FINMEM: A PERFORMANCE-ENHANCED LLM TRADING AGENT WITH LAYERED MEMORY AND CHARACTER DESIGN

Haohang Li* **Yangyang Yu*** **Zhi Chen*** **Yuechen Jiang*** **Yang Li ***

Denghui Zhang **Rong Liu** **Jordan W. Suchow** **Khaldoun Khashanah**
Stevens Institute of Technology, Hoboken, NJ, United States
{hli113, yyu44, zchen100, yjiang52, yli269, dzhang42, rliu20, jws, kkhashan}@stevens.edu

ABSTRACT

Recent advancements in Large Language Models (LLMs) have exhibited notable efficacy in natural language processing (NLP) tasks across diverse domains. Their prowess in integrating extensive knowledge has fueled interest in developing LLM-based autonomous agents. Furthermore, in the realm of finance, there is a persistent need to develop automated systems capable of transforming vast quantities of real-time data into executable decisions, while fully understanding the critical timing of various types of information. LLM agents with rational architecture, compared with their Deep Reinforcement Learning (DRL) counterparts, exceed in their ability to integrate textual data and interpretability in their decision-making process. We introduce FINMEM, a novel LLM-based agent framework devised for financial trading. It encompasses three core modules: Profiling, to customize the agent's characteristics; Memory, with layered message processing, to aid the agent in assimilating hierarchical financial data; and Decision-making, to convert insights gained from memories into investment decisions. Notably, FINMEM's memory module aligns closely with the cognitive structure of human traders, offering robust interpretability and real-time tuning. Its adjustable cognitive span allows for the retention of critical information beyond human perceptual limits, thereby enhancing trading outcomes. This framework enables the agent to self-evolve its professional knowledge, react agilely to new investment cues, and continuously refine trading decisions in the volatile financial environment. We first compare FINMEM with various algorithmic agents on a scalable real-world financial dataset, underscoring its leading trading performance in stocks. We then fine-tuned the agent's perceptual span and character setting to achieve a significantly enhanced trading performance. Collectively, FINMEM presents a cutting-edge LLM agent framework for automated trading, boosting cumulative investment returns.

KEYWORDS: Large Language Model, Trading Algorithms, Deep Learning, Financial AI

1 INTRODUCTION

With the growing influx of diverse financial information from the market, human traders are overwhelmed by data from multiple sources (Fang and Zhang (2016)). Consequently, it becomes more difficult for traders to notice critical events affecting their trading decisions. To overcome the challenge, researchers have been consistently working on designing autonomous trading agent systems that are able to integrate all necessary information with decent trading performance in various market conditions.

Research in Deep Reinforcement Learning Agents (DRL) (Millea (2021)) has been a focal point of attention in both the academic and industrial realms for years. Leveraging both Reinforcement

[1]* These authors contributed equally to this work.

Learning (RL) principles and deep learning, DRL agents effectively handle and learn from scalable and diverse financial data, including stock prices, key financial indicators, and market sentiments. However, certain inherent features of DRL algorithms exhibit notable deficiencies in financial applications. **Firstly, DRL agents lack interpretability in the rationale behind their decision-making** (Balhara et al. (2022)). **Secondly, integrating textual data with numerical features, crucial in finance, poses a challenge for DRL agents on convergence** (Gershman and Ölveczky (2020)). Thus, a backbone algorithm that offers transparent reasoning and effectively captures investment-related textual insights is essential.

Recent advancements in Large Language Models (LLMs), like Generative Pre-trained Transformers (GPTs) (OpenAI (2023)), have opened new avenues for developing trading agents, addressing past limitations. These LLM-based agents can articulate reasons and outcomes from their immediate observations. Their extensive pre-trained knowledge and ability to integrate diverse data sources, including textual and numerical information, allow them to transcend the constraints of isolated environments. This approach, when reinforced with well-designed prompt templates, markedly improves decision-making in various sectors (Wang et al. (2023)). Notably, a growing body of research has focused on utilizing LLMs to make informed trading decisions for stocks (Yang et al. (2023); Wu et al. (2023)). However, in currently available approaches, LLMs primarily serve as trading signal detectors with Question and Answering(QA) prompts rather than autonomous agents with task-specific architecture for financial applications. In particular, they are missing a core component that enables LLMs to fully manage the information timeliness of financial data and prioritize critical events, which is memory.

