

Agent Trading Arena: A Study on Numerical Understanding in LLM-Based Agents

Tianmi Ma^{1, #}, Jiawei Du^{2, #}, Wenxin Huang^{3, ✉}, Wenjie Wang⁴, Liang Xie¹,
Xian Zhong^{1, ✉}, and Joey Tianyi Zhou²

¹ Wuhan University of Technology ² Agency for Science, Technology and Research, Singapore
³ Hubei University ⁴ University of Science and Technology of China

Correspondence: wenxinhuang_wh@163.com, zhongx@whut.edu.cn

Abstract

Large language models (LLMs) have demonstrated remarkable capabilities in natural language tasks, yet their performance in dynamic, real-world financial environments remains underexplored. Existing approaches are limited to historical backtesting, where trading actions cannot influence market prices and agents train only on static data. To address this limitation, we present the *Agent Trading Arena*, a virtual zero-sum stock market in which LLM-based agents engage in competitive multi-agent trading and directly impact price dynamics. By simulating realistic bid-ask interactions, our platform enables training in scenarios that closely mirror live markets, thereby narrowing the gap between training and evaluation. Experiments reveal that LLMs struggle with numerical reasoning when given plain-text data, often overfitting to local patterns and recent values. In contrast, chart-based visualizations significantly enhance both numerical reasoning and trading performance. Furthermore, incorporating a reflection module yields additional improvements, especially with visual inputs. Evaluations on NASDAQ and CSI datasets demonstrate the superiority of our method, particularly under high volatility. All code and data are available at <https://github.com/wekjsdvm/Agent-Trading-Arena>.

1 Introduction

In recent years, large language models (LLMs) have demonstrated strong capabilities in contextual understanding and generation across tasks such as natural language processing (Koshkin et al., 2024), data analysis, and reasoning (Olausson et al., 2023; Qiao et al., 2023). Their applications are rapidly extending to complex decision-making and interactive scenarios. LLM-based autonomous agents have emerged as a promising paradigm, showing notable progress in communication (Qian et al.,

denotes equal contribution.

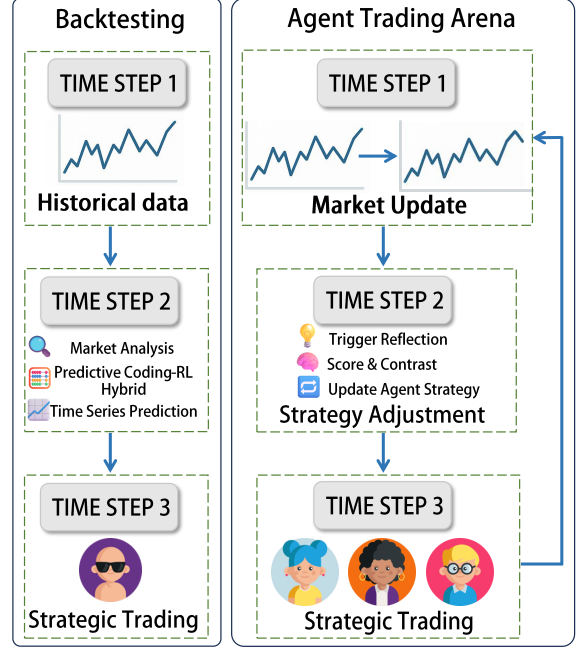


Figure 1: **Comparison between Static Backtesting and the Arena.** **Left:** Static backtesting performs one-shot analysis on historical data with fixed trading decisions, leveraging models such as Agent (market reasoning), StockFormer (predictive coding + RL hybrid), and StockMixer (time-series prediction). **Right:** The Arena, by contrast, supports real-time sensing, continuous adaptation, and feedback-driven execution for dynamic trading.

2024; Zhong et al., 2024), social simulation (Park et al., 2023, 2022; Lin et al., 2023), gaming (Wang et al., 2024a; Zhu et al., 2023), and multi-agent collaboration (Chen et al., 2023; Hu et al., 2023; Hong et al., 2024). These agents are increasingly regarded as a pathway toward general artificial intelligence, with growing potential in dynamic decision-making and interactive environments.

Despite these advances, the application of LLM-based agents in financial markets remains limited. Financial ecosystems naturally form complex multi-agent environments in which participants in-

interact through dynamic market mechanisms. However, most existing studies (Cheng and Chin, 2024; Xiao et al., 2024) rely on historical backtesting, as illustrated in Fig. 1, where agents cannot influence market prices and thus fail to capture interactivity and adaptive behavior. This gap underscores the need for a framework that supports real-time agent interactions and realistic market feedback.

To address these limitations, we present the *Agent Trading Arena*, a virtual zero-sum stock market in which LLM-based agents engage in competitive trading and directly affect price dynamics through bid-ask interactions. Our platform generates continuously evolving numerical data, forcing agents to adapt to shifting market conditions; any temporarily optimal strategy is quickly countered, rewarding adaptability and robust reasoning over memorization.

By simulating bid-ask order matching in real time, the Arena faithfully reproduces liquidity dynamics and price impact, enabling agents to experience authentic market friction and slippage. Unlike static backtests, the Arena continuously updates its state based on agents’ collective actions, creating an ever-shifting landscape that prevents overfitting to historical patterns. Because agents directly influence subsequent price trajectories, the Arena closes the loop between training and evaluation, effectively bridging the “train-test” gap of traditional financial simulations. Its modular design further supports seamless integration of new market features, such as varying transaction costs, order types, or macroeconomic shocks, making it a versatile testbed for financial AI research.

Our experiments show that LLMs struggle with numerical reasoning when processing plain-text data: they focus on absolute values, overlook percentage changes and relational patterns, and overemphasize recent trends even when earlier data are emphasized. In contrast, visual representations, such as line charts, bar graphs, and scatter plots, substantially improve both numerical reasoning and trading performance. As shown in Table 5, chart-based inputs lead to significantly higher return rates in the *Agent Trading Arena*.

We further integrate a reflection module (Zhou et al., 2022; Shinn et al., 2023), which amplifies the benefits of visual data by enabling agents to iteratively refine their reasoning. This module is particularly effective with structured visual inputs, facilitating deeper pattern recognition and more strategic decision-making.

Finally, we validate our approach on real-world datasets from NASDAQ and CSI, demonstrating consistent performance gains over baselines, particularly under high-volatility market regimes.

Our contributions are summarized threefold:

- We design the *Agent Trading Arena*, a dynamic and interactive simulation framework that moves beyond static backtesting with zero-sum multi-agent competition, enabling realistic evaluation of LLMs’ reasoning and adaptability.
- We uncover LLMs’ limitations in textual numerical reasoning, highlighting difficulties with relational patterns, percentage changes, and long-range dependencies, while demonstrating superior performance with structured visual data.
- We demonstrate that a reflection module enhances LLMs’ reasoning over complex visual inputs, leading to more accurate and strategic trading decisions.

2 Related Works

2.1 LLM-Based Agents in Dynamic Decision-Making Environments

LLMs serve as the core reasoning engines in autonomous agents, enabling planning, tool use, reflection, and adaptation within dynamic environments. Advances in NLP have empowered models such as ChatGPT (OpenAI, 2023), PaLM (Anil et al., 2023), and LLaMA (Touvron et al., 2023) to operate via natural language, making them particularly effective in interactive, text-based scenarios. Frameworks like ChatDev (Qian et al., 2024), AgentVerse (Chen et al., 2023), and MetaGPT (Hong et al., 2024) demonstrate multi-agent collaboration on software development tasks, while Voyager (Wang et al., 2024a), GROOT (Cai et al., 2024), and Optimus-1 (Li et al., 2024b) showcase autonomous exploration in open-ended environments such as Minecraft. Social simulators, including Generative Agents (Park et al., 2023) and AgentSims (Lin et al., 2023), further reveal emergent behaviors through sustained multi-agent interactions.

Together, these studies demonstrate that LLM-based agents can effectively coordinate, communicate, and make decisions in complex, evolving environments. Their adaptability and collaborative strengths highlight strong potential for real-time

decision-making in finance, an information-dense and competitive domain analogous to the dynamic settings where LLMs excel.

2.2 LLMs in Financial Analysis and Decision Making

LLMs have been applied to sentiment analysis, financial question answering, and investment recommendation. Benchmarks such as SocioDojo (Cheng and Chin, 2024) evaluate financial understanding but lack interactive components. Multi-agent systems like StockAgent (Zhang et al., 2024a), TradingAgents (Xiao et al., 2024), and CryptoTrade (Li et al., 2024a) enhance performance through role specialization and iterative reflection. In contrast, traditional forecasting models, *e.g.*, StockFormer (Gao et al., 2023), TimesNet (Wu et al., 2023), PatchTST (Nie et al., 2023), PDFormer (Jiang et al., 2023), StockMixer (Fan and Shen, 2024), and iTransformer (Liu et al., 2024b), focus on price prediction using advanced methods such as wavelet transforms and temporal modeling to capture market dynamics.

However, neither paradigm addresses the “live market” paradox: without real-time adaptive mechanisms, these systems cannot simulate how participants adjust strategies in response to multi-agent interactions, news shocks, and evolving order flows. This limitation significantly reduces their practical utility in real trading environments.

3 Agent Trading Arena

To overcome the limitations of static backtesting and capture dynamic interactions among market participants, we propose the *Agent Trading Arena*, a closed-loop, zero-sum economic system that simulates complex quantitative scenarios in real markets (Guo et al., 2024). The overall architecture is shown in Fig. 2 and detailed in Appendix A. Within this environment, agents invest in assets, earn dividends on their holdings, and pay daily expenses using virtual currency.

3.1 Virtual Market Environment

Closed-Loop System. The *Agent Trading Arena* is a closed-loop virtual economy that isolates external knowledge and avoids reliance on historical data, enabling fair evaluation of agent capabilities. Unlike traditional backtesting grounded in past market records, this sandboxed system removes real-world anchors: all assets are abstract

symbols, and their prices emerge solely from agent interactions through a bid-ask mechanism. By eliminating external context, the environment compels agents to adapt strategies based on internal dynamics and opponent behavior, fostering genuine decision-making and strategic evolution.

Bid-Ask Mechanism. Asset prices arise exclusively from a bid-ask process: each agent’s orders directly influence the market price, with no external benchmarks. This closed-loop design rigorously captures endogenous price formation, faithfully modeling liquidity constraints, order book dynamics, market depth, and slippage. Every execution reshapes the market microstructure, producing non-trivial price impacts and path-dependent outcomes that evolve with aggregate agent behavior over time.

