Language Model Guided Reinforcement Learning in Quantitative Trading

1st Adam Darmanin
University of Malta
Msida, Malta
adam.darmanin.03@um.edu.mt

2nd Vince Vella
University of Malta
Msida, Malta
vvell04@um.edu.mt

***Abstract*—Algorithmic trading requires short-term decisions aligned with long-term financial goals. While reinforcement learning (RL) has been explored for such tactical decisions, its adoption remains limited by myopic behavior and opaque policy rationale. In contrast, large language models (LLMs) have recently demonstrated strategic reasoning and multi-modal financial signal interpretation when guided by well-designed prompts.**

We propose a hybrid system where LLMs generate high-level trading strategies to guide RL agents in their actions. We evaluate (i) the rationale of LLM-generated strategies via expert review, and (ii) the Sharpe Ratio (SR) and Maximum Drawdown (MDD) of LLM-guided agents versus unguided baselines. Results show improved return and risk metrics over standard RL.

***Index Terms*—Large Language Models, Reinforcement Learning, Algorithmic Trading, Prompt Engineering, LLM Agents**

I. Introduction

Algorithmic trading systems aim to execute trades through data-driven models, sometimes leveraging machine learning (ML) to process both structured and unstructured inputs in real time. A central challenge lies in developing models that are not only high-performing but also grounded in economic reasoning [1], and capable of operating in stochastic market environments [2], [3].

In practice, institutional investment processes often involve multiple layers, including macroeconomic research, analyst input, and formal oversight, before resulting in trade execution. Capturing such structured workflows within ML systems remains a challenge. Deep learning models, while powerful, are often opaque and prone to overfitting [1], [4], limiting their trustworthiness in high-stakes financial environments. On the other hand, RL offers a framework for sequential decision-making under uncertainty, with techniques like Deep Q-Networks (DQN) and Proximal Policy Optimization (PPO) showing promise in trading tasks [5], [6], [7]. However, safety concerns, reward misalignment, and limited interpretability in the policy's decisions still hinder adoption [8], [9], [10].

LLMs offer a promising alternative, these models are capable of understanding structured and unstructured data, and are increasingly adopted in financial applications such as news-aware decisions [11], [12], [13]. However, LLMs also present challenges, including prompt fragility, numerical inaccuracies, and confabulations [14], [15]. Prompt engineering techniques have emerged as a means of mitigating these issues [16], [17].

This paper introduces a hybrid architecture that combines the long-term strategic reasoning capabilities of LLMs with the execution strengths of RL agents. The system is composed of three agents:

- **Strategist Agent:** LLM that generates high-level trading policies over a fixed horizon using multi-modal market signals.
- **Analyst Agent:** LLM that structures financial news into directional signals utilized by the Strategist Agent.
- **RL Agent:** Executes short-term local trading actions in real time, adapting to immediate market conditions.

While purely functional, this architecture reflects a separation of roles, analogous to workflows in some institutional settings.

The contributions of this work are twofold. First, we introduce a method for generating expert-aligned trading strategies using LLMs, leveraging a structured prompt framework and a multi-modal financial dataset. Second, we propose a hybrid LLM+RL architecture in which the LLM provides guidance by augmenting the observation space of an RL agent. This design enables the agent to adapt its behavior to new market conditions without retraining, reflecting a modular and more trustworthy alternative to traditional RL in financial environments.

II. Aim and Objectives

This research is structured around the following two objectives:

- **Objective 1: LLM Trading Strategy Generation**: Develop an approach for utilizing LLMs to generate trading strategies that are coherent and economically grounded, as evaluated by expert reviewers.
- **Objective 2: LLM-Guided RL**: Evaluate whether a hybrid LLM+RL architecture, improves the agent's performance across diverse equities and market regimes. Specifically, test whether this guidance increases SR and reduces MDD relative to unguided RL baselines, without requiring retraining or fine-tuning.

Preprint under review for FLLM 2025.

III. Background

Algorithmic trading systems have become integral to modern financial markets, with over 60% of U.S. equity volume attributed to automated systems, particularly high-frequency trading (HFT) [18]. This growth is driven by advances in electronic infrastructure, compute power, and the proliferation of large high-resolution financial data [19].

Long-horizon strategies commonly rely on econometric models such as the Fama–French Five-Factor Model, which captures systematic return drivers [20] in a portfolio. More recently, ML methods have introduced flexible, nonlinear models into portfolio construction [1]. Despite their expressiveness, these models remain prone to overfitting, spurious correlations, and structural breaks in financial time series [21], [4].

Conversely, short-term strategies exploit market inefficiencies through momentum, mean-reversion, or statistical arbitrage. Time-series momentum has been shown to persist across assets [22], while mean-reversion is often modeled via discrete Ornstein–Uhlenbeck processes [23].

RL has emerged as a compelling framework for sequential decisions in trading, as it casts decision-making as a Markov Decision Process (MDP) [24]. Value-based methods such as DQNs have shown success in simulated environments [7], [6], though challenges remain in sparse rewards and long-horizon credit assignment [25]. Enhancements such as Double DQNs (DDQN), Dueling Networks, and distributional critics address stability and Q-value estimation bias [26], [27]. Actor–critic methods like PPO and Soft Actor–Critic (SAC) further support learning in continuous action spaces [2].

Despite progress with RL, in recent years LLMs are increasingly being explored for financial tasks, including sentiment analysis, alpha extraction, and general market reasoning [28], [16], [14]. Whilst LLMs have demonstrated capability in such activities, they are sensitive to prompt design, and may generate plausible but invalid outputs [15]. Approaches to mitigate this include structured prompts, memory-augmented techniques, and human-in-the-loop (HITL) [29].

IV. Literature Review

The study [6] introduced the Trading Deep Q-Network (TDQN) for stock trading, incorporating the DDQN algorithm to mitigate overestimation bias and stabilize learning in dynamic market environments. A key aspect of their RL approach was the discretization of actions and the enforcement of capital constraints, which helped prevent infeasible or overleveraged strategies. The authors emphasized the sensitivity of the learned policy to the discount factor γ, noting that a low γ induces myopic, short-term behavior, whereas a high γ biases the agent toward long-horizon strategies that often resemble passive buy-and-hold policies.

The FinRL framework [2], [30], [31] introduced benchmark environments and unified APIs for financial RL research that features realistic data simulation, modular design, and reproducible backtesting. It includes a wide array of backtests using standard RL algorithms and focuses on two primary objectives in algorithmic trading: maximizing return (measured by cumulative return and SR) and minimizing risk (measured by MDD and return variance). The framework supports experiments in single-stock trading, multi-stock trading, and portfolio allocation.

Arulkumaran et al. [25] identified key limitations in deep reinforcement learning (DRL), including Bellman backup instability and credit assignment failures under partial observability - a central theme of their work. The authors recommend hierarchical reinforcement learning or recurrent extensions to address the lack of long-range temporal dependencies.

For LLMs in finance, Lopez-Lira et al. [32] showed that ChatGPT outperforms traditional sentiment lexicons on forward-looking financial news, but lacks temporal awareness and numerical reasoning capabilities. Additional limitations include the model's limited adaptability and its fixed knowledge cut-off date.

The FINMEM framework [16] combines structured memory with LLM-based decision modules. FINMEM's layered memory integrates recent news, financial reports, and long-term statements to inform trade recommendations, leveraging retrieval-augmented generation (RAG). Their architecture stores experiences in a vector database which is retrieved, these experiences are ranked using a decay mechanism that emulates a human's memory decay.

Prompting practices have been extensively surveyed by [15], who categorize strategies into instruction-based, example-based, reasoning-based, and critique-based families. The study highlights self-refinement and constraint enforcement as key mechanisms for improving robustness and factuality. Notably, it demonstrates that minor variations in prompt wording can systematically influence model behavior.

Huang et al. [33] showed that Chain-of-Thought (CoT) prompting significantly enhances LLM reasoning. Self-improvement frameworks such as STaR iteratively refine rationale quality, while techniques like problem decomposition and instruction tuning help LLMs solve complex tasks. The study also notes that, without structured prompting, LLMs struggle with planning problems that humans solve effortlessly.

V. Materials & Methods

This section outlines the methodology developed to evaluate the integration of LLMs into DRL agents. The proposed hybrid framework mirrors the top-down decision-making structures common in financial institutions, where investment mandates originate from the Chief Investment Officer's (CIO's) strategic planning and flow down to execution at the trading desk level.

Two experiments were conducted to address the research objectives. All LLM strategies were validated through historical backtesting and expert review prior to their deployment within the RL agents.

Benchmark Environment

For our benchmark, we utilized the trading system introduced by Theate et al. [6]. Their implementation applies a DDQN algorithm to single-asset trading tasks across a

curated universe of 30 equities and ETFs, spanning multiple geographies and sectors.

The original benchmark includes a clearly defined environment, consistent state and reward functions, and extensive empirical results. Their codebase is publicly available in GitHub.