To bridge this gap, we introduce FINMEM, a novel LLM-based autonomous trading agent. It excels in processing multi-source financial data through a layered memory module and adapts to market volatility by offering a self-adaptive character setting. Our concept draws from the Generative Agents framework by Park et al.. The memory-based autonomous agent prioritizes events in a unified memory stream, ranked by a linear combination of recency, relevancy, and importance. **However, Park et al.'s framework struggles with comprehending financial information with varying timeliness and importance**. Key challenges involve quantifying the distinct timeliness of data, optimizing information retrieval, and providing detailed reflections to improve future decisions. To tackle these challenges, we further propose FINMEM with the following improvements:

1) FINMEM maintains a modular approach similar to Park et al., but features novel design of profiling and memory modules. FINMEM's memory module innovatively incorporates working memory and layered long-term memory components, ideal for stratified information processing, which mirrors the human cognitive system Sweller (2012) and facilitates agile, real-time decisions Sun (2004). **2) FINMEM can utilize its distinctive features to expand the agent's perceptual range beyond the human limitation to make well-informed trading decisions.** Cognitive research suggests that human working memory is limited to recalling five to nine events at once Miller (1956). While this avoids information overload, it may lead to insufficient insight for accurate decision-making. In contrast, FINMEM's memory module transcends this constraint. It allows adjusting cognitive load by selecting a flexible number of top-ranked events from each layer of its long-term memory, allowing FINMEM to deliver superior trading decisions in data-rich contexts. **3) FINMEM achieves impressive trading performance using training data that is limited in volume and spans a short time period.** Experiments show that training FINMEM with a timeframe much shorter than that required by comparative models. This efficiency stems from optimally utilizing multi-source data and capturing critical trading signals. Notably, FINMEM is effective even on smaller datasets and with general-purpose LLMs, with its performance expected to enhance further with larger, higher-quality financial datasets and LLMs fine-tuned for financial applications.

2 RELATED WORK

2.1 BACKBONE ALGORITHMS OF CONTEMPORARY AUTONOMOUS TRADING AGENTS

The development of trading agents has evolved over several decades, influenced by advancements in technology, finance, and computational methodologies. Conventionally, a rule-based algorithm for trading stocks is an automated strategy that operates based on a predefined set of rules (Chen (2012); Vaidya (2020); Pätäri and Vilska (2014)). These rules are often derived from historical market patterns and trading experience. Compared with rule-based algorithms that use predefined rules and conditions, Reinforcement Learning provides a way for agents to learn by interacting with an environment and receiving feedback in the form of rewards or penalties. (Dang (2019); Jangmin et al. (2006)). Deep learning models can be integrated with RL to handle large and complex state spaces, like those in stock markets. Such models are often referred to as Deep Reinforcement Learning (DRL) (Wu et al. (2020); Xiong et al. (2018)). For example, Deep Q-Network (DQN) (Shi et al. (2021)), Advantage Actor-Critic (A2C) (Yang et al. (2020)), and Proximal Policy Optimization (PPO) (Liu et al. (2020)) are popular algorithms for such tasks. Using DRL agents as automated financial trading backbones, face two key issues: 1) A lack of interpretability, as their decisions, rooted in complex computations and high-dimensional representations, are challenging to articulate (Balhara et al. (2022)). 2) They struggle to efficiently utilize textual financial information due to the complexity and computational demands of rich text embeddings (Devlin et al. (2018); Ethayarajh (2019)). Consequently, DRL agents often rely on textual sentiment (Pricope (2021)), avoiding direct embedding use (Chen and Huang (2021); Avramelou et al. (2023)), which leads to an incomplete representation of essential market information contained in news and macroeconomic policies.

2.2 LLM Autonomous Agents

The new-generation LLMs, like Generative Pre-trained Transformer series (GPTs) (Radford et al. (2018); OpenAI (2023)) and LLM Meta AI (Llamas) (Touvron et al. (2023)), stand out in diverse tasks. As Wang et al. (2023) emphasizes, an architecture with profiling, memory, planning, and action modules is essential for LLMs autonomous agents. There are cases of two modules (e.g., planning and action modules) being integrated as one component (Park et al. (2023)). Among these modules, the memory component is essential. Acting as the operational core, it aligns an agent's actions with real-world tasks. Research indicates that leveraging insights from cognitive science studies on human memory (Wang et al. (2023); Sumers et al. (2023)) can enhance this alignment. While LLM agents for domain-specific tasks have been extensively researched (Huang et al. (2023); Liffiton et al. (2023); Park et al. (2022)), their application in financial trading remains underexplored. Existing studies in this domain, such as Wu et al. (2023); Yang et al. (2023), often lack open-source availability or have not considered an architecture specifically tailored to fit the unique environment of finance markets. Thus, there's significant value in further investigating advanced, transparent LLM agents for trading.

