 Rule-Based Long-Short Strategy Performance Result

The rule-based long-short strategy was tested in both the out-of-sample period from Jan 2024 to Jan 2025 and the full-sample period from Jan 2018 to Jan 2025. **OOS**

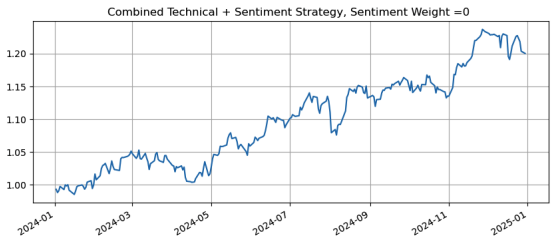
Results

The following table summarizes the key performance metrics for each sentiment weight: We observe that all the long-short strategy has robust return profile in the OOS back-test. While increasing sentiment weight tends to slightly reduce return and Sharpe ratio, it also leads to higher drawdowns and lower downside protection as reflected in the Sortino ratio.

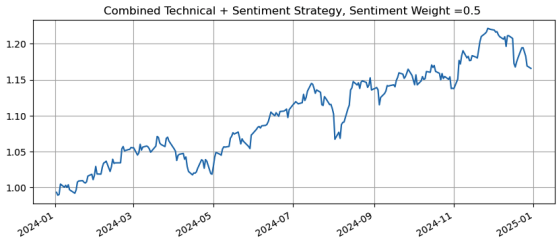
Fama-French Factor Decomposition for OOS Results

Metric	Sentiment Weight = 0	0.5	1.0
Annualized Return (%)	20.14%	16.66%	15.55%
Volatility (%)	11.78%	11.70%	11.70%
Sharpe Ratio	1.6146	1.3735	1.2916
Sortino Ratio	0.1442	0.1220	0.1092
Max Drawdown (%)	-5.63%	-6.78%	-7.66%

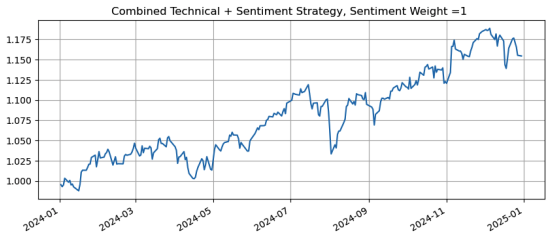
Table 1: Performance Metrics of OOS Combined Strategy with Different Sentiment Weights



(a) Sentiment Weight = 0



(b) Sentiment Weight = 0.5



(c) Sentiment Weight = 1

Figure 2: OOS Backtest NAV under Varying Sentiment Weights

To further examine the OOS return attributions, we regress the strategy return on the Fama-French 5 Factors to look at our strategy exposure. The regression result is shown below:

- None of the factor loadings are statistically significant at conventional levels(0.05) across sentiment weights, implying few meaningful linear relationship between the strategy returns and standard risk premia.
- The absence of significant exposures suggests that the OOS strategy may be capturing alpha that is unrelated to market, size, value, profitability, investment, or momentum factors.
- The R-squared values are consistently low (approximately 2–3%), indicating that the Fama-French 5-Factor + Momentum model has limited explanatory power for this strategy in the out-of-sample period.

Factor	**Sentiment = 0**	**Sentiment = 0.5**	**Sentiment = 1**
Const	0.0006	0.0004	0.0004
Market (mktrf)	-0.0196	-0.0310	-0.0206
SMB	0.0791	0.0661	0.0199
HML	0.0408	0.0636	0.0982
RMW	-0.0733	-0.0611	-0.1236
CMA	0.1166	0.1263	0.0972
UMD	0.1118	**0.1539***	**0.1380***
R-squared	0.022	0.025	0.025

Table 2: Factor Loadings from Fama-French 5-Factor + Momentum Model (OOS Strategy)

- These findings imply that the strategy's return drivers likely lie outside of conventional factor-based explanations, and may be rooted in alternative sources such as technical or sentiment-driven signals.

Since few coefficients are significant, we will later examine the full samplr return decomposition in the next part and propse several explanations for this result.