We replicate the core experimental settings of Theate et al. [6], including the asset universe, data preprocessing, and evaluation metrics. Our reproduction yielded comparable SR and MDD statistics, confirming alignment with the original study, see 8.

We note minor discrepancies that affect financial interpretability: their cumulative returns are arithmetically summed rather than geometrically compounded, and their SR assume a zero risk-free rate. For consistency, we preserve their conventions throughout our experiments.

A. Experiment 1: LLM Trading Strategy Generation

This experiment addresses Objective 1 by introducing two agents: the **Strategist Agent** and the **Analyst Agent**. The Strategist Agent generates global trading policies using a financial dataset. The Analyst Agent processes news and distills it into signals to inform the Strategist Agent. This experiment serves as the foundation for Experiment 2.

A strategy defines a directional action ($\text{dir}(\pi^g)$, where 1 = `LONG` and 0 = `SHORT`) and an associated confidence score (μ_{conf}). Each strategy is accompanied by an explanation and a weighted set of financial features. Strategies are generated on a monthly basis using time-aligned, multi-modal financial input data.

1) Data and Feature Engineering.: To support strategy generation, the LLM agents consumed a multi-modal dataset spanning 2012–2020, consistent with the benchmark dataset established by [6]. The dataset includes traditional Open, High, Low, Close, and Volume (OHLCV) price data and we augmented it with four additional categories of financial signals: market data, fundamentals, technical analytics, and alternative data. These collectively define the LLM's information context:

$$\mathcal{SIG}_{\text{uni}} = \{\mathcal{S}_{\text{mk}}, \mathcal{S}_{\text{fund}}, \mathcal{S}_{\text{an}}, \mathcal{S}_{\text{alt}}\} \tag{1}$$

Market data ($\mathcal{S}_{\text{mk}}$) were sourced from Interactive Brokers[1] and iVolatility[2], including OHLCV time series, SPX and NDX index returns, the VIX index, and Options implied volatility (IV). Fundamental data ($\mathcal{S}_{\text{fund}}$), comprising firm-level financial ratios and macroeconomic indicators (e.g., GDP, PMI, interest rates), were retrieved via SEC-API[3] and the FRED API[4]. Analytics features ($\mathcal{S}_{\text{an}}$) were computed using TA-Lib[5], applying rolling windows to extract trend and momentum indicators. Alternative data ($\mathcal{S}_{\text{alt}}$), consisting of news headlines, were collected from Alpaca[6] and processed into explanatory factors using few-shot LLM prompting to $\mathcal{A}_{\text{analyst}}$, following an LLM-Factor framework [14]. The data was aligned by timestamp and back-filled to the timestamps where the effect was observed.

The full dataset is describe in (8).

[1]https://www.interactivebrokers.com/api

[2]https://www.ivolatility.com/data-cloud-api/

[3]https://sec-api.io/

[4]https://fred.stlouisfed.org/docs/api/fred/

[5]https://ta-lib.org/

[6]https://alpaca.markets/docs/api-documentation/

B. LLM Model and Prompting

We used OpenAI's GPT-4o Mini for its strong performance in financial reasoning and cost-efficiency [32], [16], [33]. The model supports a 128k token context window with a 16k maximum prompt size, enabling the use of detailed prompts with embedded context memory, reasoning chains, and structured reflection.

1) Prompt Engineering Methodology: We began with a minimal baseline prompt that utilized only raw OHLCV data alongside standard technical indicators, including the Simple Moving Averages (SMA) over 20, 50, 100, and 200 periods, the Relative Strength Index (RSI), and the Moving Average Convergence Divergence (MACD). This configuration reflects common trading heuristics used in both algorithmic and retail contexts [7], [34], [35].

a) Prompt Tuning with the Writer–Judge Loop: Prompt refinement followed a structured loop visualized in (1), and inspired by [36]. The process began with a broad *writer-trainer prompt* that included the full set of features from the dataset. This initial prompt was progressively refined by reducing the feature space using expert-labeled examples, obtained through both human-in-the-loop (HITL) feedback and a heuristic algorithm designed to approximate expert returns, see (1). Feature importance was subsequently ranked on a Likert scale from 1 to 3 and integrated into the *writer-generator prompt*.

The prompts generated the *writer-generator prompt* π_t were evaluated via backtests (with SR V^{π_t}), judged, and updated using a Bayesian update approach to minimize regret:

$$\mathcal{R}(T) = \mathbb{E}\left[\sum_{t=1}^{T} (V^* - V^{\pi_t}) \mid \mathcal{H}_t\right] \tag{2}$$

where V^* is the SR of the optimal strategy (initialized at 0.10), V^{π_t} is the SR of the current policy, and $\mathcal{H}_t$ denotes the memory buffer containing all previously evaluated prompts.

This final tuned prompt, generalized for all equities and regimes was then subject to 3 additional improvements.

b) Prompt Improvement 1 – In-Context Memory (ICM): Inspired by [16], we introduced a structured memory buffer that stores the last global strategy prior to time T, denoted as $\{\pi^g_{T-1}\}$. Each prior strategy π^g_{T-1} is represented by its directional action, weighted features, and rationale. Within the prompt, these prior strategies are recalled and their outcomes analyzed in the LLM's context, enabling the current strategy π^g_T to be conditioned on past decisions. This reflection mechanism mitigates the risk of the model persisting in suboptimal actions by revealing whether the last strategy led to poor alignment with expected outcomes.

By referencing historical decisions, the system can avoid myopic use of features that previously underperformed. The system learns to generalize from past success and failure

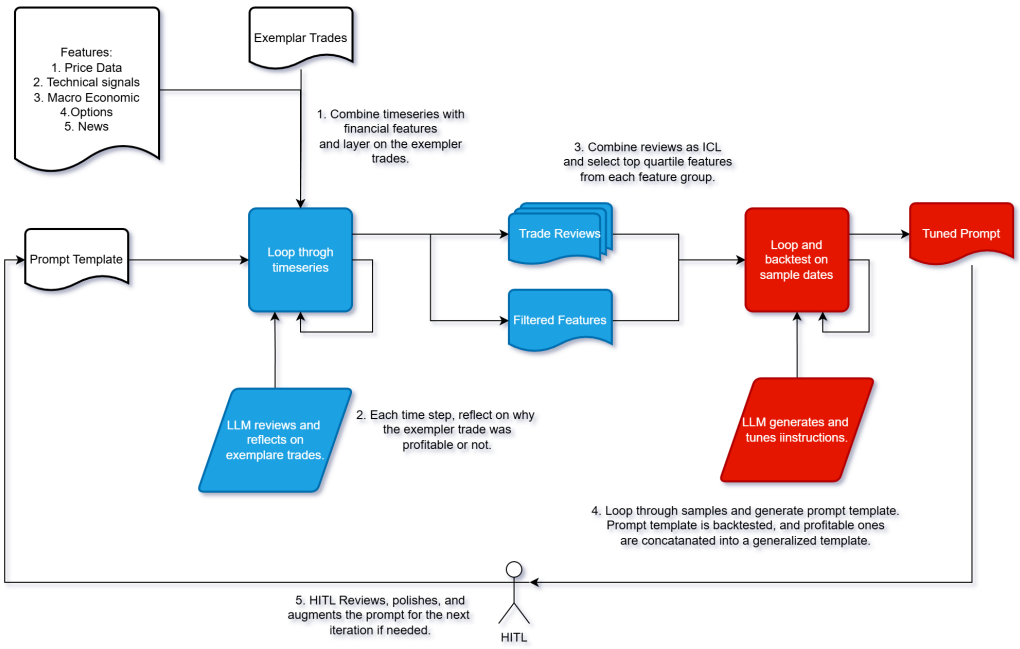

Fig. 1. Prompt tuning workflow.

patterns, allowing π_T^g to evolve through reflection rather than fine-tuning.

c) Prompt Improvement 2 – Instruction Decomposition: To enhance reasoning, instructions and their associated drivers were decomposed [15] into six feature groups: stock, technical, fundamental, macroeconomic, options, and prior strategy reflection [29] within the prompt. Each section incorporates few-shot examples and Chain-of-Thought (CoT) heuristics tailored to its domain. These components are visible in the final tuned prompt (A).

d) Prompt Improvement 3 – News Factors: Unstructured news data was introduced via the analyst agent, which applied CoT-based factor extraction [14]. Entities and timestamps were anonymized to prevent memorization bias [37]. Extracted news factors were ranked and integrated alongside structured indicators. The analyst prompt is provided in (A).

The final system, shown in 2, integrates numerical and textual signals into a global strategy policy π^g. All prompt iterations used in Experiment 1 are summarized in I.

TABLE I
PROMPT VERSIONS USED IN EXPERIMENT 1

Prompt	Description
P1	Baseline prompt containing only static technical indicators and price features.
P2	P1 extended with ICM, incorporating prior strategy and Likert-weighted feature importance.
P3	P2 extended with Instruction Decomposition and CoT reasoning across six structured signal groups.
P4	P3 enriched with macroeconomic and firm-specific news-derived directional signal.