3.2 Chat Pool

To simulate real-world information flow, we introduce a shared *Chat Pool* where agents post and read “news” items, analyst commentary, and social media-style opinions. By reproducing both informative signals and noisy interference, including hallucinations (Gekhman et al., 2024) and misinformation, the *Chat Pool* reflects how public information can shape, and at times distort, traders’ narratives and expectations. This process increases the complexity of strategic reasoning and adaptive behavior in the market.

3.3 Market Mechanisms

We incorporate two key incentives to sustain active trading and realistic dynamics: 1) *Dividends & Capital Gains*: Agents earn income from both price appreciation and dividends, distributed proportionally to holdings. This implicitly anchors prices and rewards low-cost positions. 2) *Daily Wealth Fee*: A cost proportional to each agent’s total wealth is charged daily. Only those earning sufficient dividends can cover this fee, which discourages passive holding and promotes frequent, strategic trades.

Together, these mechanisms maintain continuous market activity, encourage adaptive decision-making, and prevent reliance on static or exploitative strategies in a zero-sum environment.

3.4 Zero-Sum Competition

By design, the Arena is zero-sum: the profit of one agent is the loss of another, meaning there is

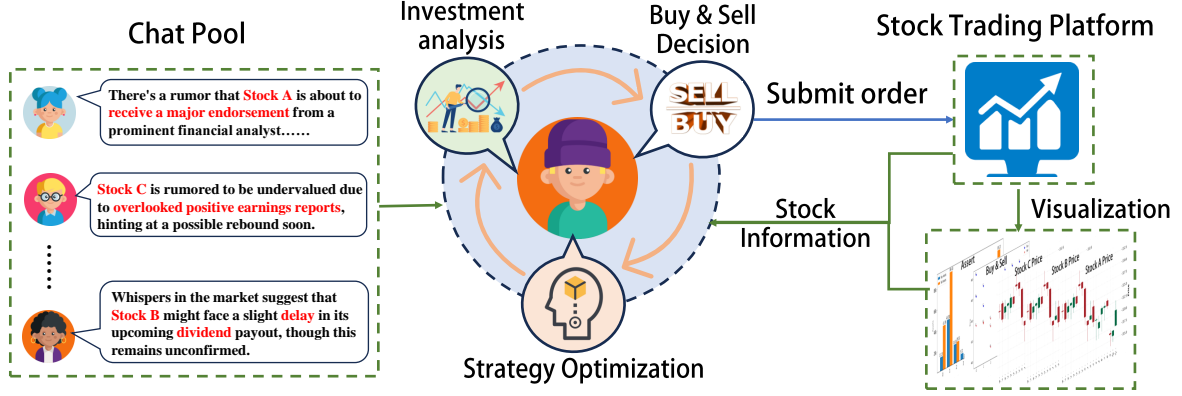


Figure 2: **Overall architecture of the Agent Trading Arena, consisting of three components:** the *Chat Pool* (left) for inter-agent communication, agent modules (center) for analysis, decision-making, and reflection, and the trading platform (right), where agents’ buy/sell actions dynamically determine asset prices in a closed-loop market simulation.

no fixed “best” strategy. Competitors counter any temporarily successful tactic, compelling agents to learn implicit rules, interpret delayed numerical feedback, and continuously adapt in real time. With results unfolding gradually and misinformation potentially introduced by rivals, agents must rely on experiential learning and develop context-aware strategies through ongoing adaptation rather than predefined rules or external knowledge.

4 Proposed Method

Our agent, ArenaTrader, integrates textual and visual modalities to form a comprehensive understanding of market conditions. It employs a structured three-stage framework, market analysis, trade execution, and strategic reflection, to leverage this multi-modal representation. This design enables informed and adaptive decision-making based on data-driven insights and iterative feedback.

4.1 Numerical Data Representations

Textual Input. Time-series stock data are represented as plain text, including timestamps, prices, and volumes, to provide explicit semantics (Hu et al., 2024; Zhang et al., 2024b). However, LLMs often fixate on absolute values, overlook percentage changes, and overemphasize recent entries, which limits their ability to reason across long horizons (see lower right of Fig. 3).

Visual Input. To complement textual data, we introduce visualizations such as line charts, bar graphs, and scatter plots (Yang et al., 2025), which highlight global trends, relational patterns, and

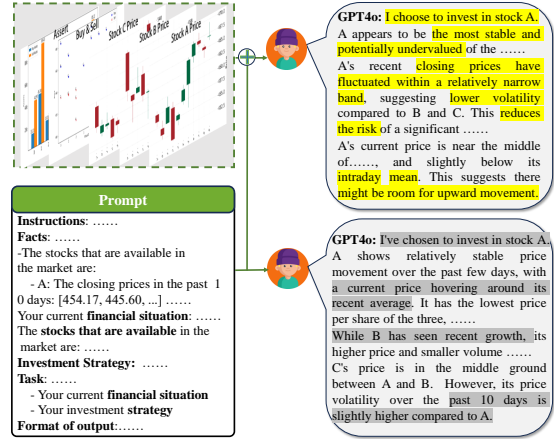


Figure 3: **Textual vs. Visual Input-Output Representations.** **Left:** Agent buy/sell records, daily trade prices, and corresponding candlestick charts. **Top right:** Visual-input outputs highlight overall trends and long-term patterns. **Bottom right:** Textual-input outputs emphasize specific price points.

volatility at a glance (see upper right of Fig. 3). These visualizations are consistent with the textual input, ensuring a cohesive representation. This structured format enables LLMs to integrate local details with broader market context more effectively.

4.2 Agent Architecture

Investment Analysis. Inspired by human trading behavior, each agent begins by analyzing market conditions before executing trades. This process involves extracting and synthesizing multi-modal signals from the *Chat Pool*, price histories, and fun-

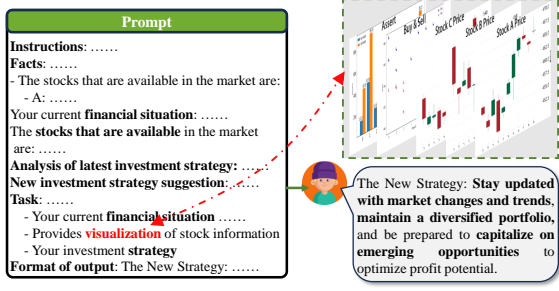


Figure 4: **Design of the Reflection Module.** **Left:** Agent trading records, strategy evaluations, and top- and bottom-performing strategies from the strategy library. **Top right:** Corresponding stock chart. **Bottom right:** Newly generated strategy based on current information.

damental indicators. By combining these heterogeneous inputs, the agent constructs a stock ranking framework based on predefined heuristics or learned criteria, allowing it to identify promising assets for portfolio allocation.

Action Decision. Agents execute multiple trades per day, making buy, sell, or hold decisions by integrating diverse signals, including analysis reports, real-time market data, and historical performance trends. They evaluate volatility, price momentum, and portfolio risk exposure to compute optimal trade volumes under capital constraints. Orders are executed via the bid-ask mechanism, with portfolios updated accordingly. Each decision aims to maximize long-term returns while maintaining a competitive edge in the evolving virtual market.

Strategy Optimization via Reflection. At the end of each trading day, the reflection module distills daily experiences into bidirectional learning signals (see Fig. 4). It first evaluates the effectiveness of trading trajectories and strategies using a scoring function. Then, by contrasting top- and bottom-performing strategies, agents generate refined action plans and archive unsuccessful tactics in a strategy library. This iterative process drives continuous strategy evolution and more robust decision-making.

5 Experimental Results

5.1 Experimental Setup

Environment. The *Agent Trading Arena* is lightweight and accessible, requiring only API credits and no GPU for deployment. Details on the selection of different LLMs are provided in

ID	Name	Duration (years)	Initial Capital	Profession
1	Amy	1	100,000	AI Researcher
2	Bruce	2	100,000	Lawyer
3	Charles	1	100,000	Doctor
4	David	3	100,000	Engineer
5	Ella	2	100,000	Teacher
6	Frank	5	100,000	Entrepreneur
7	Grace	4	100,000	Accountant
8	Hank	2	100,000	Architect
9	Ivy	3	100,000	Marketing Manager
...

Table 1: **Agent Characteristics.** Duration denotes each agent’s investment horizon, and Initial Capital indicates the starting funds for trading.

ID	Ticker	DPS	Closing Prices	Initial Quantity
1	A	22	454.17, ..., 445.60	1,200
2	B	23	354.17, ..., 465.80	1,000
3	C	25	500.47, ..., 440.60	1,600
...

Table 2: **Stock Details.** DPS denotes dividend per share; Closing Prices are historical end-of-day closing prices; and Initial Quantity specifies the starting number of shares.

Appendix D.

Datasets. We configure the Arena with varying numbers of agents and stocks, all initialized with the same capital (see Tables 1 and 2). To validate our agents, we simulate portfolio investments on subsets of NASDAQ and CSI datasets. For NASDAQ, we run a two-month simulation (September 3–October 29, 2024) with an initial capital of 100,000 units, excluding weekends and holidays. For CSI, we conduct a one-year simulation (January 2–December 31, 2024) with the same capital, also excluding non-trading days. Both datasets, collected from Yahoo Finance and Baostock, provide daily records of opening, closing, high, and low prices, trading volume, and associated technical indicators, offering a holistic and up-to-date depiction of market behavior. Additional details are given in Appendix B.

Baselines. We compare ArenaTrader against the following strategies:

1) **Buy & Hold:** A passive strategy that purchases assets initially and holds them throughout the period without further trading.

2) **SMA (Gencay, 1996):** A moving average crossover strategy used to detect bullish and bearish momentum. We evaluate window sizes [5, 10, 15,

20] and select the best based on validation results.

3) ZMR (Poterba and Summers, 1988): A zone-based reversal strategy that trades on deviations from preset bounds, with thresholds and holding periods tuned via validation.

4) MACD (Chong and Ng, 2008): A momentum strategy based on MACD and signal line crossovers, where MACD is the 12-day EMA minus the 26-day EMA, and the signal line is the 9-day EMA.

5) StockFormer (Gao et al., 2023): A hybrid model combining predictive coding and reinforcement learning to capture future dynamics and asset correlations. We follow the default setup from the original implementation.

6) TimesNet (Wu et al., 2023): Transforms 1D time series into 2D representations to model multi-period temporal variations. We follow the original setup and select the look-back window from [5, 10, 15, 20] based on validation.

7) StockMixer (Fan and Shen, 2024): A lightweight MLP that mixes indicators, time, and stocks for efficient modeling of market dynamics. We follow the settings specified in the original work.

Metrics. We evaluate using five metrics:

1) Total Return (TR): $(C_1 - C_0)/C_0$, where C_0 and C_1 are initial and final capital.

2) Win Rate (WR): N_w/N_t , the ratio of profitable trading days to total trading days.