3 Architecture of FinMem

3.1 Profiling Module

The dynamic character of FinMem, as depicted in Figure 4 of Appendix C, comprises two principal components: firstly, a foundational professional knowledge base akin to a trading expert, and secondly, an agent with three distinct investment risk inclinations. The first component includes two types of information: an introduction to the primary trading sectors relevant to the company stock FinMem will trade in, and a concise overview of the historical financial performance of the specified ticker, spanning from the beginning to the end of the training period.

The second component of FinMem's design, illustrated in Figure 4 of Appendix C, encompasses three distinct risk inclination options: risk-seeking, risk-averse, and self-adaptive risk character. The risk-seeking setting gears FinMem towards an aggressive, high-reward approach, while the risk-averse setting gears it towards a conservative, lower-risk strategy. A distinctive aspect of FinMem is its ability to dynamically alternate between these risk settings in response to current market conditions. Specifically, it shifts risk preferences when the Cumulative Return falls to below zero within a brief period, such as three days, and reversely. This flexible design functions as a protective mechanism, mitigating prolonged downturns in the market environments. During the initial stage of the training phase, FinMem is configured with a chosen risk preference, each supplemented with comprehensive textual explanations through LLM prompts. These guidelines shape how FinMem processes incoming messages and determines its subsequent actions in alignment with its designated risk inclination. The system maintains a catalog of all risk inclinations and their detailed explanations in a backlog, enabling seamless adaptation to different stocks by switching among these risk profiles as needed. The dynamic character setting in FinMe's profiling module provides crucial context for filtering and retrieving trading-relevant information and memory events, thus improving accurate inferencing and adaptability to fluctuating market conditions.

3.2 MEMORY MODULE

3.2.1 WORKING MEMORY

Working memory refers to the human cognitive system's functions for temporary storage and diverse operations. FINMEM's working memory encompasses three key operations: summarization, observation, and reflection as depicted in the middle box of Figure 1.

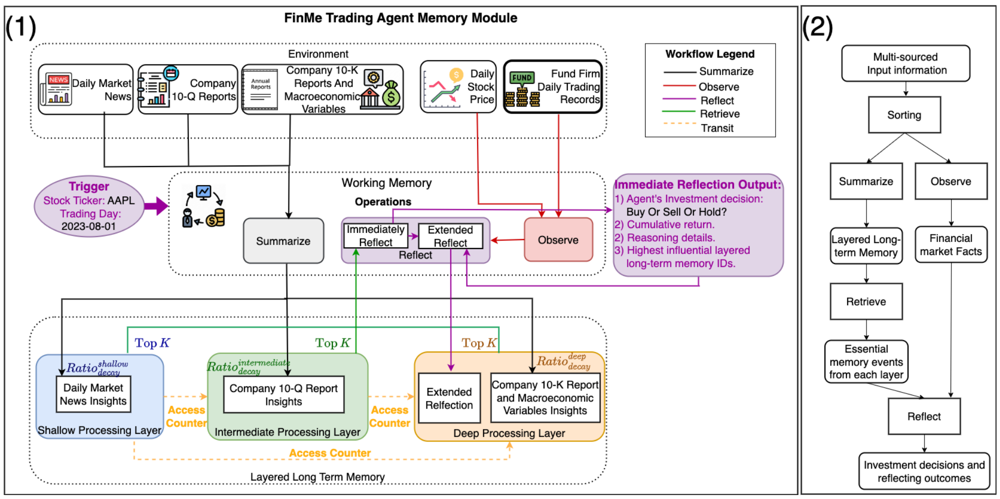

Figure 1: Memory module structure of FINMEM with a detailed view of components, operations, and workflow. The cognitive architectures of FINMEM's memory module have two core components – Working Memory and Layered Long-term Memory.

Summarization and Observation: FINMEM leverages external data to derive critical investment insights and market sentiment. As illustrated in Figure 3(2) of Appendix A, in the **summarization operation**, the system summarizes the original text input into compact yet informative paragraphs, thereby enhancing processing efficiency and reducing inference costs. In the **observation operation**, the agent observes market information and reacts differently in the training and test phase. In the training phase, FINMEM has full access to stock price data within a given period. Upon receiving trading inquiries, FINMEM focuses on the current day's and the following day's adjusted close price difference. The observed price differences are used as market ground truth actions, with a positive difference as "Buy" and vice versa. This design helps the agent align its evaluation to the true market dynamics. In the testing phase, FINMEM loses the access to the future information. Its focus shifts to historical price momentum, depending on a retrospective evaluation of the cumulative return from the last M trading days to infer future market trends, which forces the system to adequately establish logical connections between stock price trends and various financial information sources.