2) Parameters and Evaluation: Prompt tuning was conducted with a temperature setting of 0.7, following the approach of [16], [32], alongside empirically calibrated hyperparameters: a frequency penalty of 1.0 and a presence penalty of 0.25. These settings were chosen to discourage the model from lazily repeating strategies previously reflected on via the ICM buffer.

For strategy generation tasks, temperature was set to 0 with a fixed random seed of 49 to ensure reproducibility. Strategies were generated on a monthly (20-trading-day) cadence, consistent with the guidance and rebalancing cycles typical in institutional portfolio management, and remained computationally tractable given the LLM's inference cost.

During initial prompt tuning, a maximum of three refinement iterations was allowed. Convergence was defined as achieving regret $\mathcal{R}(T) < 0.15$ and exceeding the initial Sharpe Ratio threshold $V^* = 0.10$, which was updated dynamically if surpassed, see (2).

All technical indicators were computed using a 20-trading-day rolling window and standard *TA-Lib* defaults (e.g., 14-day RSI).

a) Quantitative Metrics: We evaluate LLM-generated strategies using three complementary metrics: risk-adjusted returns, model confidence, and model uncertainty.

The SR serves as the core risk-adjusted returns metric:

$$\text{SR} = \frac{\mathbb{E}[R_t - R_f]}{\sigma_R} \tag{3}$$

where R_t is the portfolio return, R_f is the risk-free rate, and σ_R is the return volatility. SR also serves as a proxy for the LLM's financial reasoning [32], [16]. To ensure comparability across different periods, we annualize the SR to 252 trading days per year: Annualized SR $= \text{SR} \cdot \sqrt{252}$.

As a proxy for strategy coherence, we compute the Perplex-

ity (PPL) [38] over the LLM-generated strategies:

$$\text{PPL} = \exp\left(-\frac{1}{N}\sum_{t=1}^{N}\log p(w_t \mid w_{<t})\right) \tag{4}$$

where $p(w_t \mid w_{<t})$ denotes the conditional token probability. Lower values indicate higher model confidence and fluency.

To complement this, we report token-level entropy H_{LLM}, approximated using top-k distributions:

$$H_{\text{LLM}} = \frac{1}{N}\sum_{t=1}^{N}\left(\sum_{v\in V_k} -p_t(v)\log p_t(v) - p_{\text{tail},t}\log p_{\text{tail},t}\right) \tag{5}$$

where V_k is the top-k token set and $p_{\text{tail},t}$ captures the unobserved mass [39]. Lower entropy reflects greater decisiveness; higher values suggest uncertainty.

Together, PPL and H_{LLM} enable a measurement of prompt quality and strategy confidence. Strategies exhibiting low PPL and low entropy were found to perform best in backtests (see Section (VI-A)).

b) Qualitative Evaluation: Qualitative assessment was conducted via the Expert Review Score (ERS), a human-grounded rubric evaluating LLM-generated trading rationales along three dimensions: economic rationale, domain fidelity, and trade safety (risk awareness). Each dimension was scored on a 3-point ordinal scale {1 = poor, 2 = average, 3 = good}, based on the rubric shown in Table II.

The review process followed a similar setup to that of [40], involving ten participants: five senior finance professionals and five retail traders (or professionals in the industry who do not actively trade). Each reviewer evaluated anonymized data for three instruments over one year, including price data, fundamental and macroeconomic metrics, and firm-level news headlines.

Before reviewing the LLM rationale, expert participants made their own directional prediction (`LONG`/`SHORT`) to activate their internal domain models [41]. They then reviewed the LLM's reasoning and scored it using the rubric. Each session concluded with a 60-minute structured discussion to elicit prompt critiques and identify high-quality exemplars. Surveys took approximately 15 minutes to complete. All scores were normalized to a 1–3 range.

TABLE II
EXPERT RUBRIC FOR SCORING LLM RATIONALES.

Criterion	1	2	3
Rationale	Flawed	Partial	Sound
Fidelity	Unrealistic	Plausible	Professional
Safety	Ignored	Mentioned	Addressed

C. Experiment 2: LLM-Guided RL

This experiment addressed Objective 2 (Section II).

a) Data and Feature Engineering: The LLM outputs from Experiment 1 were reused. The RL agent adopted the DDQN configuration of [6], with a single LLM-derived interaction term τ appended to the observation space. This feature consisted of:

- **Signal Direction** ($\mathsf{dir}(\pi_g)$): The discrete directional recommendation from the LLM, mapped from $\{0, 1\}$ to $\{-1, 1\}$ to represent `SHORT` and `LONG` positions.
- **Signal Strength** ($\mathsf{str}(\pi_g)$): The LLM's entropy-adjusted confidence score as a Likert-3 score.

The interaction term was defined as:

$$\tau = \mathsf{dir}(\pi_g)\cdot\mathsf{str}(\pi_g) \tag{6}$$

The LLM's signal strength was derived from the normalized LLM's confidence score:

$$\mu_{\text{conf}} = \frac{\text{Likert}}{3} \tag{7}$$

and adjusted using entropy-based certainty:

$$C = \varepsilon + (1-\varepsilon)(1-H) \tag{8}$$

where $H \in [0, 1]$ is the normalized entropy of the LLM output, and $\varepsilon = 0.01$ ensures numerical stability. The final strength term is:

$$\mathsf{str}(\pi_g) = \mu_{\text{conf}}\cdot C \tag{9}$$

This entropy-adjusted confidence follows the approach of [42], providing a soft weighting of the LLM's signal by its certainty.

The interaction term τ was selected empirically. Initial variants used direction only ($\mathsf{str}(\textit{dir})$), followed by LLM's confidence ($\mathsf{str}(\pi_g)$) and direction. The final form was chosen based on empirical performance and compatibility with DDQN's continuous normalized input space [6].

b) LLM+RL Hybrid Architecture: Figure 2 illustrates the integrated system. The baseline DDQN agent is augmented by the Strategist Agent and Analyst Agent, which produce monthly strategies for the stock's behavior. For practical reasons, outputs from the LLM were precomputed per instrument and fixed throughout training.

c) Training and Parameters: Hyperparameters mirror [6] and the LLM settings follow those in Experiment 1. Training was conducted over 25 runs × 50 episodes per instrument using an NVIDIA RTX 3050, with each equity trained for 3 hours.

To ensure comparability with the benchmark [6], we replicated all baseline metrics within acceptable statistical bounds, see 8.

d) Evaluation Metrics: Two measures were considered:

- **SR:** quantifies the trade-off between excess return and volatility as shown in (3). Higher SR values indicate superior risk-adjusted performance.
- **MDD:** captures the largest observed loss from a historical peak to a subsequent trough:

$$\text{MDD} = \frac{P_{\text{peak}} - P_{\text{low}}}{P_{\text{peak}}} \tag{10}$$

where P_{peak} is the highest portfolio value observed before the largest drop, and P_{low} is the lowest value reached before a new peak is established. MDD quantifies the

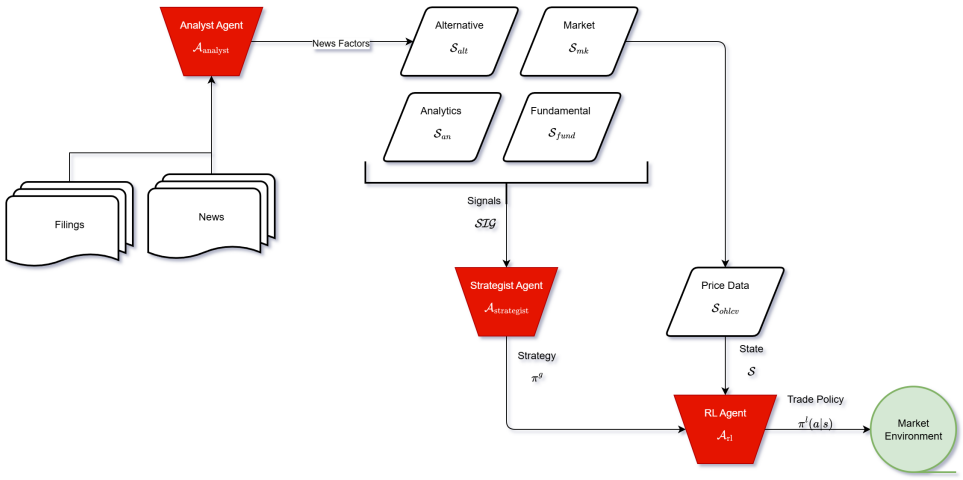

Fig. 2. LLM-guided RL architecture.