3) Sharpe Ratio (SR): $(R_p - R_f)/\sigma_p$, where R_p is the mean daily return, σ_p is the standard deviation of daily returns, and R_f is set to 0 to emphasize risk-adjusted returns.

4) Mean Daily Return (Mean): Average daily return over the simulation.

5) Return Volatility (Std): Standard deviation of daily returns, measuring risk.

5.2 Experimental Results

In the *Agent Trading Arena*, we configure multiple agents and LLM backbones to promote diversity. The primary agent, ArenaTrader, consistently uses our full-featured design, including visual perception and reflection, enabling focused analysis. We evaluate ArenaTrader on real-world datasets, with corresponding prompts detailed in Appendix C.

Vision-Enhanced Agent Performance. Table 3 and Fig. 5 report the returns of ArenaTrader under different vision-capable LLMs. GPT-4o (Hurst et al., 2024) achieves the highest TR, followed

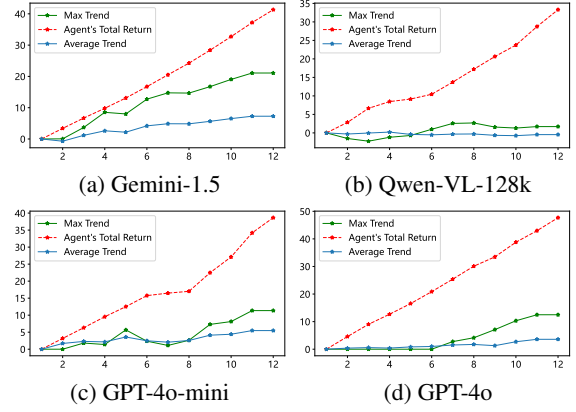


Figure 5: ArenaTrader Performance vs. Market Volatility. Total return of ArenaTrader in the *Agent Trading Arena*, compared with the most volatile single stock and the average return across all stocks. This comparison highlights the agent’s robustness under varying market conditions.

by Gemini-1.5 (Reid et al., 2024). For volatility-adjusted returns, GPT-4o again leads, with Qwen-VL-32k (Wang et al., 2024b) second. These results highlight GPT-4o’s strong data-driven reasoning and robustness across varying volatility regimes, underscoring its ability to integrate visual and numerical signals in dynamic markets.

Evaluation on Real-World Data. Using GPT-4o as the backbone, Table 4 presents results on NASDAQ and CSI. Training details for StockFormer (Gao et al., 2023), TimesNet (Wu et al., 2023), and StockMixer (Fan and Shen, 2024) are provided in Appendix B. Despite extensive training, these models underperform our method, which uses a 10-day input window selected via ablation. The Sharpe Ratio (SR) of NASDAQ-100 benchmark during the same period is 0.189, while CSI 300 records 0.054. In contrast, our model achieves 0.348 on NASDAQ-100 (an improvement of 0.159) and 0.123 on CSI 300 (an improvement of 0.069).

Building on these results, Fig. 6 provides a detailed analysis of portfolio trajectories and market conditions, illustrating our agent’s near-optimal performance, especially during high-volatility periods. Notably, our approach operates without task-specific retraining, relying solely on historical price data. This design underscores the robustness and strong generalization ability of our agent in real-world, dynamic market environments.

Model	Avg. Trend	TR (%) \uparrow	Mean (%) \uparrow	Std (%) \downarrow	WR (%) \uparrow	SR \uparrow	Δ (%) \uparrow
ArenaTrader w/ Qwen-VL-128k	-0.4632	33.3018	2.6526	1.0150	100.00	2.6135	33.7650
ArenaTrader w/ GPT-4o-mini	5.4722	38.6460	3.0249	1.4235	100.00	2.1249	33.1738
ArenaTrader w/ Gemini-1.5	7.2743	<u>41.3264</u>	<u>3.1946</u>	0.1588	100.00	20.1128	34.0521
ArenaTrader w/ Qwen-VL-32k	-0.5216	35.8150	3.1213	1.6079	90.00	1.9412	<u>36.3366</u>
ArenaTrader w/ GPT-4o	3.5881	47.6851	3.6095	<u>0.5327</u>	100.00	<u>6.7765</u>	44.0970

Table 3: **Investment Performance of ArenaTrader with Different LLMs.** Avg. Trend denotes the average market trend, and Δ indicates the gain relative to this trend. GPT-4o achieves the highest returns and stability, demonstrating superior data-driven reasoning among vision-capable LLMs. Bold and underlined values denote the best and second-best results, respectively.

Strategy	NASDAQ					CSI				
	TR \uparrow	Mean \uparrow	Std \downarrow	WR \uparrow	SR \uparrow	TR \uparrow	Mean \uparrow	Std \downarrow	WR \uparrow	SR \uparrow
Buy & Hold	6.06	0.155	0.916	61.54	0.169	0.73	0.004	0.403	43.15	0.010
SMA	0.19	0.006	<u>0.561</u>	46.15	0.011	-3.45	-0.014	0.319	33.20	-0.044
ZMR	1.04	0.030	<u>0.814</u>	41.03	0.037	1.24	0.006	<u>0.361</u>	41.91	0.016
MACD	5.62	0.146	1.033	58.97	0.141	-0.67	-0.002	0.373	40.66	-0.006
StockFormer (Gao et al., 2023)	11.23	0.281	0.121	56.41	0.231	14.83	0.636	1.128	41.08	0.056
TimesNet (Wu et al., 2023)	11.41	0.283	1.035	56.41	0.273	32.93	0.139	2.053	<u>48.55</u>	0.068
StockMixer (Fan and Shen, 2024)	<u>14.64</u>	<u>0.357</u>	1.173	56.41	<u>0.305</u>	<u>43.00</u>	0.153	0.907	27.80	0.168
ArenaTrader w/ GPT-4o (Ours)	15.99	0.450	1.110	<u>58.97</u>	0.348	48.93	<u>0.176</u>	1.432	51.45	<u>0.123</u>

Table 4: **Strategy Performance on NASDAQ and CSI.** Our method demonstrates strong robustness and generalization, achieving near-optimal performance without additional training.

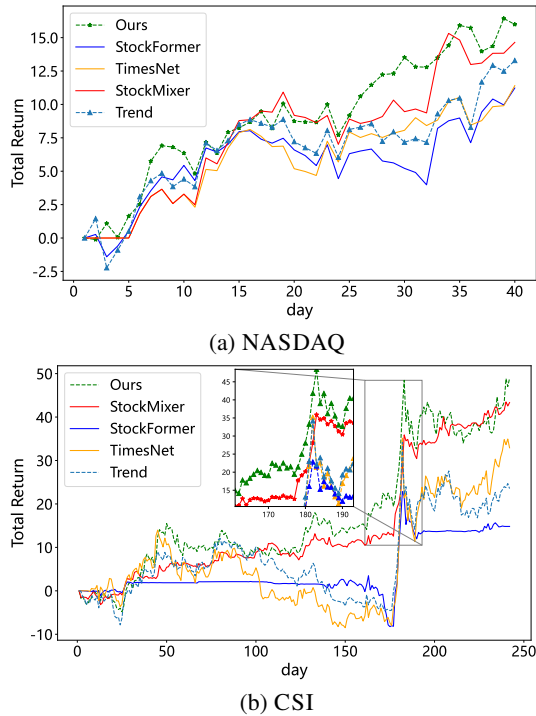


Figure 6: **Backtesting Results on Real-World Stock Datasets.** Two-month backtest on NASDAQ and one-year backtest on CSI. The proposed method consistently delivers near-optimal performance and outperforms all baselines, particularly during periods of high market volatility.

5.3 Ablation Studies

We conduct ablation studies in the Arena to evaluate the impact of input formats and the reflection mechanism on agent performance. To further assess the effect of sequence length across textual and visual modalities, we perform controlled backtesting on NASDAQ. Backtesting mitigates dynamic data fluctuations, allowing sequence length to be isolated as the primary variable.

Textual vs. Visual Inputs. We compare agents using textual-only inputs with those leveraging visualizations (scatter plots, line charts, bar graphs). Non-vision LLMs receive text only without reflection, while vision-capable LLMs are evaluated under text-only, visual-only, and combined-input settings. Corresponding prompts are provided in Appendix C, and details of LLM choices are in Appendix D. Results in Table 5 show that visual inputs substantially improve returns, with combined inputs achieving the best overall performance.

Impact of Reflection. Incorporating the reflection module further enhances performance across all LLMs. For GPT-4o, TR improves from 33.65% with text-only to 47.70% with visual input, a relative gain of 41.7%. Beyond TR, reflection also boosts SR and WR, indicating higher profitability, stability, and consistency in trading outcomes.

	Model	Textual	Visual	TR \uparrow	Mean \uparrow	Std \downarrow	WR \uparrow	SR \uparrow
w/o Reflection	LLaMa-3 (Dubey et al., 2024)	●	○	4.09	0.686	2.270	62.50	0.196
	DeepSeek (Liu et al., 2024a)	●	○	27.31	2.248	2.755	92.21	1.530
	Qwen-2.5 (Yang et al., 2024)	●	○	30.37	2.680	1.459	93.75	1.703
	Gemini-1.5 (Reid et al., 2024)	●	○	14.32	1.221	2.778	83.84	1.729
		○	●	19.04	1.604	1.165	90.91	1.376
		●	●	23.96	1.981	1.334	100.00	1.485
	GPT-4o (Hurst et al., 2024)	●	○	13.04	1.139	1.929	54.55	0.590
		○	●	17.07	1.454	1.504	72.73	0.967
		●	●	26.18	2.157	2.058	90.91	1.048
w/ Reflection	LLaMa-3 (Dubey et al., 2024)	●	○	10.25	1.667	2.403	66.67	0.694
	DeepSeek (Liu et al., 2024a)	●	○	30.62	2.470	1.512	100.00	1.633
	Qwen-2.5 (Yang et al., 2024)	●	○	38.91	<u>3.324</u>	1.073	100.00	3.099
	Gemini-1.5 (Reid et al., 2024)	●	○	29.45	2.519	3.012	93.51	2.210
		○	●	37.01	2.911	1.235	100.00	2.357
		●	●	<u>41.33</u>	3.195	0.159	100.00	20.113
	GPT-4o (Hurst et al., 2024)	●	○	33.65	2.671	2.171	98.70	2.142
		○	●	35.76	2.821	0.678	100.00	4.159
		●	●	47.69	3.610	<u>0.533</u>	100.00	<u>6.777</u>

Table 5: **Performance Comparison with and without Reflection Using Textual and Visual Inputs.** Agents using visual data consistently outperform those using text alone, while combining both modalities yields the strongest results.