Full Sample Results

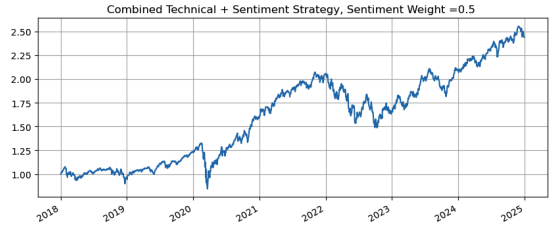
The following table summarizes the key performance metrics for each sentiment weight for the full sample period from 2018 to 2025:

Metric	**Sentiment Weight = 0**	**0.5**	**1.0**
Annualized Return (%)	13.84%	13.58%	13.25%
Volatility (%)	20.10%	20.14%	20.15%
Sharpe Ratio	0.7475	0.7350	0.7203
Sortino Ratio	0.0566	0.0561	0.0547
Max Drawdown (%)	-37.32%	-36.24%	-35.49%

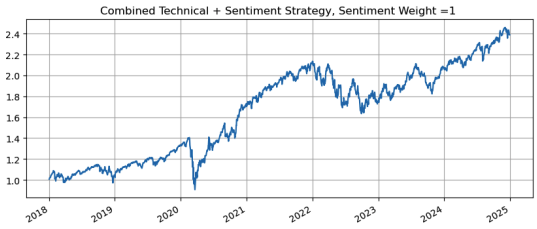
Table 3: Performance Metrics of Combined Strategy with Different Sentiment Weights (2018–2025)

We observe that the combined strategy maintains a relatively stable return profile across different sentiment weights over the full-sample period from 2018 to 2025. As sentiment weight increases, both annualized return and Sharpe ratio decrease slightly, while maximum drawdown narrows modestly. This suggests that a a higher sentiment weight may offer marginal risk reduction, but at the cost of a lower risk-adjusted return.

(a) Sentiment Weight = 0



(b) Sentiment Weight = 0.5



(c) Sentiment Weight = 1

Figure 3: Full Sample Backtest NAV under Varying Sentiment Weights

Fama-French Factor Decomposition for Full Sample Results

Similarly, we did the return attribution analysis on the full sample period and get the FF5 Factor exposure results as follows:

Factor	**Sentiment = 0**	**Sentiment = 0.5**	**Sentiment = 1**
Const	0.0006*	0.0006*	0.0006*
Market (mktrf)	**-0.1675*****	**-0.1648*****	**-0.1841*****
SMB	**0.1763*****	**0.1776*****	**0.1711*****
HML	**-0.1342*****	**-0.1524*****	**-0.1229*****
RMW	-0.0857	0.0687	0.0528
CMA	-0.0372	-0.0253	-0.0253
UMD	-0.0416	**-0.0507***	**-0.0559****
R-squared	0.033	0.034	0.038

Table 4: Fama-French 5-Factor + Momentum Regression: Full Sample Loadings Across Sentiment Weights

- The strategy exhibits a consistently significant and negative loading on the market factor (MKT), suggesting a contrarian beta profile across all sentiment specifications.
- Exposure to the size factor (SMB) is positive and highly significant throughout, indicating a persistent tilt toward small-cap names.
- The strategy also shows significantly negative exposure to the value factor (HML), implying a preference for growth stocks over value.

- The momentum factor (UMD) becomes increasingly significant as sentiment weight increases. For sentiment = 1, the strategy loads negatively and significantly on UMD, reflecting potential anti-momentum behavior when sentiment signals dominate.

- Other factors such as RMW and CMA do not exhibit statistically meaningful influence, indicating limited sensitivity to profitability or investment style.

- Overall R^2 values are low (3–4%), suggesting that traditional factor models explain only a modest portion of the strategy's returns, reinforcing its potential uniqueness or alpha beyond standard risk premia.

Potential Explanation of the OOS Return Decomposition Insignificance

- **Market concentration and regime shift**: In 2024, the U.S. equity market experienced significant concentration, with the "Magnificent 7" stocks comprising a substantial portion of the S&P 500's market capitalization. This concentration may have altered traditional factor exposures, affecting the performance of the strategy and its relationship with standard risk factors.