TABLE III
SHARPE RATIO PER PROMPT AND BENCHMARK

Ticker	P1	P2	P3	P4	**BM**
AAPL	1.09	1.07	1.07	**2.09**	1.27
AMZN	0.35	0.38	0.63	**0.84**	0.21
GOOGL	0.26	0.52	0.52	**1.12**	0.19
META	-0.06	-0.28	0.30	**0.77**	0.63
MSFT	1.07	1.11	**1.31**	0.50	1.17
TSLA	0.71	0.75	0.43	**0.79**	0.67
Mean	0.57	0.59	0.71	**1.02**	0.69

TABLE IV
PERPLEXITY (PPL) PER PROMPT

Ticker	P1	P2	P3	P4
AAPL	1.85	**1.31**	1.55	1.44
AMZN	1.74	1.35	1.68	**1.31**
GOOGL	1.77	1.49	1.78	**1.33**
META	1.73	**1.31**	1.39	1.38
MSFT	1.83	1.44	1.49	**1.24**
TSLA	1.77	1.50	1.63	**1.39**
Mean	1.78	1.40	1.59	**1.35**

TABLE V
ENTROPY (H_{LLM}) PER PROMPT

Ticker	P1	P2	P3	P4
AAPL	0.70	0.67	**0.66**	0.69
AMZN	0.69	0.69	0.69	**0.67**
GOOGL	0.67	0.67	0.70	**0.66**
META	**0.66**	0.70	0.73	0.67
MSFT	0.66	0.68	0.72	**0.65**
TSLA	0.68	0.70	0.74	**0.65**
Mean	0.68	0.68	0.71	**0.67**

maximum percentage decline from peak to trough over the evaluation period. Lower values indicate stronger downside protection.

These metrics together assess whether LLM-guided RL agents can adapt to different equities without changing the core architecture.

VI. RESULTS AND DISCUSSION

A. Experiment 1 Results

This section presents empirical results across four prompt versions, P1 to P4 in I. We assess their impact on quantitative metrics: SR, PPL, and entropy (H_{LLM})—and, from Prompt 4 onward, qualitative evaluation via ERS. Expert review surveys were introduced only from Prompt 4, once prompt structure stabilized sufficiently to support HITL assessment.

By examining these metrics, we evaluate how structured prompting influences the quality and safety of the generated strategies, addressing Objective 1, see II.

Tables III–V summarize results across prompt versions compared to the benchmark (BM). Prompt 1 (baseline) relied on static technical features and performed worst, with the lowest SR and highest PPL and entropy. Prompt 2 added ICM, yielding moderate gains in SR (mean 0.59), suggesting improvements in confidence from prior reflection.

Prompt 3 introduced decomposed CoT reasoning and outperformed the benchmark with mean SR of 0.71. Prompt 4 further included unstructured news signals and achieved the highest mean SR (1.02), lowest PPL and entropy, and showed higher confidence particularly on sentiment-sensitive tickers like TSLA.

Expert evaluation of Prompt 4 confirmed its effectiveness in synthesizing structured and unstructured signals. Reviewers rated the LLM's rationale highly (mean = 2.7 out of 3), highlighting its ability to integrate valuation, sentiment, and analytics.

Fidelity received a slightly lower score (mean = 2.65), with critiques focused on inconsistent thresholding. For instance, one reviewer noted, *"Calling RSI near 40 'oversold' is debatable,"* prompting refinements in numerical phrasing.

Feedback varied by background: buy-side professionals emphasized transparency in feature weighting, whereas retail

TABLE VI
EXPERT REVIEWER SCORES FOR PROMPT 4

Dimension	ERS (1–3)
Rationale	2.70
Fidelity	2.65
Safety	2.80

reviewers focused on coherence between technical and macro signals. All commented on the lack of a neutral or hold signal, which was done to align with [6].

Overall, results validated Prompt 4's modular design and market narrative awareness. It outperformed earlier prompts and was selected as the global policy generation prompt for the LLM-RL hybrid in Experiment 2.

B. Experiment 2 Results

This experiment addressed Objective 2 by comparing three agent architectures: (i) a baseline RL-only [6], (ii) the best-performing LLM prompt from Experiment 1, and (iii) a hybrid LLM+RL agent. All agents were trained in identical environments.

To determine whether the hybrid agent outperformed the baseline, we conducted two-sided paired t-tests on the SR across 25 runs for each stock. The null hypothesis H_0 assumed no difference in mean performance: $H_0 : \mu_{\text{LLM+RL}} = \mu_{\text{RL-only}}$. All resulting p-values were below 0.05, indicating statistically significant differences from (8).

TABLE VII
EXPERIMENT 2 RESULTS FOR SHARPE RATIO.

Ticker	LLM+RL (σ)	RL-Only (σ)	LLM-Only
AAPL	1.70 (0.43)	1.42 (0.05)	**2.09**
AMZN	**1.21 (0.58)**	0.42 (0.23)	0.84
GOOGL	**1.16 (0.17)**	0.23 (0.37)	1.12
META	0.46 (0.75)	0.15 (0.61)	**0.77**
MSFT	**1.16 (0.28)**	0.99 (0.30)	0.50
TSLA	**0.92 (0.19)**	0.62 (0.60)	0.87
Mean	**1.10**	0.64	1.03

TABLE VIII
EXPERIMENT 2 RESULTS FOR MAXIMUM DRAWDOWN.

Ticker	LLM+RL (σ)	RL-Only (σ)	LLM-Only
AAPL	0.29 (0.20)	0.45 (0.01)	**0.28**
AMZN	0.26 (0.12)	**0.19 (0.14)**	0.34
GOOGL	0.28 (0.06)	**0.25 (0.18)**	0.35
META	0.35 (0.11)	0.45 (0.27)	**0.30**
MSFT	0.19 (0.08)	**0.17 (0.09)**	0.21
TSLA	**0.46 (0.05)**	0.65 (0.13)	0.59
Mean	**0.31**	0.36	0.35

Results in VII confirm that the LLM+RL agent outperformed the RL-only baseline in 4 out of 6 assets.

AAPL and META did not show consistent individual outperformance. Figure (3) illustrates AAPL's trading behavior during one episode. The top panel plots price, technical indicators, and trades: hollow triangles mark RL trades; filled arrows show LLM monthly guidance. The LLM issued sparse but confident signals (strength ¿ 0.6), often aligned with technical points of interest (e.g., MA interactions). In contrast, the RL agent frequently mistimed entries and exits.

From December 2018 to January 2019, the RL agent has oscillated between LONG and SHORT positions with punishing results and despite receiving strong signals from the LLM. The LLM has issued high-confidence guidance for a SHORT in December followed by a LONG in January, both with signal strengths exceeding 0.8. Regardless, the RL agent has held a LONG position throughout the decline. As seen in (6), the DDQN infers comparatively low Q-values to SHORT actions, indicating weak to no learned confidence. This behavior is a result of the trade environment's constraints designed to limit leverage [6].

The bottom panel confirms that the LLM has maintained high confidence near key inflection points and reduced conviction when trends have persisted (possibly awaiting a reversal from its training corpus). However, the RL agent has not fully exploited these signals due to limitations in the underlying RL configuration, which has remained fixed for the purposes of this experiment.

Figure (4) illustrates the evolution of the Sharpe ratio for AAPL through out the training episodes. The hybrid LLM+RL agent (orange line) outperformed the baseline RL agent (blue line) in both mean Sharpe and stability, as reflected in the narrower shaded confidence intervals. The LLM's SR is shown for reference (black dashed line).

Figures (5) and (6) show Q-values for `LONG` and `SHORT` actions respectively, with y-axis clipped to $[-0.03, 0.03]$ to highlight late-episode convergence. Early training was noisy for both agents. The LLM+RL agent converged faster with lower variance. Although Q-value separation rarely exceeded 0.01, the hybrid showed slightly stronger directional signals. These gains emerged without modifying the DDQN or imposing reward shaping, thus isolating the effect of the LLM's guidance. The constrained Q-range stems from the RL baseline design.

The hybrid agent did not consistently minimize MDD per stock but achieved values close to the best across agents, with the lowest overall mean (0.31). This suggests overall smoother drawdowns under uncertainty across the universe.

VII. CONCLUSION AND FUTURE WORK

This study has explored am RL+LLM hybrid architecture for algorithmic trading, where LLMs generate guidance for RL agents to act as tactical executors.

Experiment 1 has shown that carefully constructed prompts improve the LLM's performance, with Prompt 4 achieving the highest SR and lowest uncertainty. Expert evaluations confirmed the rationale of generated strategies within the domain.