Input	Window	TR \uparrow	Mean \uparrow	Std \downarrow	WR \uparrow	SR \uparrow
Textual	5 days	3.63	0.102	1.588	62.5	0.064
	10 days	5.58	0.145	1.334	60.0	0.109
	15 days	6.47	0.171	1.681	57.5	0.102
	20 days	1.67	0.050	1.296	62.5	0.039
Visual	5 days	8.21	0.207	1.396	57.5	0.149
	10 days	6.90	0.174	1.164	67.5	0.149
	15 days	6.90	0.174	1.216	60.0	0.143
	20 days	2.81	0.075	1.101	62.5	0.068
Textual + Visual	5 days	14.33	0.372	1.719	60.0	0.216
	10 days	15.99	0.450	<u>1.110</u>	59.0	0.348
	15 days	<u>14.73</u>	<u>0.380</u>	1.684	57.5	<u>0.226</u>
	20 days	13.51	0.352	1.577	<u>65.0</u>	0.223

Table 6: **Performance with Textual and Visual Inputs Across Different Window Sizes.** Window denotes the number of consecutive trading days provided as historical input before the current trading day. Textual + Visual consistently achieves the best performance, particularly with a 10-day window, highlighting the importance of both modality choice and time-window design.

These findings confirm that strategic reflection, particularly with visual data, strengthens agents’ reasoning and decision-making under volatile market conditions.

Impact of Time-Series Length. Experiments with GPT-4o as the backbone are summarized in Table 6, which reports performance across modalities and time windows. Overall, “Textual + Visual” consistently outperforms single modalities, particu-

larly with a 10-day window, achieving the highest TR (15.99%), SR (0.348), and Mean (0.450%). Visual inputs perform best in the 5–10 day range, offering strong returns, stability, and win rates. Performance declines at 20 days, likely due to noise accumulation or overfitting. These results underscore the importance of both multi-modal fusion and careful time-window selection for robust time-series decision-making.

6 Conclusion

We presented the *Agent Trading Arena*, a closed-loop, zero-sum simulation framework designed to model complex market dynamics and evaluate LLMs on numerical reasoning tasks. Our experiments show that while LLMs struggle with plain-text numerical data, they achieve significantly higher returns when supported by visual representations, underscoring the importance of charts and graphs in complex decision-making. Integrating a reflection module further improves performance by enabling iterative analysis and strategic refinement. Validation on both NASDAQ and CSI confirms that vision-enhanced LLM agents consistently outperform text-only counterparts without additional training. Overall, this work highlights both the capabilities and limitations of LLMs in numerical reasoning and establishes the *Agent Trading Arena* as a robust testbed for advancing research in adaptive, real-world financial applications.

Limitations

This study investigates the visual reasoning abilities of LLMs in a virtual stock trading environment. While such a controlled setup enables systematic analysis, it inevitably simplifies many aspects of real-world financial systems, including uncertainty, feedback loops, and human interactions. Bridging the gap between virtual environments and real-world applications therefore remains a significant challenge.

A promising direction for future work is to explore how agents can be gradually integrated into real financial workflows, interacting with live data, adapting to real-time signals, and contributing meaningful decisions or strategies. This would move beyond passive evaluation toward active participation, where agents not only respond to environments but also generate novel behavioral data and insights to inform downstream analysis, model training, and system design. Expanding the task scope and incorporating more diverse modalities may further support the development of generalizable and robust evaluation frameworks.

Acknowledgments

This work was supported in part by the National Natural Science Foundation of China under Grants 62301213 and 62271361, the Hubei Provincial Key Research and Development Program under Grant 2024BAB039, and the Open Project Funding of the Hubei Key Laboratory of Big Data Intelligent Analysis and Application, Hubei University, under Grant 2024BDIAA01.

References

- Rohan Anil, Andrew M Dai, Orhan Firat, Melvin Johnson, Dmitry Lepikhin, Alexandre Passos, Siamak Shakeri, Emanuel Taropa, Paige Bailey, Zhifeng Chen, et al. 2023. Palm 2 technical report. *arXiv:2305.10403*.
- Shaofei Cai, Bowei Zhang, Zihao Wang, Xiaojian Ma, Anji Liu, and Yitao Liang. 2024. GROOT: learning to follow instructions by watching gameplay videos. In *Proc. Int. Conf. Learn. Represent.*
- Weize Chen, Yusheng Su, Jingwei Zuo, Cheng Yang, Chenfei Yuan, Chen Qian, Chi-Min Chan, Yujia Qin, Yaxi Lu, Ruobing Xie, et al. 2023. Agentverse: Facilitating multi-agent collaboration and exploring emergent behaviors in agents. *arXiv:2308.10848*.
- Junyan Cheng and Peter Chin. 2024. SocioDojo: Building lifelong analytical agents with real-world text and time series. In *Proc. Int. Conf. Learn. Represent.*
- Terence Tai-Leung Chong and Wing-Kam Ng. 2008. Technical analysis and the london stock exchange: Testing the macd and rsi rules using the ft30. *Appl. Econ. Lett.*, pages 1111–1114.
- Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. 2024. The LLaMa 3 herd of models. *arXiv:2407.21783*.
- Jinyong Fan and Yanyan Shen. 2024. StockMixer: A simple yet strong MLP-Based architecture for stock price forecasting. In *Proc. AAAI Conf. Artif. Intell.*, pages 8389–8397.
- Siyu Gao, Yunbo Wang, and Xiaokang Yang. 2023. StockFormer: Learning hybrid trading machines with predictive coding. In *Proc. Int. Joint Conf. Artif. Intell.*, pages 4766–4774.
- Zorik Gekhman, Gal Yona, Roei Aharoni, Matan Eyal, Amir Feder, Roi Reichart, and Jonathan Herzig. 2024. Does fine-tuning LLMs on new knowledge encourage hallucinations? In *Proc. Conf. Empir. Methods Nat. Lang. Process.*, pages 7765–7784.
- Ramazan Gencay. 1996. Non-linear prediction of security returns with moving average rules. *J. Forecast.*, 15(3):165–174.
- Shangmin Guo, Haoran Bu, Haochuan Wang, Yi Ren, Dianbo Sui, Yuming Shang, and Siting Lu. 2024. Economics arena for large language models. *arXiv:2401.01735*.
- Sirui Hong, Mingchen Zhuge, Jonathan Chen, Xiawu Zheng, Yuheng Cheng, Jinlin Wang, Ceyao Zhang, Zili Wang, Steven Ka Shing Yau, Zijuan Lin, Liyang Zhou, Chenyu Ran, Lingfeng Xiao, Chenglin Wu, and Jürgen Schmidhuber. 2024. Metagpt: Meta programming for A multi-agent collaborative framework. In *Proc. Int. Conf. Learn. Represent.*
- Chenxu Hu, Jie Fu, Chenzhuang Du, Simian Luo, Junbo Zhao, and Hang Zhao. 2023. Chatdb: Augmenting llms with databases as their symbolic memory. *arXiv:2306.03901*.
- Yebowen Hu, Kaiqiang Song, Sangwoo Cho, Xiaoyang Wang, Hassan Foroosh, Dong Yu, and Fei Liu. 2024. Sportsmetrics: Blending text and numerical data to understand information fusion in LLMs. In *Proc. Annu. Meet. Assoc. Comput. Linguist.*, pages 267–278.
- Aaron Hurst, Adam Lerer, Adam P Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, et al. 2024. GPT-4o system card. *arXiv:2410.21276*.
- Jiawei Jiang, Chengkai Han, Wayne Xin Zhao, and Jingyuan Wang. 2023. PDFFormer: Propagation delay-aware dynamic long-range transformer for traffic flow prediction. In *Proc. AAAI Conf. Artif. Intell.*, volume 37, pages 4365–4373.