- **Sample size limitations**: The OOS period encompasses only one year (2024), providing a limited number of observations for regression analysis. This smaller sample size reduces statistical power, making it more challenging to detect significant relationships between the strategy returns and risk factors.

- **Structural changes in market dynamics**: The unique economic and political events of 2024 may have led to structural changes in market dynamics, rendering traditional factor models less effective in capturing the drivers of strategy returns during this period.

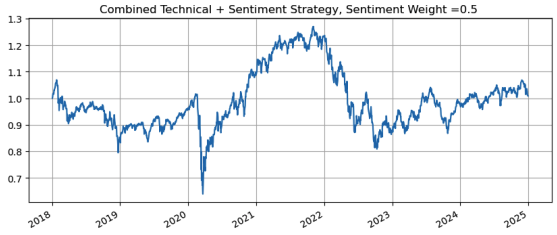
Transaction Cost Analysis

The rule-based strategy was tested under the assumption that no transaction cost incurred during the execution process. As a result, there is an significant alpha as showned in the full sample return decomposition. To test the strategy robustness, a transaction cost of 5 basis points is applied to each trade in the following backtests, which is high and conservative for the large-cap stocks in the portfolio, mitigating potential overstatement of performance results. The results are reported as follows:

Metric	Sentiment Weight = 0	0.5	1.0
Annualized Return (%)	3.66%	0.13%	1.22%
Volatility (%)	20.10%	20.14%	20.15%
Sharpe Ratio	0.2806	0.1075	0.1614
Sortino Ratio	0.0214	0.0083	0.0124
Max Drawdown (%)	-37.86%	-40.20%	-36.25%

Table 5: Full Sample (2018–2025) Performance of Combined Strategy, Transaction Cost = 5bps

(a) Sentiment Weight = 0



(b) Sentiment Weight = 0.5

(c) Sentiment Weight = 1

Figure 4: Full Sample Backtest NAV under Varying Sentiment Weights, Transaction Cost = 5bps

Impact of Transaction Costs on Strategy Performance

Comparing the two full sample performance tables, we observe a clear deterioration in annualized return once a 5bps transaction cost is applied. For example, under sentiment weight = 0, the annualized return drops from 13.84% to 3.66%, representing a more than 10% absolute reduction. Similar erosions are observed across other sentiment weights, with Sharpe ratios also significantly declining. The effect is more significant in higher turnover regimes where strategy is assigned with higher sentiment weights, and this explains the more substantial drop in return for sentiment weight = 0.5 and 1.0. Despite volatility remaining stable, the deterioration in both absolute and risk-adjusted returns underscores the importance of cost-aware strategy design in practical implementation.

6.2 RL-driven Strategy Performance Results

Figure 5 plots the net asset value (NAV) of the TD3 policy against the long-only buy-and-hold benchmark in the 2024 out-of-sample window. Table 6 summarizes the key statistics for the strategy.