Experiment 2 has demonstrated that an RL agent guided by LLM signals outperforms the RL-only baseline in four out of

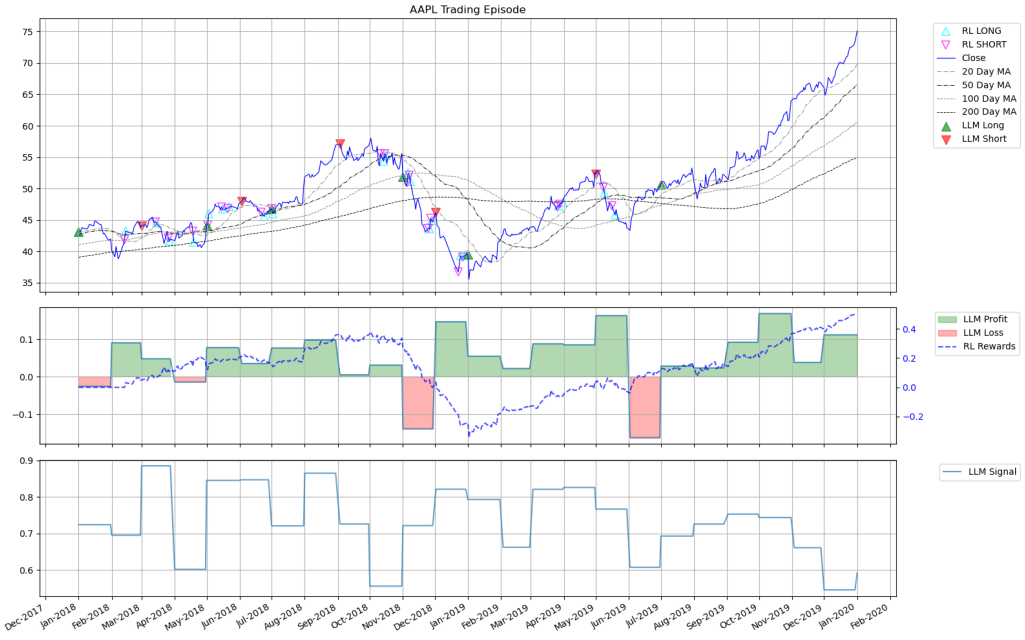

Fig. 3. AAPL's performance with LLM+RL model.

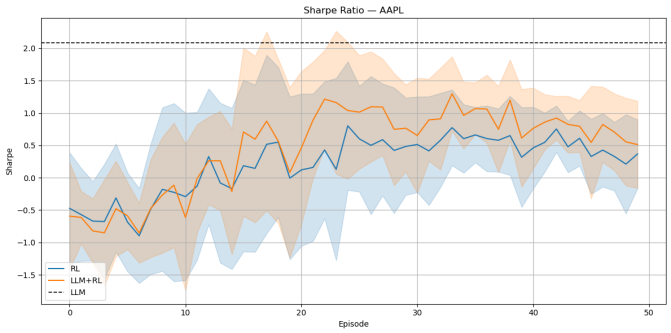

Fig. 4. Training behavior for AAPL: Sharpe ratio.

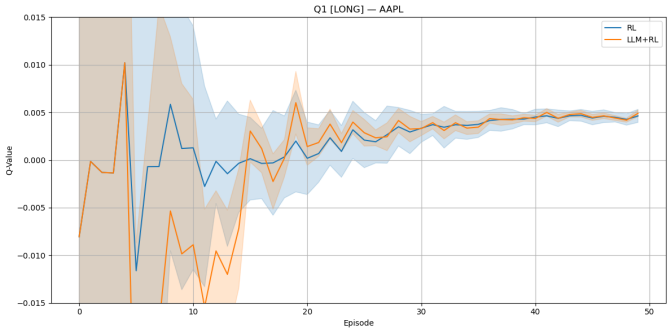

Fig. 5. Training behavior for AAPL: Q-values for LONG.

six stocks when evaluated by their Sharpe Ratio. While MDD was not consistently reduced, the overall drawdowns remained low on average. Importantly, the underlying RL architecture

Fig. 6. Training behavior for AAPL: Q-values for SHORT.

was not modified; all observed improvements stemmed from LLM guidance.

Future research should address two main directions. First, while the LLM can guide the RL, reward shaping is necessary to attain optimal results. Second, modular specialization through multiple LLM agent prompted for specific domains may enable a mixture-of-experts architecture, and lessen the risk of confabulation.

Overall, this work presents a novel LLM+RL system that improves both return and risk outcomes. It supports modular, agentic setups where LLMs operate as trustworthy planners in financial decision making.

ACKNOWLEDGMENT

We thank the expert reviewers who contributed their domain expertise to this work.

REFERENCES

[1] M. López de Prado, "Beyond econometrics: A roadmap towards financial machine learning," *Ssrn*, 2020. [Online]. Available: https://ssrn.com/abstract=3365282

[2] X.-Y. Liu, H. Yang, Q. Chen, R. Zhang, L. Yang, B. Xiao, and C. D. Wang, "Finrl: A deep reinforcement learning library for automated stock trading in quantitative finance," 2022. [Online]. Available: https://arxiv.org/abs/2011.09607

[3] T.-V. Pricope, "Deep reinforcement learning in quantitative algorithmic trading: A review," 2021. [Online]. Available: https://arxiv.org/abs/2106.00123

[4] J. Joubert, D. Sestovic, I. Barziy, W. Distaso, and M. López de Prado, "The three types of backtests," *Ssrn*, 2024. [Online]. Available: https://ssrn.com/abstract=4897573

[5] M. Xu, Z. Lan, Z. Tao, J. Du, and Z. Ye, "Deep reinforcement learning for quantitative trading," 2023. [Online]. Available: https://arxiv.org/abs/2312.15730

[6] T. Théate and D. Ernst, "An application of deep reinforcement learning to algorithmic trading," *Expert Systems with Applications*, vol. 173, p. 114632, Jul. 2021. [Online]. Available: http://dx.doi.org/10.1016/j.eswa.2021.114632

[7] L. Takara, A. Santos, V. Mariani, and L. Coelho, "Deep reinforcement learning applied to a sparse-reward trading environment with intraday data," *Expert Systems with Applications*, vol. 238, p. 121897, Oct. 2023.

[8] S. Booth, W. Knox, J. Shah, S. Niekum, P. Stone, and A. Allievi, "The perils of trial-and-error reward design: Misdesign through overfitting and invalid task specifications," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 37, pp. 5920–5929, 06 2023.

[9] X. Wang *et al.*, "Improving generalization in reinforcement learning with mixture regularization," *Advances in Neural Information Processing Systems*, 2020.

[10] R. Devidze, P. Kamalaruban, and A. K. Singla, "Exploration-guided reward shaping for reinforcement learning under sparse rewards," in *Proceedings of the Conference on Neural Information Processing Systems*, 2022.

[11] L. Onozo, F. Arthur, and B. Gyires-Tóth, "Leveraging llms for financial news analysis and macroeconomic indicator nowcasting," *IEEE Access*, vol. Pp, pp. 1–1, 01 2024.

[12] H. Yang, X.-Y. Liu, and C. D. Wang, "Fingpt: Open-source financial large language models," 2023. [Online]. Available: https://arxiv.org/abs/2306.06031

[13] A. H. Huang, H. Wang, and Y. Yang, "Finbert: A large language model for extracting information from financial text," *Contemporary Accounting Research*, vol. 40, no. 2, pp. 806–841, 2023.

[14] M. Wang, K. Izumi, and H. Sakaji, "Llmfactor: Extracting profitable factors through prompts for explainable stock movement prediction," 2024. [Online]. Available: https://arxiv.org/abs/2406.10811

[15] S. Schulhoff, M. Ilie, N. Balepur, K. Kahadze, A. Liu, C. Si, Y. Li, A. Gupta, H. Han, S. Schulhoff, P. S. Dulepet, S. Vidyadhara, D. Ki, S. Agrawal, C. Pham, G. Kroiz, F. Li, H. Tao, A. Srivastava, H. D. Costa, S. Gupta, M. L. Rogers, I. Goncearenco, G. Sarli, I. Galynker, D. Peskoff, M. Carpuat, J. White, S. Anadkat, A. Hoyle, and P. Resnik, "The prompt report: A systematic survey of prompting techniques," 2024. [Online]. Available: https://arxiv.org/abs/2406.06608

[16] Y. Yu, H. Li, Z. Chen, Y. Jiang, Y. Li, D. Zhang, R. Liu, J. W. Suchow, and K. Khashanah, "Finmem: A performance-enhanced llm trading agent with layered memory and character design." in *AAAI Spring Symposia*, R. P. A. Petrick and C. W. Geib, Eds. AAAI Press, Jan. 2024, pp. 595–597. [Online]. Available: http://dblp.uni-trier.de/db/conf/aaaiss/aaaiss2024.html#YuLCJLZLSK24

[17] Y. Hu, X. Wang, W. Yao, Y. Lu, D. Zhang, H. Foroosh, D. Yu, and F. Liu, "Define: Enhancing llm decision-making with factor profiles and analogical reasoning," 2024. [Online]. Available: https://arxiv.org/abs/2410.01772

[18] M. Chlistalla, "High-frequency trading and long-term investment: A systematic review," Deutsche Bank Research, Frankfurt am Main, Germany, Tech. Rep. Research Briefing, Feb. 2011, editor: Bernhard Speyer, Technical Assistant: Sabine Kaiser. [Online]. Available: www.dbresearch.com

[19] S. M. Bartram, J. Branke, and M. Motahari, *Artificial intelligence in asset management*. CFA Institute Research Foundation, 2020.