Reflection: Two types of reflections exist, immediate and extended reflection. (a) Immediate reflection is activated upon receiving a daily trading inquiry for a specific ticker. Using LLM and specific prompts exemplified in Figure 3 (2) of Appendix A , the agent merges market indications and top-K-ranked events from each long-term memory layer. Market indications are derived from the outcomes of the observation operation and differ between the training and testing phases. During testing, this process yields three types of outputs: the trading direction ("Buy," "Sell", or "Hold"), the underlying rationale for this decision, and the most influential memory events, along with their IDs from each layer that informed the decision. In the training phase, specifying the trading direction is unnecessary, as FINMEM is pre-informed about future stock trends. The top-K-ranked memory events encapsulate key insights and sentiments derived from critical investment-related incoming messages, all distilled by FINMEM's advanced summarization capabilities. (b) Extended reflection reevaluates outcomes for a ticker over a specified M-day look-back period with evidence of the profit and loss for a given trade action. While immediate reflections enable direct trading execution and record current feedback, extended reflections align the importance of each memory with the market feedback (detailed in Section 3.2.2). The memory event associated with a profited trade will be rewarded by increasing the importance score and thus more likely to be selected, and vice versa.

3.2.2 LAYERED LONG-TERM MEMORY

FINMEM's long-term memory organizes hierarchical financial data insights in a stratified structure, as illustrated in the lower section of Figure 1. Drawing inspiration from the varying decay speeds in the human cognitive system's information processing layers (Craik and Lockhart (1972)), FINMEM employs a layered structure to accommodate the diverse time sensitivities inherent to different types of financial data. This structure categorizes summarized insights by their timeliness and decay rates. Insights are derived by the working memory's summarization operation. Those directed to deeper layers receive smaller decay rates, indicating longer retention, while those in shallower layers are assigned larger decay rates for shorter retention. where $p_1+p_2+p_3=1$, but their values vary by shallow, intermediate, and deep processing. when shallow processing $p_1, p_2, p_3 = \{0.8, 0.15, 0.05\}$, intermediate processing, $p_1, p_2, p_3 = \{0.05, 0.8, 0.15\}$ and deep processing, $p_1, p_2, p_3 = \{0.05, 0.15, 0.8\}$.

$$\gamma_l^E = S^E_{\text{Recency}_l} + S^E_{\text{Relevancy}_l} + S^E_{\text{Importance}_l}, \quad (1) \qquad S^E_{\text{Recency}_l} = e^{-\frac{\delta^E}{Q_l}}, \quad \delta^E = t_{\text{P}} - t_E, \quad (2)$$

where each memory event is only associated with one score and can only belong to a single layer.

Upon receiving an investment inquiry, FINMEM retrieves the top-K pivotal memory events from each layer and channels them to the immediate reflection component of the working memory. These events are chosen according to the descending order of their information retrieval score, denoted as γ_l^E, where l belongs to the set shallow, intermediate, deep, as specified in Equation 3. E denotes a given memory event. This score, adapted Park et al. (2023) but with modified recency and importance computations, especially tailoring to handle data with various timelines. It encapsulates three metrics: recency, relevancy, and importance. Individual metric scores exceeding 1.0 are scaled to the [0,1] range before being summed. The modification is to achieve the layered processing function and represent the various periodicity of the financial environment.

$$\theta_l = (\alpha_l)^{\delta^E}, \quad l = \text{shallow, intermediate, deep}, \quad (3) \qquad S^E_{\text{Importance}_l} = v_l^E * \theta_l, \quad (4)$$

Upon a trade inquiry P arrival in processing layer l via LLM prompt, the agent computes the recency score $S^E_{\text{Recency}_l}$ per Equation 2. $S^E_{\text{Recency}_l}$ inversely correlates with the time gap between the inquiry and the event's memory timestamp, mirroring Ebbinghaus's forgetting curve (**?**). The stability term Q_l in Equation 2 partially controls memory decay rates across layers, indicating longer memory persistence in the long-term layer with a higher stability value. In the context of trading, company annual reports, such as Form 10-Ks, are considered to have more extended timeliness compared to daily financial news. Therefore, they are assigned a higher stability value and are categorized within the deeper processing layer. This classification reflects their extended relevance and impact in financial decision-making scenarios.