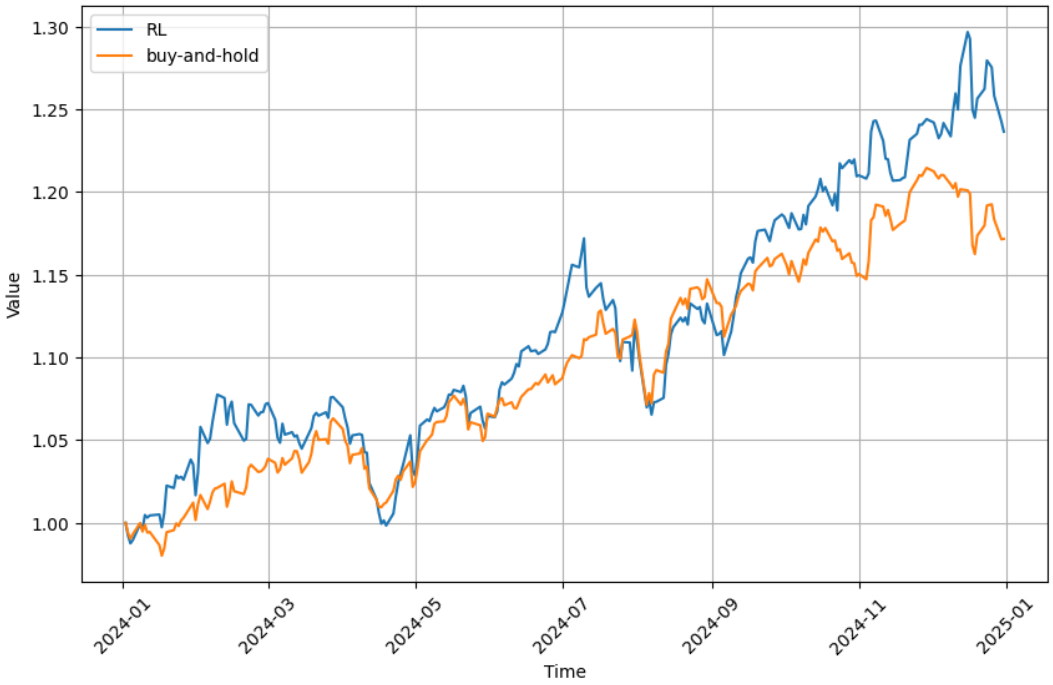


Figure 5: OOS portfolio growth: TD3 vs. Buy & Hold (Jan 2024 –Jan 2025)

Metric	TD3 Policy	Buy & Hold
Annualised return	23.65%	17.17%
Annualised volatility	13.46%	10.06%
Sharpe ratio	1.38	1.20
Sortino ratio	1.96	1.59
Portfolio turnover	52.3%	0.0%
Max drawdown	–9.09%	–5.06%

Table 6: Out-of-sample performance metrics (Jan 2024 – Jan 2025)

FF-5 Factor Decomposition:

A daily-company-level regression of the excess returns of TD3 in the Fama-French 5-factor + momentum model yields $R^2 = 0.65$ and an insignificant intercept ($p = 0.93$), indicating that the market beta explains most of the variation, while the idiosyncratic alpha is statistically non-existant once transaction costs are considered. The significant factor loadings are $\beta_{\text{mktrf}} = 0.91$, $\beta_{\text{smb}} = -0.17$ and $\beta_{\text{umd}} = -0.14$, implying a defensive,

large-cap, anti-momentum stance—consistent with the policy holding winners longer and rotating into laggards after momentum reversals.

7 Conclusion and Discussion

In this paper, we leverage domain-specific large language model to perform sentiment analysis on financial news articles and integrate the sentiment signals with traditional technical indicators into dynamic quantitative trading strategies. Our approaches include both a conventional rule-base integration and a more complex reinforcement learning-driven framework. Specifically, the comparison between the two strategies shed light on the effectiveness of the sentiment signal itself and the reinforcement learning algorithm.

The results from both the rule-based and RL-driven strategies indicate that the sentiment signal extracted from Thomson Reuters and generated by FinGPT provides some added value over strategies based solely on technical indicators. In the rule-based long-short strategy, portfolios with positive sentiment weights exhibit lower maximum drawdowns, although they tend to yield lower annual returns and Sharpe ratios compared to purely technical strategies. In the rule-base long-only strategy shown in the appendix, portfolio with positive sentiment weights exhibit better performance. In the RL-driven strategy, the portfolio exceeds the benchmark buy-and-hold portfolio in annual return, Sharpe ratio and Sortino ratio when incorporating the sentiment signals.

Furthermore, the RL agent delivers a Sharpe ratio of 1.38 despite a 52 % annual turnover even after imposing a conservatively high-level transaction cost of 10bps, demonstrating that dynamic re-allocation outweighs trading frictions.In a realistic trading environment, the transaction costs for these mega-cap stocks are expected to be much lower, which would likely result in more favorable returns for the RL-driven strategy. This paper shows that reinforcement learning is to some degree an effective and promising way to integrate traditional factors with signals from unstructured data such as LLM-generated sentiment scores.