[20] E. F. Fama and K. R. French, "A five-factor asset pricing model," *Journal of Financial Economics*, vol. 116, no. 1, pp. 1–22, 2015. [Online]. Available: https://ssrn.com/abstract=2287202

[21] M. López de Prado, "The 10 reasons most machine learning funds fail," True Positive Technologies, Tech. Rep., Jan. 2018, first version: December 25, 2017. This version: January 27, 2018. [Online]. Available: https://ssrn.com/abstract=3104816

[22] T. J. Moskowitz, Y. H. Ooi, and L. H. Pedersen, "Time series momentum," *Journal of Financial Economics*, vol. 104, no. 2, pp. 228–250, 2012.

[23] E. Chan, *Algorithmic trading: winning strategies and their rationale*. John Wiley & Sons, 2013.

[24] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction*, 2nd ed. The MIT Press, 2018. [Online]. Available: http://incompleteideas.net/book/the-book-2nd.html

[25] K. Arulkumaran, M. P. Deisenroth, M. Brundage, and A. A. Bharath, "Deep reinforcement learning: A brief survey," *IEEE Signal Processing Magazine*, vol. 34, no. 6, pp. 26–38, 2017. [Online]. Available: https://doi.org/10.1109/MSP.2017.2743240

[26] H. van Hasselt, A. Guez, and D. Silver, "Deep reinforcement learning with double q-learning," 2015, cite arxiv:1509.06461Comment: AAAI 2016. [Online]. Available: http://arxiv.org/abs/1509.06461

[27] A. Lazaridis, A. Fachantidis, and I. Vlahavas, "Deep reinforcement learning: A state-of-the-art walkthrough," *Journal of Artificial Intelligence Research*, vol. 69, pp. 1421–1471, 2020.

[28] H. Ding, Y. Li, J. Wang, and H. Chen, "Large language model agent in financial trading: A survey," *arXiv preprint arXiv:2408.06361*, 2024. [Online]. Available: https://arxiv.org/abs/2408.06361

[29] W. Zhang, L. Zhao, H. Xia, S. Sun, J. Sun, M. Qin, X. Li, Y. Zhao, Y. Zhao, X. Cai, L. Zheng, X. Wang, and B. An, "A multimodal foundation agent for financial trading: Tool-augmented, diversified, and generalist," 2024. [Online]. Available: https://arxiv.org/abs/2402.18485

[30] X.-Y. Liu, H. Yang, J. Gao, and C. D. Wang, "Finrl: deep reinforcement learning framework to automate trading in quantitative finance," in *Proceedings of the Second ACM International Conference on AI in Finance*. Acm, Nov. 2021. [Online]. Available: http://dx.doi.org/10.1145/3490354.3494366

[31] X.-Y. Liu, J. Rui, J. Gao, L. Yang, H. Yang, Z. Wang, C. D. Wang, and J. Guo, "Finrl-meta: A universe of near-real market environments for data-driven deep reinforcement learning in quantitative finance," 2022. [Online]. Available: https://arxiv.org/abs/2112.06753

[32] A. Lopez-Lira and Y. Tang, "Can chatgpt forecast stock price movements? return predictability and large language models," May 2023. [Online]. Available: https://arxiv.org/abs/2304.07619

[33] J. Huang and K. C.-C. Chang, "Towards reasoning in large language models: A survey," 2023. [Online]. Available: https://arxiv.org/abs/2212.10403

[34] A. Chaddha and S. Yadav, "Examining the predictive power of moving averages in the stock market," *Journal of Student Research*, vol. 11, no. 3, 2022. [Online]. Available: https://www.jsr.org

[35] R. Alsini, Q. A. Al-Haija, A. A. Alsulami, B. Alturki, A. A. Alqurashi, M. D. Mashat, A. Alqahtani, and N. Alhebaishi, "Forecasting cryptocurrency's buy signal with a bagged tree learning approach to enhance purchase decisions," *Frontiers in Big Data*, vol. 7, p. 1369895, 2024. [Online]. Available: https://www.frontiersin.org/articles/10.3389/fdata.2024.1369895/full

[36] S. Wang, H. Yuan, L. M. Ni, and J. Guo, "Quantagent: Seeking holy grail in trading by self-improving large language model," 2024. [Online]. Available: https://arxiv.org/abs/2402.03755

[37] A. Lopez-Lira, Y. Tang, and M. Zhu, "The memorization problem: Can we trust llms' economic forecasts?" 2025. [Online]. Available: https://arxiv.org/abs/2504.14765

[38] H. Gonen, S. Iyer, T. Blevins, N. Smith, and L. Zettlemoyer, "Demystifying prompts in language models via perplexity estimation," in *Findings of the Association for Computational Linguistics: EMNLP 2023*, H. Bouamor, J. Pino, and K. Bali, Eds. Singapore: Association for Computational Linguistics, Dec. 2023, pp. 10 136–10 148. [Online]. Available: https://aclanthology.org/2023.findings-emnlp.679/

[39] A. Kaltchenko, "Entropy heat-mapping: Localizing gpt-based ocr errors with sliding-window shannon analysis," *arXiv preprint arXiv:2505.00746*, 2025.

[40] L. M. Demajo, V. Vella, and A. Dingli, "Explainable ai for interpretable credit scoring," in *Computer Science & Information Technology*

(CS & IT), ser. Acity 2020. AIRCC Publishing Corporation, Nov. 2020. [Online]. Available: http://dx.doi.org/10.5121/csit.2020.101516

[41] R. R. Hoffman, S. T. Mueller, G. Klein, and J. Litman, "Metrics for explainable ai: Challenges and prospects," DARPA XAI Program, Technical Report arXiv:1812.04608, 2018. [Online]. Available: https://arxiv.org/abs/1812.04608

[42] G. Yona, R. Aharoni, and M. Geva, "Can large language models faithfully express their intrinsic uncertainty in words?" *arXiv preprint arXiv:2405.16908*, 2024.

[43] H. Yang, B. Zhang, N. Wang, C. Guo, X. Zhang, L. Lin, J. Wang, T. Zhou, M. Guan, R. Zhang, and C. D. Wang, "Finrobot: An open-source ai agent platform for financial applications using large language models," 2024. [Online]. Available: https://arxiv.org/abs/2405.14767

[44] J. Yoon and J. Fan, "Forecasting the direction of the fed's monetary policy decisions using random forest," *Journal of Forecasting*, vol. 43, no. 7, pp. 2848–2859, 2024. [Online]. Available: https://doi.org/10.1002/for.3144

[45] F. Dakalbab, M. A. Talib, Q. Nasir, and T. Saroufil, "Artificial intelligence techniques in financial trading: A systematic literature review," *Journal of King Saud University - Computer and Information Sciences*, vol. 36, p. 102015, Mar. 2024. [Online]. Available: https://www.sciencedirect.com

APPENDIX

STRATEGY PROMPT

The final tuned prompt from Experiment 1 and the LLM strategy generator for Experiment 2, is available in (1).