- Roman Koshkin, Katsuhito Sudoh, and Satoshi Nakamura. 2024. TransLLaMa: LLM-based simultaneous translation system. In *Proc. Conf. Empir. Methods Nat. Lang. Process.*, pages 461–476.
- Yuan Li, Bingqiao Luo, Qian Wang, Nuo Chen, Xu Liu, and Bingsheng He. 2024a. CryptoTrade: A reflective LLM-based agent to guide zero-shot cryptocurrency trading. In *Proc. Conf. Empir. Methods Nat. Lang. Process.*, pages 1094–1106.
- Zaijing Li, Yuquan Xie, Rui Shao, Gongwei Chen, Dongmei Jiang, and Liqiang Nie. 2024b. Optimus-1: Hybrid multimodal memory empowered agents excel in long-horizon tasks. In *Adv. Neural Inf. Process. Syst.*
- Jiaju Lin, Haoran Zhao, Aochi Zhang, Yiting Wu, Huqiyue Ping, and Qin Chen. 2023. Agentsims: An open-source sandbox for large language model evaluation. *arXiv:2308.04026*.
- Aixin Liu, Bei Feng, Bing Xue, Bingxuan Wang, Bochao Wu, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenyu Zhang, Chong Ruan, Damai Dai, Daya Guo, Dejian Yang, Deli Chen, Dongjie Ji, Erhang Li, Fangyun Lin, Fucong Dai, Fuli Luo, Guangbo Hao, Guanting Chen, Guowei Li, H. Zhang, Han Bao, Hanwei Xu, Haocheng Wang, Haowei Zhang, Honghui Ding, Huajian Xin, Huazuo Gao, Hui Li, Hui Qu, J. L. Cai, Jian Liang, Jianzhong Guo, Jiaqi Ni, Jiashi Li, Jiawei Wang, Jin Chen, Jingchang Chen, Jingyang Yuan, Junjie Qiu, Junlong Li, Junxiao Song, Kai Dong, Kai Hu, Kaige Gao, Kang Guan, Kexin Huang, Kuai Yu, Lean Wang, Lecong Zhang, Lei Xu, Leyi Xia, Liang Zhao, Litong Wang, Liyue Zhang, Meng Li, Miaojun Wang, Mingchuan Zhang, Minghua Zhang, Minghui Tang, Mingming Li, Ning Tian, Panpan Huang, Peiyi Wang, Peng Zhang, Qiancheng Wang, Qihao Zhu, Qinyu Chen, Qiushi Du, R. J. Chen, R. L. Jin, Ruiqi Ge, Ruisong Zhang, Ruizhe Pan, Runji Wang, Runxin Xu, Ruoyu Zhang, Ruyi Chen, S. S. Li, Shanghao Lu, Shangyan Zhou, Shanhuang Chen, Shaoqing Wu, Shengfeng Ye, Shengfeng Ye, Shirong Ma, Shiyu Wang, Shuang Zhou, Shuiping Yu, Shunfeng Zhou, Shuting Pan, T. Wang, Tao Yun, Tian Pei, Tianyu Sun, W. L. Xiao, and Wangding Zeng. 2024a. DeepSeek-V3 technical report. *arXiv: 2412.19437*.
- Yong Liu, Tengge Hu, Haoran Zhang, Haixu Wu, Shiyu Wang, Lintao Ma, and Mingsheng Long. 2024b. itransformer: Inverted transformers are effective for time series forecasting. In *Proc. Int. Conf. Learn. Represent.*
- Yuqi Nie, Nam H. Nguyen, Phanwadee Sinthong, and Jayant Kalagnanam. 2023. A time series is worth 64 words: Long-term forecasting with transformers. In *Proc. Int. Conf. Learn. Represent.*
- Theo Olausson, Alex Gu, Benjamin Lipkin, Cedegao E. Zhang, Armando Solar-Lezama, Joshua B. Tenenbaum, and Roger Levy. 2023. LINC: A neurosymbolic approach for logical reasoning by combining language models with first-order logic provers. In *Proc. Conf. Empir. Methods Nat. Lang. Process.*, pages 5153–5176.
- OpenAI. 2023. GPT-4 technical report. *arXiv:2303.08774*.
- Joon Sung Park, Joseph C. O’Brien, Carrie Jun Cai, Meredith Ringel Morris, Percy Liang, and Michael S. Bernstein. 2023. Generative agents: Interactive simulacra of human behavior. In *Proc. ACM Symp. User Interface Softw. Technol.*, pages 2:1–2:22.
- Joon Sung Park, Lindsay Popowski, Carrie Cai, Meredith Ringel Morris, Percy Liang, and Michael S. Bernstein. 2022. Social simulacra: Creating populated prototypes for social computing systems. In *Proc. ACM Symp. User Interface Softw. Technol.*, pages 1–18.
- James M Poterba and Lawrence H Summers. 1988. Mean reversion in stock prices: Evidence and implications. *J. Financ. Econ.*, 22(1):27–59.
- Chen Qian, Wei Liu, Hongzhang Liu, Nuo Chen, Yufan Dang, Jiahao Li, Cheng Yang, Weize Chen, Yusheng Su, Xin Cong, Juyuan Xu, Dahai Li, Zhiyuan Liu, and Maosong Sun. 2024. Chatdev: Communicative agents for software development. In *Proc. Annu. Meeting Assoc. Comput. Linguist.*, pages 15174–15186.
- Shuofei Qiao, Yixin Ou, Ningyu Zhang, Xiang Chen, Yunzhi Yao, Shumin Deng, Chuanqi Tan, Fei Huang, and Huajun Chen. 2023. Reasoning with language model prompting: A survey. In *Proc. Annu. Meet. Assoc. Comput. Linguist.*, pages 5368–5393.
- Machel Reid, Nikolay Savinov, Denis Teplyashin, Dmitry Lepikhin, Timothy P. Lillicrap, Jean-Baptiste Alayrac, Radu Soricut, Angeliki Lazaridou, Orhan Firat, Julian Schrittwieser, Ioannis Antonoglou, Rohan Anil, Sebastian Borgeaud, Andrew M. Dai, Katie Millican, Ethan Dyer, Mia Glaese, Thibault Sottiaux, Benjamin Lee, Fabio Viola, Malcolm Reynolds, Yuanzhong Xu, James Molloy, Jilin Chen, Michael Isard, Paul Barham, Tom Hennigan, Ross McIlroy, Melvin Johnson, Johan Schalkwyk, Eli Collins, Eliza Rutherford, Erica Moreira, Kareem Ayoub, Megha Guel, Clemens Meyer, Gregory Thornton, Zhen Yang, Henryk Michalewski, Zaheer Abbas, Nathan Schucher, Ankesh Anand, Richard Ives, James Keeling, Karel Lenc, Salem Haykal, Siamak Shakeri, Pranav Shyam, Aakanksha Chowdhery, Roman Ring, Stephen Spencer, Eren Sezener, and et al. 2024. Gemini 1.5: Unlocking multimodal understanding across millions of tokens of context. *arXiv:2403.05530*.
- Noah Shinn, Federico Cassano, Ashwin Gopinath, Karthik Narasimhan, and Shunyu Yao. 2023. Reflexion: language agents with verbal reinforcement learning. In *Adv. Neural Inf. Process. Syst.*
- Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal

- Azhar, et al. 2023. Llama: Open and efficient foundation language models. *arXiv:2302.13971*.
- Guanzhi Wang, Yuqi Xie, Yunfan Jiang, Ajay Mandlekar, Chaowei Xiao, Yuke Zhu, Linxi Fan, and Anima Anandkumar. 2024a. Voyager: An open-ended embodied agent with large language models. *Trans. Mach. Learn. Res.*
- Peng Wang, Shuai Bai, Sinan Tan, Shijie Wang, Zhihao Fan, Jinze Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Yang Fan, Kai Dang, Mengfei Du, Xuancheng Ren, Rui Men, Dayiheng Liu, Chang Zhou, Jingren Zhou, and Junyang Lin. 2024b. Qwen2-VL: Enhancing vision-language model’s perception of the world at any resolution. *arXiv:2409.12191*.
- Haixu Wu, Tengge Hu, Yong Liu, Hang Zhou, Jianmin Wang, and Mingsheng Long. 2023. TimesNet: Temporal 2D-variation modeling for general time series analysis. In *Proc. Int. Conf. Learn. Represent.*
- Yijia Xiao, Edward Sun, Di Luo, and Wei Wang. 2024. TradingAgents: Multi-agents llm financial trading framework. *arXiv:2412.20138*.
- An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jingren Zhou, Junyang Lin, Kai Dang, Keming Lu, Keqin Bao, Kexin Yang, Le Yu, Mei Li, Mingfeng Xue, Pei Zhang, Qin Zhu, Rui Men, Runji Lin, Tianhao Li, Tingyu Xia, Xingzhang Ren, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yu Wan, Yuqiong Liu, Zeyu Cui, Zhenru Zhang, and Zihan Qiu. 2024. Qwen2.5 technical report. *arXiv:2412.15115*.
- Li Yang, Zhiding Xiao, Wenxin Huang, and Xian Zhong. 2025. StoryLLaVA: Enhancing visual storytelling with multi-modal large language models. In *Proc. Int. Conf. Comput. Linguist.*, pages 3936–3951.
- Chong Zhang, Xinyi Liu, Zhongmou Zhang, Mingyu Jin, Lingyao Li, Zhenting Wang, Wenyue Hua, Dong Shu, Suiyuan Zhu, Xiaobo Jin, et al. 2024a. When ai meets finance (stockagent): Large language model-based stock trading in simulated real-world environments. *arXiv:2407.18957*.
- Zhihan Zhang, Yixin Cao, Chenchen Ye, Yunshan Ma, Lizi Liao, and Tat-Seng Chua. 2024b. Analyzing temporal complex events with large language models? A benchmark towards temporal, long context understanding. In *Proc. Annu. Meet. Assoc. Comput. Linguist.*, pages 1588–1606.
- Wanjuan Zhong, Lianghong Guo, Qiqi Gao, He Ye, and Yanlin Wang. 2024. Memorybank: Enhancing large language models with long-term memory. In *Proc. AAAI Conf. Artif. Intell.*, pages 19724–19731.
- Pei Zhou, Hyundong Cho, Pegah Jandaghi, Dong-Ho Lee, Bill Yuchen Lin, Jay Pujara, and Xiang Ren. 2022. Reflect, not reflex: Inference-based common ground improves dialogue response quality. In *Proc. Conf. Empir. Methods Nat. Lang. Process.*, pages 10450–10468.
- Xizhou Zhu, Yuntao Chen, Hao Tian, Chenxin Tao, Weijie Su, Chenyu Yang, Gao Huang, Bin Li, Lewei Lu, Xiaogang Wang, et al. 2023. Ghost in the minecraft: Generally capable agents for open-world environments via large language models with text-based knowledge and memory. *arXiv:2305.17144*.

A Agent Trading Arena

A.1 Overview

As illustrated in Fig. 7, each agent’s workflow integrates LLMs for chat pool interactions, stock analysis, decision-making, and reflection. In the stock analysis and decision-making modules, all outputs are validated for consistency with both common sense and operational requirements before execution.

A.1.1 Action Decision-Making

Action generation follows the LLM framework. The agent responsible for generating actions receives corresponding prompts via SQLite. Based on these prompts and specified output formats, the agent decides whether to buy, sell, or hold stocks. The action generation process is outlined in Eq. (1), with input prompts shown in Fig. 8 and the corresponding outputs displayed in the adjacent figure.

$$\begin{cases} A_{\text{date}+1}^t = \Psi(\text{Ins}, Z_{\text{date}}^t, S_{\text{date}}), & \text{if } t \text{ is Iters,} \\ A_{\text{date}}^{t+1} = \Psi(\text{Ins}, Z_{\text{date}}^t, S_{\text{date}}), & \text{otherwise,} \end{cases} \quad (1)$$

where Ins represents the environment introduction, Z_{date}^t denotes the memory of the stock transaction on day date retrieved from the database, and S_{date} is the strategy for day date generated via reflection.

A.1.2 Environmental Interaction

To isolate external influences, we created a virtual sandbox environment where each agent is assigned a unique ID, and their actions affect the environment. The function ϕ facilitates environmental interactions, as shown in Algorithm 1, where “OPS” retrieves agent actions, “date” refers to the trading date, and “Z” represents the memory used for interaction with the environment. Through ϕ , each agent’s actions, such as buying or selling stocks, determine the current stock price and update the trading platform, including stock prices and available shares. Stock prices are independent of external factors and are influenced solely by the sandbox’s internal dynamics. The stock price is updated with each transaction according to the following formula:

$$\begin{aligned} \text{Price}_{\text{curr}} &= \delta(Q, F, \text{Price}_{\text{curr}}, \text{Price}_{\text{deal}}) \\ &= \frac{\text{Price}_{\text{deal}} \cdot Q \cdot F + \text{Price}_{\text{curr}} \cdot Q_{\text{total}}}{Q \cdot F + Q_{\text{total}}}, \end{aligned} \quad (2)$$

where Q is the quantity of stock traded, F is the fluctuation constant, $\text{Price}_{\text{curr}}$ is the current stock

price, Q_{total} is the total number of shares available, and $\text{Price}_{\text{deal}}$ is the price at which the trade occurs.