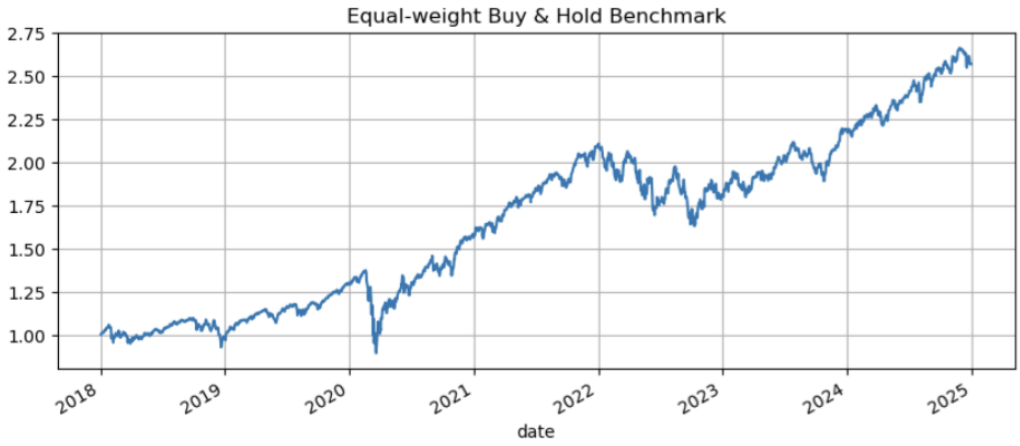
In particular, we find that both rule-base strategy with positive sentiment weight and RL-driven strategy both exhibit high turnover ratio. In the rule-based strategy, portfolio returns deteriorate significantly after accounting for transaction costs. The RL-driven strategy also delivers a 52% annual turnover ratio, albeit much lower than that in the rule-base portfolio. This indicates a key characteristic of sentiment signals - the rich and frequently updated news flow calls for more frequent portfolio re-balancing.

Limitations and Future Research First, due to limited GPU resources, our current analysis is restricted to a smaller universe of stocks. We focus on mega-cap stocks, as they tend to receive more consistent news coverage, which is an essential consideration given our limited access to diverse news sources. Expanding the stock universe to include smaller-cap stocks would require additional data sources to ensure sufficient sentiment signal quality. Second, the RL-driven strategy's performance is sensitive to the cash bias introduced by the softmax projection. Future research will explore alternative portfolio constraints, including risk budgeting and variance-penalized reward functions, to better control tracking error and improve allocation stability.

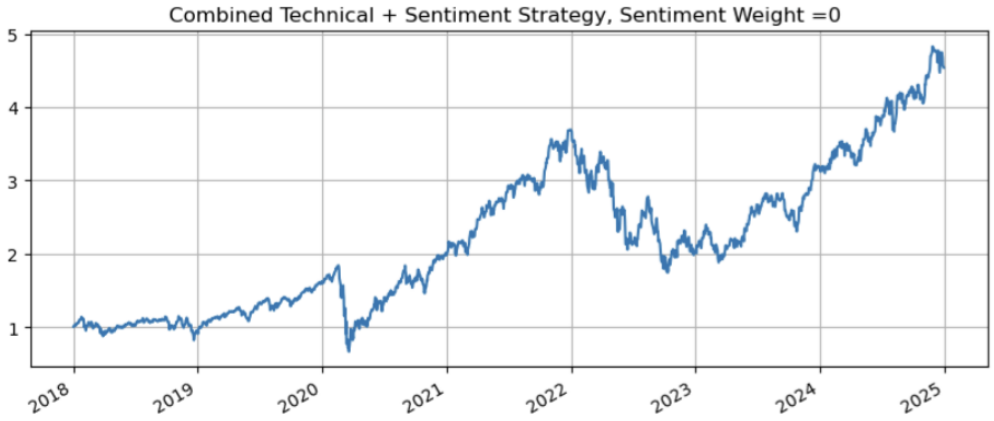
8 Appendix

Full Sample Result for Long-Only Strategy

The following table summarizes the key performance metrics for each sentiment weight for the full sample period of long-only strategy from 2018 to 2025:

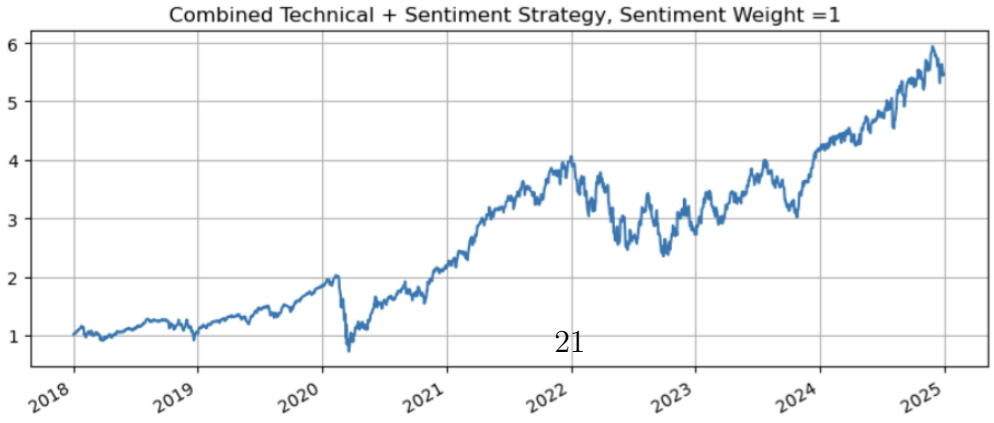


	Annualized Return	Volatility	Sharpe Ratio	Sortino Ratio	Max Drawdown
0	0.144452	0.190218	0.806725	0.05991	-0.348103



	Annualized Return	Volatility	Sharpe Ratio	Sortino Ratio	Max Drawdown
0	0.241298	0.402238	0.743416	0.05642	-0.636401

	Annualized Return	Volatility	Sharpe Ratio	Sortino Ratio	Max Drawdown
0	0.25584	0.40203	0.77188	0.058963	-0.642379



References

Bernard, D., E. Blankespoor, T. de Kok, and S. Toynbee (2023). A modular measure of business complexity. *Available at SSRN 4480309*.

Glasserman, P. and C. Lin (2023). Assessing look-ahead bias in stock return predictions generated by gpt sentiment analysis. *arXiv preprint arXiv:2309.17322*.

Glasserman, P., H. Mamaysky, and J. Qin (2023). New news is bad news. *arXiv preprint arXiv:2309.05560*.

Kim, A., M. Muhn, and V. Nikolaev (2024). Financial statement analysis with large language models. *arXiv preprint arXiv:2407.17866*.

Kwon, W., Z. Li, S. Zhuang, Y. Sheng, L. Zheng, C. H. Yu, J. E. Gonzalez, H. Zhang, and I. Stoica (2023). Efficient memory management for large language model serving with pagedattention.

Liu, X.-Y., G. Wang, H. Yang, and D. Zha (2023). Fingpt: Democratizing internet-scale data for financial large language models.

Liu, X.-Y., Z. Xia, J. Rui, J. Gao, H. Yang, M. Zhu, C. Wang, Z. Wang, and J. Guo (2022). Finrl-meta: Market environments and benchmarks for data-driven financial reinforcement learning. *Advances in Neural Information Processing Systems 35*, 1835–1849.

Liu, X.-Y., H. Yang, Q. Chen, R. Zhang, L. Yang, B. Xiao, and C. D. Wang (2020). Finrl: A deep reinforcement learning library for automated stock trading in quantitative finance. *arXiv preprint arXiv:2011.09607*.

Lopez-Lira, D. and Y. Tang (2023). Financial statement analysis with large language models. *Working Paper*. Available at SSRN 4437843.

Sarkar, S. K. and K. Vafa (2024). Lookahead bias in pretrained language models. *Available at SSRN*.

Wu, S., O. Irsoy, S. Lu, V. Dabravolski, M. Dredze, S. Gehrmann, P. Kambadur, D. Rosenberg, and G. Mann (2023). Bloomberggpt: A large language model for finance.

Zhou, Z. and R. Mehra (2025). An end-to-end llm enhanced trading system. *arXiv preprint arXiv:2502.01574*.