Listing 1. Tuned Strategy Prompt

```
User_Context:
  Last_Strategy_Used_Data:
    last_returns: "{Last_LLM_Strat_Returns}"
    last_action: "{Last_LLM_Strat_Action}"
    Rationale: |
      """{Last_LLM_Strat}"""


  Stock_Data:
    General:
      Beta: {Market_Beta}
      Classification: {classification}

    Last_Weeks_Price:
      Close: "{Close}"
      Volume: "{Volume}"

    Weekly_Past_Returns: "{Weekly_Past_Returns}"

    Historical_Volatility:
      HV_Close: "{HV_Close}"

    Implied_Volatility:
      IV_Close: "{IV_Close}"

  Fundamental_Data:
    Ratios:
      Current_Ratio: "{Current_Ratio}"
      Quick_Ratio: "{Quick_Ratio}"
      Debt_to_Equity_Ratio: "{Debt_to_Equity_Ratio}"
      PE_Ratio: "{PE_Ratio}"
    Margins:
      Gross_Margin: "{Gross_Margin}"
      Operating_Margin: "{Operating_Margin}"
      Net_Profit_Margin: "{Net_Profit_Margin}"
    Growth Metrics:
      EPS_YoY: "{EPS_YoY_Growth}"
      Net_Income_YoY: "{Net_Income_YoY_Growth}"
      Free_Cash_Flow_YoY: "{Free_Cash_Flow_Per_Share_YoY_Growth}"

  Technical_Analysis:
    Moving_Averages:
        20MA: "{20MA}"
        50MA: "{50MA}"
        200MA: "{200MA}"
    MA_Slopes:
        20MA_Slope: "{20MA_Slope}"
        50MA_Slope: "{50MA_Slope}"
        100MA_Slope: "{100MA_Slope}"
        200MA_Slope: "{200MA_Slope}"
    MACD:
        Value: "{MACD}"
        Signal_Line: "{Signal_Line}"
        MACD_Strength: {MACD_Strength}
    RSI:
        Value: "{RSI}"
    ATR: "{ATR}"

  Macro_Data:
    Macro_Indices:
      SPX:
        Close: "{SPX_Close}"
        Close_20MA: "{SPX_Close_MA}"
        Close_Slope: "{SPX_Close_Slope}"
      VIX:
        Close: "{VIX_Close}"
        Close_20MA: "{VIX_Close_MA}"
        Close_Slope: "{VIX_Close_Slope}"
    Economic_Data:
      GDP_QoQ: "{GDP_QoQ}"
      PMI: "{PMI}"
      Consumer_Confidence_QoQ: "{Consumer_Confidence_QoQ}"
      M2_Money_Supply_QoQ: "{M2_Money_Supply_QoQ}"
      PPI_YoY: "{PPI_YoY}"
      Treasury_Yields_YoY: "{Treasury_Yields_YoY}"

  Options_Data:
    Put_IV_Skews:
      OTM_Skew: "{OTM_Skew}"
      ATM_Skew: "{ATM_Skew}"
      ITM_Skew: "{ITM_Skew}"
    20Day_Moving_Averages:
      OTM_Skew_MA: "{MA_OTM_Skew}"
      ATM_Skew_MA: "{MA_ATM_Skew}"
      ITM_Skew_MA: "{MA_ITM_Skew}"

  News_Sentiment: {news_sentiment}
  News_Impact_Score: {news_impact_score}

System_Context(System):
  Persona: {persona}
  Portfolio_Objectives: {portfolio_objectives}
  Instructions: |
    Develop a LONG or SHORT trading strategy for a single stock only for the next
          Month that aligns with the `portfolio_objectives`. Follow these
        guidelines:

    1. Stock Analysis:
       - Evaluate price trends: Compare the Close price against 20MA, 50MA, and
            200MA to assess momentum or reversals.
       - Analyze returns: Use Weekly Past Returns to validate trend
            sustainability.
       - Contextualize volatility: Align `HV_Close` and `HV_High` with recent
            price action for trend validation.
       - Incorporate beta: Use `beta` to gauge sensitivity to market movements.

       - ICL Example: "Close␣price␣above␣20MA␣and␣50MA␣with␣steep␣20MA␣slope␣
            signals␣bullish␣momentum.␣Weekly␣returns␣confirm␣a␣sustainable␣
            uptrend."

    2. Technical Analysis:
       - Use RSI: Identify momentum signals (>70 overbought; <30 oversold) and
            divergences for reversals.
       - Validate with `MACD`: Use crossovers of `MACD.Value` and `Signal_Line`,
            and `MACD_Strength` for directional confidence.
       - Leverage `RSI.value` divergences, and steep `Moving_Averages` slopes. Or
             focus on stable `Moving_Averages` patterns on stable historical
            volatility `HV_Close`.
       - ICL Example: "RSI␣at␣65,␣a␣positive␣MACD␣crossover␣indicate␣bullish␣
            momentum."

    3. Fundamental Analysis:
       - Evaluate growth metrics: Use `EPS_YoY`, `Net_Income_YoY`, and `
            Free_Cash_Flow_YoY`for profitability and sustainability.
       - Prioritize ratios: Low `Debt_to_Equity_Ratio` and `Current_Ratio`
            reflect financial stability.
       - Focus on aggressive `Growth Metrics` and earnings news.
       - ICL Example: "EPS␣YoY␣growth␣of␣25%␣and␣low␣Debt-to-Equity␣ratio␣of␣0.5␣
            support␣strong␣financial␣health,␣aligning␣with␣a␣LONG␣strategy."

    4. Macro Analysis:
      - Align with market sentiment across `Macro_Data`:
        - "SPX␣Close␣Slope␣>␣0␣&&␣VIX␣Close␣Slope␣<␣0": Bullish (Risk-On)
        - "SPX␣Close␣Slope␣<␣0␣&&␣VIX␣Close␣Slope␣>␣0": Bearish (Risk-Off)
      - Validate with `Economic_Data`:
        - "GDP␣QoQ␣>␣0␣&&␣`PMI`␣>␣50" leads to Economic Expansion
        - "`Treasury␣Yields␣YoY`␣<␣0" Signals Recession Risk, especially if
             already mentioned in `Rationale`.

      - ICL Examples:
        - "`SPX␣Close␣Slope`␣>␣0␣&&␣`VIX␣Close␣Slope`␣<␣0␣We␣have␣Market␣
            Confidence"
        - "`GDP␣QoQ`␣Falling␣&&␣`PMI`␣<␣50␣We␣have␣an␣Economic␣Slowdown."

    5. Options Analysis:
      - Compare `OTM_Skew`, `ATM_Skew`, and `ITM_Skew` IV Skews: Assess
            differences to gauge market sentiment and directional bias using
            their `20Day_Moving_Averages`.
      - Leverage IV spikes to capitalize on speculative directional trades.
      - Example: "Rising␣`ATM_Skew_MA`␣>␣0,␣market␣pricing␣up␣move,␣with␣stable␣
            HV␣supports␣a␣LONG␣position,␣as␣it␣indicates␣growing␣upside␣
            expectations␣without␣excessive␣fear."

    6. News Analysis:
      - Use `News_Sentiment` and `News_Impact_Score` (1-3).
      - Only strong directional news (score = 3) should override other signals.
      - Medium news (score = 2) supports but does nost lead.
      - Always check if news contradicts macro or technical trend.

    7. Performance Reflection and Strategic Adaptation:
      - If `Last_Strategy_Used_Data` is available:
            - Assess the outcome of the previous strategy by examining `
                last_returns` and the chosen `last_action`.
            - Determine if the result aligns with the expectations outlined in
                the previous `Rationale`.
            - Identify if the direction (LONG or SHORT) led to desirable or
                undesirable outcomes.
            - You must NOT reuse or copy the previous `Rationale`. It is only
                context for reflection.
            - Summarize in 1-2 sentences whether the previous strategy performed
                as expected.
            - Example: "The␣previous␣LONG␣strategy␣yielded␣positive␣returns,␣
                confirming␣the␣bullish␣setup␣based␣on␣RSI␣and␣moving␣averages.
                "
            - Do NOT include language or phrasing from the previous rationale.
      - Confidence assignment:
```

```
      - Assign a Likert score (1 to 3) to your 'action_confidence':
        - 1: Low confidence; contradictory or weak alignment across features.
        - 2: Moderate confidence; partial alignment with moderate evidence.
        - 3: High confidence; strong convergence across key features.
    - Feature Attribution:
      - Rank the importance of each major feature used in your current rationale using a Likert scale (1 to 3):
        - 1: Minimal contribution; not required for the decision.
        - 2: Moderate contribution; relevant but not critical.
        - 3: High contribution; pivotal to the trading decision.

Output:
  action: Str. LONG or SHORT.
  action_confidence: int. Likert scale (1-3) confidence in the proposed 'action', adjusted based on prior strategy outcome if 'Last_Strategy_Used_Data' is available.
  explanation: >
    A concise rationale (max 350 words) justifying the proposed 'action'.
    Include:
      - The top 5 weighted features used in the decision, each labeled with its Likert importance (1-3).
        (e.g., "Stock_Data.Price.Close, Weight 3, Technical_Analysis.RSI.Value, Weight 1, Options_Data.ATM_Skew, Weight 2")
      - A reflective assessment of 'Last_Strategy_Used_Data', including whether the past 'action' was successful and was it maintained given prior 'Rationale'.
  features_used:
    - feature: the features used from the prompt's context.
      direction: LONG, SHORT, or NEUTRAL
      weight: A Likert score (1 to 3) described in Feature Attribution.
```

Analyst Prompt

The Analyst prompt used in Experiment 1 is presented in Listing 2, adapted from [14]. News corpora were anonymized prior to prompting.