Algorithm 1: Environmental Interaction

Input: OPS: Function to retrieve agents’ actions, date: The current trading date, Z: Memory used for interaction with the environment

Output: Z

```

for  $P \in \text{Persons}$  do
   $A \leftarrow \text{OPS}(t, P)$ ;
   $O, N, Q, \text{Price}_{\text{deal}} \leftarrow \text{Extract}(A)$ ;
   $\text{Price}_{\text{curr}}, Q_{\text{total}} \leftarrow \text{Stocks}(N)$ ;
   $\text{Price}_{\text{curr}} \leftarrow \delta(Q, F, \text{Price}_{\text{curr}}, \text{Price}_{\text{deal}})$ ;
  if  $O = \text{"buy"}$  then
     $\text{Cash} \leftarrow \text{Price}_{\text{curr}} \cdot Q$ ;
    if  $\text{Cash} < P.\text{Cash}$  then
       $Z \leftarrow \text{SubmitOrder}(O, N, t, \text{Price}_{\text{curr}}, Q)$ ;
  if  $O = \text{"sell"}$  then
     $\text{Hold} \leftarrow P(N)$ ;
    if  $\text{Hold} \neq \text{None}$  then
       $Q_N \leftarrow \text{Hold}[\text{"Q"}] - Q$ ;
      if  $Q_N > 0$  then
         $Z \leftarrow \text{SubmitOrder}(O, N, t, \text{Price}_{\text{curr}}, Q)$ ;
   $\text{Market}(\text{date}, \text{Persons})$ ;
return Z;

```

The function ϕ executes trading orders and updates the stock price in real-time based on the agent’s actions. To prevent excessive volatility and mitigate risk during trading cycles, a daily price fluctuation cap is enforced. Before executing any transaction, each agent evaluates its available funds and refrains from proceeding if insufficient capital is available.

A.1.3 Memory

The superior performance of LLM-based agents arises from the extensive internal knowledge acquired during pre-training. The large number of parameters in LLMs enables the retrieval of diverse information and supports logical and inferential reasoning. To further enhance knowledge retrieval across various tasks, we incorporate a memory module that empowers LLM-based agents with self-improvement capabilities. This memory module facilitates strategy reflection through time-series feedback. Unlike qualitative tasks, quantitative feedback evolves incrementally with subtle differences, presenting a challenge for the generalization of existing LLM-based agents.

To minimize the influence of pre-existing knowledge, we assign specific roles to each agent. The memory module records sensory inputs and accu-

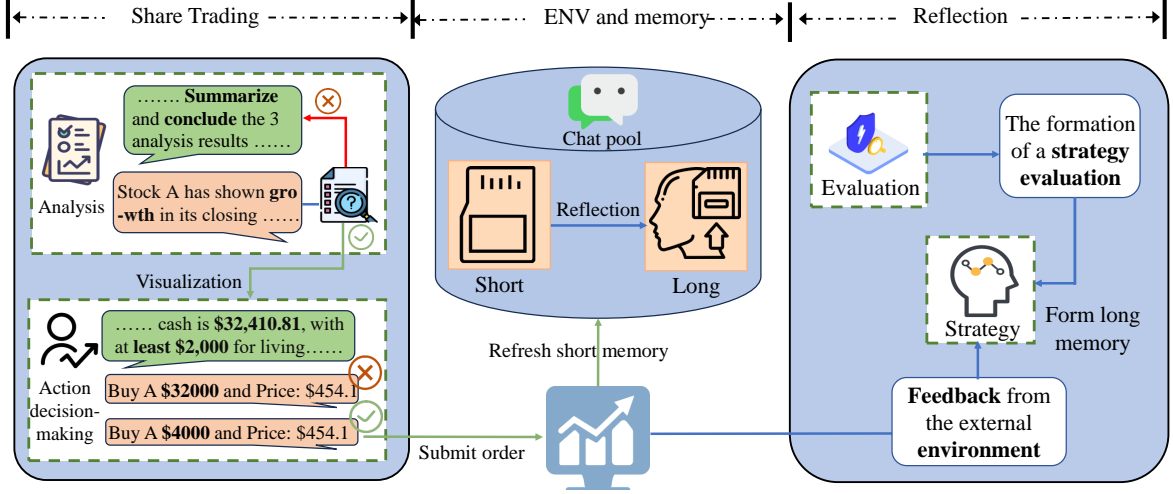


Figure 7: **Agent Workflow Components:** Share Trading involves stock analysis, decision validation, and trade execution; Environment and Memory manage memory and process trade orders; and Reflection focuses on strategy assessment and refinement based on feedback.

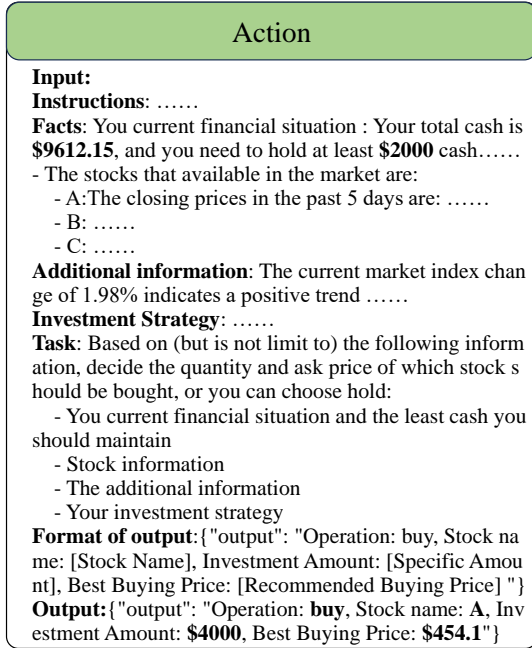


Figure 8: **Inputs and Outputs in Action Decision-Making.**

mulates valuable experiences based on immediate feedback following actions. These experiences are stored in a database for future reference. The interaction history between an agent’s behavior and the environment constitutes short-term memory, enabling the agent to retain recent events. The historical trajectory is defined as:

$$M_{\text{date}}^t = \zeta(I_{\text{date}}^t, \text{Out}_{\text{date}}^t), \quad (3)$$

where ζ processes key information, I_{date}^t represents the input prompt, and $\text{Out}_{\text{date}}^t$ denotes the resulting output.

The day’s trajectory constitutes short-term memory, which is expressed as:

$$Z_{\text{date}}^t = (M_{\text{date}}^0, M_{\text{date}}^1, \dots, M_{\text{date}}^t). \quad (4)$$

The process for updating short-term memory is given by:

$$Z_{\text{date}}^{t+1} = M_{\text{date}}^t \cup Z_{\text{date}}^t, \quad t \in \{0, 1, \dots, T\}, \quad (5)$$

where Z_{date}^t represents the short-term memory, and T denotes the maximum number of iterations.

The reflection model serves as long-term memory, enabling self-reflection and the consolidation of knowledge.

A.1.4 Reflection

We propose a strategy distillation method that transforms quantitative results into descriptive text, which is then used as prompts for LLMs. This approach aids in the analysis of results and the generation of actionable, qualitative summaries, enabling LLMs to derive new strategies. These strategies are implemented, monitored, and evaluated over time, while underperforming strategies are archived for future review.

Initially, we evaluate the day’s trajectory memory and associated strategies. The evaluation function is defined as:

$$E_{\text{date}} = \Psi(\text{Ins}, Z_{\text{date}}^T, S_{\text{date}}), \quad (6)$$

where Ψ represents the evaluation function based on LLMs, Ins contains the agent role descriptions and output requirements, and Z_{date}^T denotes the memory for the given day.

Based on the evaluation results and previous strategies, we generate the latest strategy. Past strategies are stored in a library and scored. For the new strategy, we select the top five best-performing and the bottom five worst-performing strategies to provide both positive and negative feedback. The strategy update formula is:

$$S_{\text{date}+1} = \Psi(\text{Ins}, Z_{\text{date}}^T, E_{\text{date}}, S_{\text{date}}), \quad (7)$$

where E_{date} represents the evaluation results for the day, and S_{date} is the strategy for that day.

Long-term memory is generated through reflection, as represented by:

$$Z'_{\text{date}} = (S_0, S_1, \dots, S_{\text{date}}). \quad (8)$$

The process for updating long-term memory is defined as follows:

$$\begin{aligned} Z'_{\text{date}+1} &= S_{\text{date}} \cup Z'_{\text{date}}, \\ \text{date} &\in \{0, 1, \dots, \text{DAYS}\}. \end{aligned} \quad (9)$$

where Z'_{date} represents the long-term memory, and DAYS denotes the maximum number of days.

Together, short-term and long-term memory provides essential context for the agents. Success in this environment depends on their ability to understand the game rules and develop strategies that outmaneuver competitors. Agents continuously refine their strategies based on incremental quantitative feedback, adjusting their actions to align with long-term objectives.

B NASDAQ and CSI

This study selects seven stocks from the NASDAQ exchange: AAPL, AMZA, GOOGL, MSFT, NFLX, NVDA, and TSLA. These stocks represent leading companies in the technology, energy, and automotive sectors, providing high market representativeness and significant trading volumes. The NASDAQ dataset from Yahoo Finance spans July 3, 2023, to October 29, 2024, excluding weekends and holidays. This dataset reflects current market trends and serves as a timely foundation for our research. It includes daily records of opening price, closing price, highest price, lowest price, and trading volume, as well as relevant technical indicators, offering a comprehensive view of market behavior.

This study also selects seven representative stocks from the Chinese stock market (CSI), covering various sectors with industry-leading companies: Kweichow Moutai (600519.SH), China Merchants Bank (600036.SH), CATL (300750.SZ), BYD (002594.SZ), LONGi Green Energy (601012.SH), ZTE (000063.SZ) and Mindray (300760.SZ). These stocks span consumer goods, finance, new energy, telecommunications, and healthcare, representing leading companies with strong market influence, stable performance, and innovative potential. For CSI, we conducted a one-year simulation from January 2 to December 31, 2024, using the same initial capital setup, while excluding weekends and official trading holidays. The data set is sourced from Baostock and includes daily records such as the opening price, closing price, highest price, lowest price, trading volume, and multiple technical indicators. This comprehensive data set provides a solid foundation for analyzing market trends and investment strategies within the CSI framework.

The training and testing periods for StockFormer (Gao et al., 2023), TimesNet (Wu et al., 2023), StockMixer (Fan and Shen, 2024), and our system is shown in Table 7. StockFormer, TimesNet, and StockMixer require longer training periods, and their training data sets span a broader time range compared to the other models. In contrast, our system does not require training and relies solely on historical stock data to make trading decisions.

The testing period is set from September 3, 2024, to October 29, 2024, to prevent potential data leaks that could provide prior knowledge to the GPT-4o system. This timeframe ensures the fairness of the evaluation by mitigating biases from such leaks, thereby enhancing the reliability of the experimental outcomes.

We also select two stocks from the CSI dataset, each representing a different growth pattern, for simulation. As shown in Table 8, our method consistently outperforms the average trend of the selected stocks.

C Prompt Design Across Modalities

The prompts used for stock analysis with text-only, vision-only, and combined inputs are shown below. Additional prompts for others can be found at: <https://github.com/wekjsdvn/Agent-Trading-Arena>.

Text-only Prompt Design

Instructions:

- You are a player participating in a Simulated Stock Trading Challenge where the prices of stocks are determined mainly by the bids made by the participants in the challenge.
- You will be given information related to specific stocks, the whole market, and your existing investments, which is used to analyze the current stock market and investment situation.
- Your overall objective is to make as much profit as possible.

Stock information:

- A:
 - The closing prices in the past 10 days are: [437.80, 459.44, 465.25, 490.38, 501.034, 511.65, 511.72, 511.79, 511.78, 511.79]
 - Dividend per share: 22
 - Current price change: +0.00%, Current price: 511.79
 - Intraday High: 511.79
 - Intraday Low: 511.79
 - Intraday Mean: 511.79
- B:
 - The closing prices in the past 10 days are: [460.75, 465.80, 493.27, 502.06, 502.49, 497.32, 486.28, 468.01, 480.61, 480.61]
 - Dividend per share: 23
 - Current price change: +0.00%, Current price: 480.61
 - Intraday High: 480.61
 - Intraday Low: 480.61
 - Intraday Mean: 480.61
- C:
 - The closing prices in the past 10 days are: [455.90, 440.532, 424.91, 419.75, 420.48, 421.31, 420.12, 421.09, 435.33, 435.33]
 - Dividend per share: 25
 - Current price change: +-0.00%, Current price: 435.33
 - Intraday High: 435.33
 - Intraday Low: 435.33
 - Intraday Mean: 435.33

Market information:

Current market index change: 0.58%

Gossip from other people:

- Exciting developments in the stock market are stirring, with whispers suggesting a potential merger between companies holding Stock A and B. This merger is rumored to lead to significant synergies, possibly resulting in a stock price surge for both A and B. Consequently, many traders anticipate a rise in demand for these stocks, suggesting a timely investment could yield remarkable returns. Meanwhile, analysts rave about Stock C's steady position and attractive dividends, marking it as a fortress stock that continues to deliver.....

Existing Investments:

Your total portfolio balance is 134807.00, you are holding the following stocks:

- B:
 - You have held 162 shares of this stock
 - the total portfolio value is 73839.6 the total capital gain is 5.00% PROFIT
 - The prices in the past 10 days are: prices: [460.75, 465.80, 493.27, 502.06, 502.49, 497.32, 486.28, 468.01, 480.61, 480.61]
 - current price change: +0.00%, current price: 480.61, cost price: 455.8
- A:
 - You have held 130 shares of this stock
 - the total portfolio value is 60967.4 the total capital gain is 6.00% PROFIT
 - The prices in the past 10 days are: prices: [437.80, 459.44, 465.25, 490.38, 501.034, 511.65, 511.72, 511.79, 511.78, 511.79]
 - current price change: +0.00%, current price: 511.79, cost price: 468.98

Investment strategy:

try to maximize profit.

Task:

- Based on (but not limited to) the following information, you need to summarize and conclude the 3 most important and valuable analysis results from the above information. The analysis results must fully fit the focus of the investment strategy, and each result should describe the relationship between its content and investment strategy. In the next stage, the analysis results will be referred for the decision to buy or sell:
 - Stock information
 - Market information
 - Your existing investments
 - Your investment strategy
 - Gossip from other people (Gossip may be real or fake news.)

Output the response to the prompt above in JSON. Each analysis result should be started with "-", and ended with a line break.

Please only provide the response in the following format:

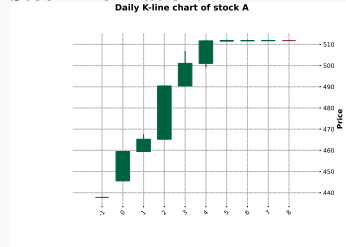
```
{"output": "The analysis results: [analysis results]"}
```

Visual-only Prompt Design

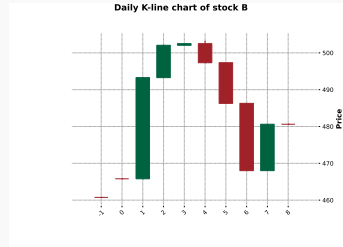
Instructions:

- You are a player participating in a Simulated Stock Trading Challenge where the prices of stocks are determined mainly by the bids made by the participants in the challenge.
- You will be given information related to specific stocks, the whole market, and your existing investments, which is used to analyze the current stock market and investment situation.
- Your overall objective is to make as much profit as possible.

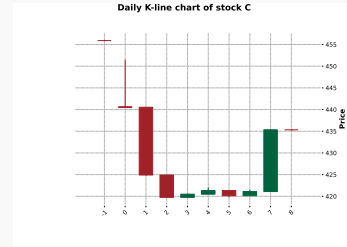
Stock information:



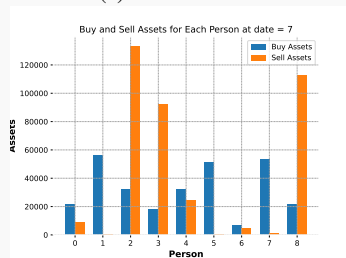
(a) Stock A Price



(b) Stock B Price



(c) Stock C Price



(d) Buy & Sell Asset



(e) Trading Record

Market information:

Current market index change: 0.58%

Gossip from other people:

- Exciting developments in the stock market are stirring, with whispers suggesting a potential merger between companies holding Stock A and B. This merger is rumored to lead to significant synergies, possibly resulting in a stock price surge for both A and B. Consequently, many traders anticipate a rise in demand for these stocks, suggesting a timely investment could yield remarkable returns. Meanwhile, analysts rave about Stock C's steady position and attractive dividends, marking it as a fortress stock that continues to deliver consistent profits with minimal risk. As traders weigh their options, strategic portfolio rebalancing tips towards these dividends-rich stocks seem to be the talk of the town. However, with the volatile nature of stock rumors, it remains essential to tread carefully and keep an eye on reliable sources for confirmation. Will this buzz hold substance, or is it merely hot air? Only time will reveal the truth behind the whispers, leaving the market in suspense.

Existing Investments:

Your total portfolio balance is 134807.00, you are holding the following stocks:

- C:
 - You have held 162 shares of this stock
 - the total portfolio value is 73839.6 the total capital gain is 5.00% PROFIT
- A:
 - You have held 130 shares of this stock
 - the total portfolio value is 60967.4 the total capital gain is 6.00% PROFIT

Investment strategy:

try to maximize profit.

Task:

- Based on (but not limited to) the following information, you need to summarize and conclude the 3 most important and valuable analysis results from the above information. The analysis results must fully fit the focus of the investment strategy, and each result should describe the relationship between its content and investment strategy. In the next stage, the analysis results will be referred for the decision to buy or sell:

- Stock information
- Market information
- Your existing investments
- Your investment strategy
- Gossip from other people (Gossip may be real or fake news.)
- The provided stock information visualizations.

Output the response to the prompt above in JSON. Each analysis result should be started with "-", and ended with a line break.

Please only provide the response in the following format:

{"output": "The analysis results: [analysis results]"}

Combined-Input Prompt Design

Instructions:

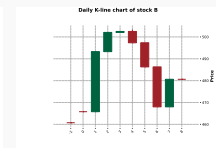
- You are a player participating in a Simulated Stock Trading Challenge where the prices of stocks are determined mainly by the bids made by the participants in the challenge.
- You will be given information related to specific stocks, the whole market, and your existing.....

Stock information:

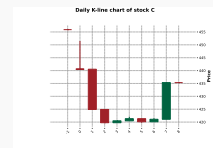
- A:
 - The closing prices in the past 10 days are: [437.80, 459.44, 465.25, 490.38, 501.034, 511.65, 511.72, 511.79, 511.78, 511.79]
 - Dividend per share: 22
 - Current price change: +0.00%, Current price: 511.79
 - Intraday High: 511.79
 - Intraday Low: 511.79
 - Intraday Mean: 511.79
- B:
 - The closing prices in the past 10 days are: [460.75, 465.80, 493.27, 502.06, 502.49, 497.32, 486.28, 468.01, 480.61, 480.61]
 - Dividend per share: 23
 - Current price change: +0.00%, Current price: 480.61
 - Intraday High: 480.61
 - Intraday Low: 480.61
 - Intraday Mean: 480.61
- C:
 - The closing prices in the past 10 days are: [455.90, 440.532, 424.91, 419.75, 420.48, 421.31, 420.12, 421.09, 435.33, 435.33]
 - Dividend per share: 25
 - Current price change: +-0.00%, Current price: 435.33
 - Intraday High: 435.33
 - Intraday Low: 435.33
 - Intraday Mean: 435.33



(a) Stock A Price



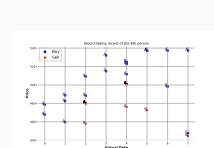
(b) Stock B Price



(c) Stock C Price



(d) Buy & Sell Asset



(e) Trading Record

Market information:

Current market index change: 0.58%

Gossip from other people:

- Exciting developments in the stock market are stirring, with whispers suggesting a potential merger between companies holding Stock A and B. This merger is rumored to lead to significant synergies.....

Existing Investments:

Your total portfolio balance is 134807.00, you are holding the following stocks:

- B:
 - You have held 162 shares of this stock
 - the total portfolio value is 73839.6 the total capital gain is 5.00% PROFIT
 - The prices in the past 10 days are: prices: [460.75, 465.80, 493.27, 502.06, 502.49, 497.32, 486.28, 468.01, 480.61, 480.61]
 - current price change: +0.00%, current price: 480.61, cost price: 455.8
- A:
 - You have held 130 shares of this stock
 - the total portfolio value is 60967.4 the total capital gain is 6.00% PROFIT
 - The prices in the past 10 days are: prices: [437.80, 459.44, 465.25, 490.38, 501.034, 511.65, 511.72, 511.79, 511.78, 511.79]
 - current price change: +0.00%, current price: 511.79, cost price: 468.98

Investment strategy:

try to maximize profit.

Task:

- Based on (but not limited to) the following information, describe the relationship between its content and investment strategy. In the next stage, the analysis results will be referred for the decision to buy or sell:
- Stock information
- Market information
- Your existing investments
- Your investment strategy
- Gossip from other people (Gossip may be real or fake news.)
- The provided stock information visualizations.

Output the response to the prompt above in JSON. Each analysis result should be started with "-",.....