Listing 2. Analyst Prompt

```
User_Context:
  Monthly_News_Articles_List: |
    "{articles_list}"

System_Context:
  Persona: Financial Market Analyst
  Instructions: |
    Extract the 'Top 3' news factors influencing stock price movements from the 'Monthly_News_Articles_List'. Follow these steps:

    1. Rank the news by relevance to stock price movements:
       - Prioritize news related to significant financial or market impacts (e.g., acquisitions, partnerships, guidance revisions).
       - Weigh industry trends, macroeconomic influences, and analyst ratings based on their expected effect on the company valuation.
       - News with broad or long-term implications ranks higher.

    2. Summarize content into key factors and corporate events affecting stock prices, using concise language and causal relationships.

    3. For each factor, assign:
       - 'Sentiment': +1 for positive, -1 for negative, 0 for neutral or mixed
       - 'Market_Impact_Score': Likert scale from 1 to 3, where:
         - 1 = minimal relevance
         - 2 = moderate influence
         - 3 = high impact driver

    Examples of factors influencing stock prices include:
      - Strategic partnerships or competitor activity.
      - Industry trends or macroeconomic influences.
      - Product launches or market expansions.
      - Analyst ratings, significant stock price moves, or expectations.
      - Corporate events: guidance revisions, acquisitions, contracts, splits, repurchases, dividends.

    Example:
      'A_major_tech_company_partners_with_a_leading_automotive_firm_for_EV_battery_innovation._Analysts_predict_this_could_boost_revenues_significantly.'

      Ranked Factors:
        1. factor: Strategic partnership in EV battery technology expected to increase revenue.
           sentiment: +1
           market_impact: 3
        2. factor: Positive sentiment driven by projected long-term gains.
           sentiment: +1
           market_impact: 2
        3. factor: Growing demand for EV technology anticipated to support future earnings.
           sentiment: +1
           market_impact: 2

Output:
  factors:
    - factor: str. Summary of the news item. Max 70 words.
    - sentiment: int. One of Positive +1, Negative -1, or Neutral 0
    - market_impact: int. Likert scale 1 to 3
```

Algorithms

The labeling algorithm emulates expert trading behavior by deliberately leveraging future return information to assign proxy trade actions in hindsight. This approach offers a cost-effective and scalable addition to manual annotation, capturing the general direction an informed trader might take. These synthetic labels are then provided to the LLM, along with a smaller set of HITL annotated examples.

Algorithm 1: Expert Trade Heuristic

Data: Time-indexed price series
Result: Trade action: LONG (1) or SHORT (0)

foreach *date* t *in dataset* **do**
  $P_t \leftarrow \text{Close}(t)$;
  $r^{(10)} \leftarrow \frac{P_{t+10}}{P_t} - 1$, $r^{(20)} \leftarrow \frac{P_{t+20}}{P_t} - 1$;
  $r^{\text{weighted}} \leftarrow 0.4 \cdot r^{(10)} + 0.6 \cdot r^{(20)}$;
  if $r^{weighted} >= 0$ **then**
    Action $\leftarrow$ LONG (`Trade_Action` $= 1$);
  else
    Action $\leftarrow$ SHORT (`Trade_Action` $= 0$);

Dataset

Market Data

This market data ($\mathcal{S}_{\text{mk}}$) included OHLCV price series as well as macro-level indicators and forward-looking sentiment signals. Specifically, it comprised:

- Daily returns of the S&P 500 Index (SPX) and NASDAQ-100 Index (NDX). These are market and sector indices,
- Implied Volatility (IV) and Historical Volatility (HV) metrics, derived from the stock's derivatives,
- The CBOE Volatility Index (VIX) as a proxy for market fear and option market expectations,
- *Weekly Past Returns*, which record the percentage change over the past four weekly intervals. The four-week span was selected empirically to align with the model's monthly strategy generation frequency.

These features are help in modeling short-term market dynamics.

Fundamental Data

Fundamental data ($\mathcal{S}_{\text{fund}}$) has firm-level fundamentals and macroeconomic indicators, to serve as low-frequency anchors for the high-frequency noise in market data [43]. Macroeconomic variables provided contextual narrative for interpreting observed signals, and supporting regime identification [44], [29], [11]. This set covered:

- **Liquidity ratios:** Current Ratio, Quick Ratio;
- **Leverage and coverage:** Debt-to-Equity, Interest Coverage;
- **Profitability metrics:** Gross Margin, Operating Margin, Return on Equity (ROE), Return on Assets (ROA);
- **Valuation:** Price-to-Earnings (P/E), Price-to-Book (P/B), Enterprise Value (EV), and Earnings Before Interest, Taxes, Depreciation, and Amortization (EBITDA).
- **Growth:** Revenue and Earnings Growth;

- **Macroeconomic indicators:** Gross Domestic Product (GDP), Purchasing Managers' Index (PMI), Producer Price Index (PPI), Consumer Confidence Index (CCI), U.S. 10-Year Treasury Yield, and the 10Y–2Y yield curve slope.

To enhance temporal abstraction, all variables were computed as quarter-over-quarter (QoQ) or year-over-year (YoY) percentage changes. It is critical to take first-order dynamics as LLMs can recall absolute numbers for economic details, allowing look-ahead bias in the backtests [37].

Analytics

Technical indicators ($\mathcal{S}_{\text{an}}$) were computed over rolling 20-day windows using the open-source TA-Lib[7] library. These features include:

- Simple Moving Averages (SMA) over 20, 50, 100, 200 trading-day horizons,
- Relative Strength Index (RSI),
- Average True Range (ATR) for volatility,
- Moving Average Convergence Divergence (MACD) with its signal line and derived strength,
- Volume-Weighted Average Price (VWAP) as a reference anchor for intraday valuations.

Each indicator was extended with slope and z-score to assist the LLM in capturing directional shifts and the statistical significance of deviations. These technical indicators are widely used in trading practice and academic research [34], [3], [45].

Alternative Data

Structured representations of financial news headlines ($\mathcal{S}_{\text{alt}}$) were extracted using a large language model (LLM), which anonymized and synthesized the content into latent factors. Following the LLMFactor methodology [14], each news item was distilled into 2–5 interpretable factors, capturing macroeconomic and firm-specific signals.

To mitigate memorization and data leakage risks, named entities and dates were anonymized (e.g., "Tesla" becomes "the Company").

Replicated Benchmark Metrics

We report the replicated benchmark metrics in (8) for the assets used in [6]. We include the mean SR and MDD, each averaged across 25 runs with standard deviation σ.

For the SR, we conduct a two-sided one-sample t-test to assess whether the metric is significantly different from the published value. The null hypothesis H_0 assumes equivalence: $H_0 : \mu_{\text{SR}} = \text{SR}_{\text{paper}}$. The alternative hypothesis H_1 tests for deviation: $H_1 : \mu_{\text{SR}} \neq \text{SR}_{\text{paper}}$.

Since this is a replication test, rejecting H_0 indicates successful replication. p-values are computed only for SR; other metrics are reported without significance testing.

All assets have been successfully replicated within acceptable bounds, with exceptions highlighted in bold. Notably, GOOGL, one of the stocks included in our test environment, exhibited a statistically significant deviation from the original benchmark, with a p-value below 0.05.

[7]https://ta-lib.org/

Instrument	Paper SR	SR ($\pm\sigma$) [p]	MDD ($\pm\sigma$)
AB InBev	0.187	**1.21 (0.3) [0.00]**	0.18 (0.08)
Alibaba	0.021	**0.06 (0.02) [0.00]**	0.09 (0.01)
Amazon	0.419	0.39 (0.45) [0.85]	0.30 (0.09)
Apple	1.424	1.19 (0.55) [0.22]	0.29 (0.09)
Baidu	0.08	**0.20 (0.17) [0.00]**	0.36 (0.09)
CCB	0.202	**0.33 (0.25) [0.04]**	0.24 (0.14)
Coca Cola	1.068	1.07 (0.53) [0.50]	0.25 (0.04)
Dow Jones	0.684	0.70 (0.30) [0.91]	0.25 (0.05)
ExxonMobil	0.098	0.10 (0.35) [0.91]	0.34 (0.08)
FTSE 100	0.103	**0.50 (0.23) [0.00]**	0.31 (0.08)
Google	0.227	**-0.54 (0.59) [0.00]**	0.43 (0.13)
HSBC	0.011	**0.38 (0.17) [0.00]**	0.29 (0.05)
JPMorgan Chase	0.722	0.72 (0.31) [0.98]	0.26 (0.06)
Kirin	0.852	0.85 (0.42) [0.99]	0.39 (0.07)
Meta	0.151	**0.63 (0.61) [0.01]**	0.45 (0.27)
Microsoft	0.987	0.70 (1.00) [0.38]	0.28 (0.16)
NASDAQ 100	0.845	0.85 (0.35) [1.00]	0.16 (0.05)
Nikkei 225	0.019	**0.26 (0.29) [0.02]**	0.29 (0.07)
Nokia	-0.094	**0.07 (0.24) [0.00]**	0.57 (0.15)
PetroChina	0.156	0.22 (0.29) [0.29]	0.67 (0.00)
Philips	0.675	**1.40 (0.50) [0.00]**	0.25 (0.03)
S&P 500	0.834	0.83 (0.25) [1.00]	0.14 (0.04)
Shell	0.425	0.42 (0.37) [0.95]	0.51 (0.05)
Siemens	0.426	0.39 (0.23) [0.43]	0.26 (0.12)
Sony	0.424	0.42 (0.36) [0.97]	0.16 (0.04)
Tesla	0.621	0.48 (0.41) [0.29]	0.52 (0.09)
Tencent	-0.198	-0.19 (0.33) [0.98]	0.10 (0.09)
Toyota	0.304	0.36 (0.27) [0.37]	0.45 (0.10)
Volkswagen	0.216	**0.45 (0.18) [0.00]**	0.48 (0.09)

TABLE IX
REPLICATION METRICS FOR [6].