Please only provide the response in the following format:

```
{"output": "The analysis results: [analysis results]"}
```

Strategy	NASDAQ					CSI				
	Training			Testing		Training			Testing	
	Start	End	Start	End	Start	End	Start	End	Start	End
Buy & Hold	w/o	25/8/2024	30/8/2024	3/9/2024	29/10/2024	w/o	21/12/2023	29/12/2023	2/1/2024	31/12/2024
SMA	w/o	1/8/2024	30/8/2024	3/9/2024	29/10/2024	w/o	1/12/2023	29/12/2023	2/1/2024	31/12/2024
ZMR	w/o	10/8/2024	30/8/2024	3/9/2024	29/10/2024	w/o	8/12/2023	29/12/2023	2/1/2024	31/12/2024
MACD	w/o	4/8/2024	30/8/2024	3/9/2024	29/10/2024	w/o	4/12/2023	29/12/2023	2/1/2024	31/12/2024
StockFormer	w/	3/7/2023	30/8/2024	3/9/2024	29/10/2024	w/	2/1/2020	29/12/2023	2/1/2024	31/12/2024
TimesNet	w/	3/7/2023	30/8/2024	3/9/2024	29/10/2024	w/	2/1/2020	29/12/2023	2/1/2024	31/12/2024
StockMixer	w/	3/7/2023	30/8/2024	3/9/2024	29/10/2024	w/	2/1/2020	29/12/2023	2/1/2024	31/12/2024
Ours	w/o	16/8/2024	30/8/2024	3/9/2024	29/10/2024	w/o	1/12/2023	29/12/2023	2/1/2024	31/12/2024

Table 7: Training and Testing Periods for Various Models on NASDAQ and CSI.

Trend			LLMs	TR \uparrow	SR \uparrow
Average	Min	Max			
-5.86	-13.77	-0.21	GPT4o	-0.50	-0.021
			Qwen	2.01	0.207
			LlaMa3	1.44	0.122
11.37	-4.25	26.05	GPT4o	20.61	0.465
			Qwen	16.46	0.420
			LlaMa3	4.14	0.167

Table 8: Performance on CSI Stocks with Different Growth Patterns. Our method consistently outperforms the average trend of both selected stocks.

LLMs	T	V	TR \uparrow	SR \uparrow
LLaMa-3	●	○	100.00	100.00
DeepSeek	●	○	+ 35.03	+ 45.63
Gemini-1.5	●	○	+ 10.99	+ 14.93
	○	●	+ 11.49	+ 29.47
GPT-4o	●	○	+ 10.39	+ 28.29
	○	●	<u>+ 17.18</u>	<u>+ 40.77</u>

Table 9: Performance Comparison in Trading Decisions *w/o* Reflection, Competing Pairwise with LLaMa-3. DeepSeek may possess unique strengths or optimizations that allow it to better adapt to the task’s complexities when competing with other models.

D Simulation Process

The experiments involved several LLMs, including LLaMa-3 (Dubey et al., 2024), GPT-4o (Hurst et al., 2024), GPT-4o-mini (Hurst et al., 2024), DeepSeek (Liu et al., 2024a), Qwen-2.5 (Yang et al., 2024), Qwen-VL-32k (Wang et al., 2024b), Qwen-VL-128kk (Wang et al., 2024b), and Gemini-1.5 (Reid et al., 2024), corresponding to the models Meta-LLaMa-3-70B-Instruct, gpt-4o-2024-08-06, gpt-4o-mini-2024-07-18, DeepSeek-chat, Qwen2.5-72B-Instruct, qwen-vl-max, qwen-vl-max-latest, and Gemini-1.5-pro.

E Visualization Input

Fig. 9 illustrates the system’s input prompts and corresponding outputs during the strategy update process. The input prompts consist of both textual and visual components, including daily K-line charts, transaction histories, and agent trading volumes, all of which inform the strategy update.

F Simulation Process

In the *Agent Trading Arena*, the simulation process unfolds as follows: First, rumors are generated in the chat pool based on the previous day’s stock

market analysis. Next, historical stock data is analyzed, followed by decision-making and execution. Short-term memory is formed through interactions with the environment. Finally, the system evaluates this memory, updates the strategy, and consolidates it into long-term memory. This entire process is illustrated in Fig. 10.

Impact of Modality on LLM Competitiveness.

We employed a relative evaluation method for this experiment. The first and second agents used various LLMs in textual and visual settings, respectively, while the remaining agents were based on LLaMa-3 (Dubey et al., 2024) as the baseline. This setup aimed to explore the impact of different agents and modalities on LLM performance. The results are shown in Table 9. The findings indicate that DeepSeek (Liu et al., 2024a) exhibited stronger competitive performance across different LLM environments, suggesting unique strengths or optimizations that enable it to adapt more effectively to the task’s complexities.

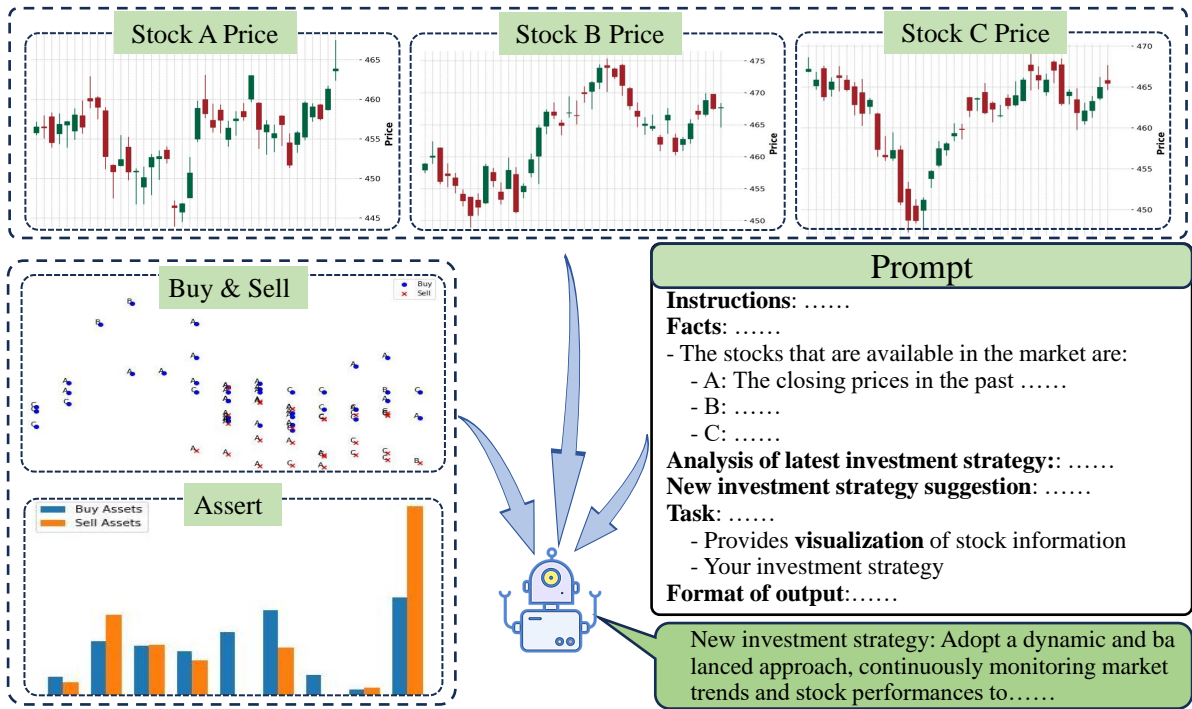


Figure 9: Visualization of Stock Inputs and Corresponding Trading Strategy Outputs.

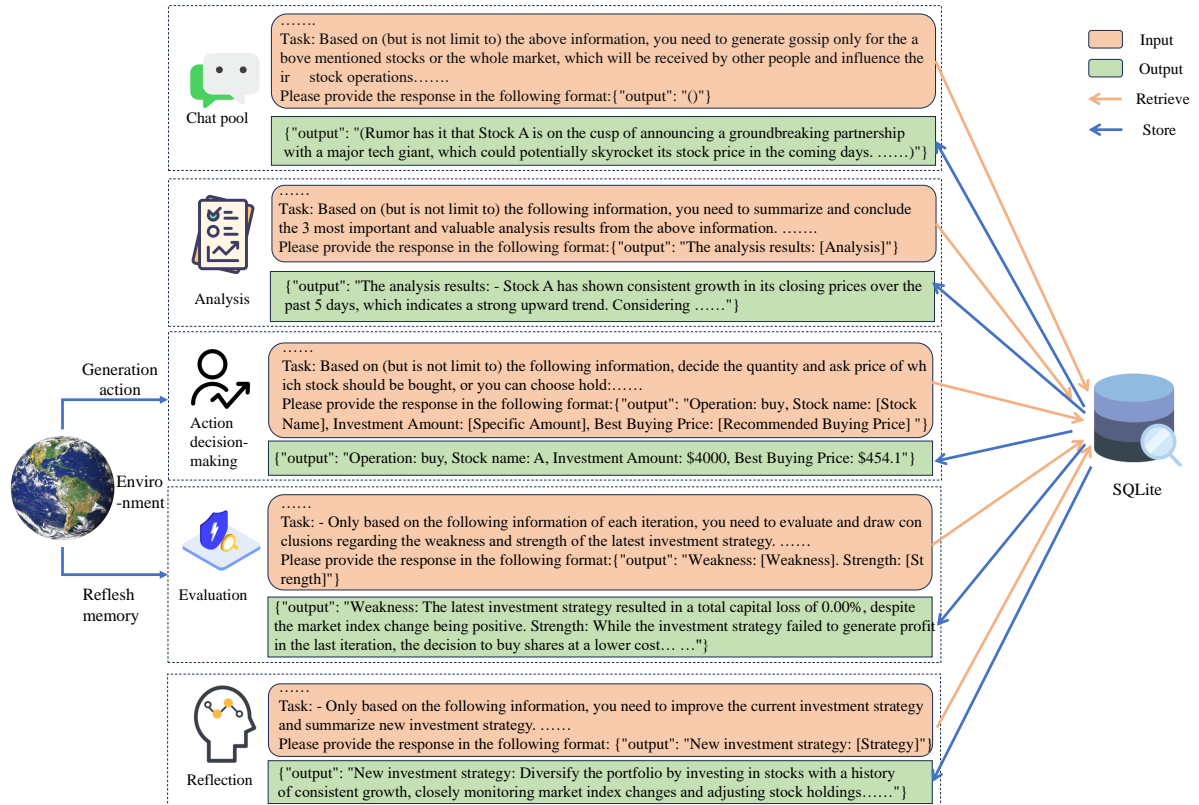


Figure 10: Overview of Complete Workflow.