The relevancy score, denoted as $S^E_{\text{relevancy}_l}$, quantifies the cosine similarity between embedding vectors. These vectors are derived from the textual content of the memory event, $\mathbf{m_E}$, and the LLM prompt query, $\mathbf{m_P}$, using OpenAI's "text-embedding-ada-002" model, as depicted in Equation 5. The LLM prompt query incorporates inputs related to trading inquiries and the trading agent's character setting.

The importance score $S^E_{\text{Importance}_l}$ is computed using the value v_l^E from a uniform piecewise scoring function (Formula 6), multiplied by degrading ratio θ_l (Formula 3) as per Equation 4. The likelihood of higher v_l^E values increases from shallow to deep layers. θ_l measures the diminishing importance of an event over time, which has a close form design of Park et al. (2023). But our approach tailors θ_l to the stratified structure of long-term memory. It adopts unique exponential functions for each layer. The base α_l for each layer is a hyperparameter, set to follow the sequence: $\alpha_{\text{shallow}} < \alpha_{\text{intermediate}} < \alpha_{\text{deep}}$. These values correlate with the rate at which their importance degrades after a certain period, providing another angle to measure importance variances across different memory types. Through experimentation, we set $\alpha_{\text{shallow}} = 0.9$, $\alpha_{\text{intermediate}} = 0.967$ and $\alpha_{\text{deep}} = 0.988$. This ensures that θ_l decreases to a threshold score of 5 after intervals of $30, 90$, and 365 days for shallow, intermediate, and deep layers, respectively. The three-piece-wise functions for $S^E_{\text{Importance}_l}$ and $S^E_{\text{Recency}_l}$ enable FINMEM to have layered processing in the long-term memory component. Memory events are purged when $S^E_{\text{Recency}_l}$ is below 0.05 or $S^E_{\text{Importance}_l}$ is under 5 (pre-scaling).

$$S^E_{\text{Relevancy}_l} = \frac{\mathbf{m_E} \cdot \mathbf{m_P}}{\|\mathbf{m_E}\|_2 \times \|\mathbf{m_P}\|_2}, \quad (5) \qquad v_l^E = \begin{cases} 40 & \text{with probability } p_1 \\ 60 & \text{with probability } p_2 \\ 80 & \text{with probability } p_3 \end{cases} \quad (6)$$

where δ^E refers to the time difference between the memory event occurrence and the trading inquiry arrival. $Q_{\text{shallow}} = 14$, $Q_{\text{intermediate}} = 90$, and $Q_{\text{deep}} = 365$ correspond to day counts of two weeks, a quarter, and a year for shallow, intermediate, and deep processing layers, respectively.

Furthermore, an access counter function oversees the transfer of memory events among layers, ensuring that significant events influencing trading decisions ascend from shallower to deeper layers for extended retention and recurrent access by FINMEM. Conversely, less pertinent events gradually diminish. This process is facilitated by the LLM validation tool Guardrails AI [2], which monitors critical memory IDs across different layers. An event identified as pivotal for investment success receives an additional 5 points in its importance score $S^E_{\text{Importance}_l}$. Upon meeting the criteria for upgrading to a deeper layer, an event's recency score $S^E_{\text{Recency}_l}$ is reset to 1.0, emphasizing its importance and preventing rapid decay. By implementing this access counter, FINMEM effectively identifies and prioritizes key events, taking into account their nature and frequency of retrieval.

3.3 DECISION-MAKING MODULE

The decision-making module of FINMEM efficiently integrates operational outcomes from the profiling and memory modules to support well-informed investment decisions, as depicted in Figure 3 (1) of Appendix A. In its daily trading decisions, FINMEM is asked to select from three distinct actions for a single share of a specific stock by Guardrails AI text validation function: "Buy," "Sell," or "Hold." Additionally, the inputs and results required by FINMEM's decision-making module vary between its training and testing phases, with each phase's specifics detailed as follows:

During the training phase, FINMEM accesses a wide array of multi-source information relevant to the entire time period. When FINMEM is prompted with trading inquiries containing stock ticker and date, as well as trader character-related texts, it observes the market ground labels mentioned in the description about the observation operation in Section 3.2.1, which involve daily adjusted close price differences between consecutive days, indicating "Buy" or "Sell" actions. Utilizing these price change signals, FINMEM identifies and prioritizes the top-K memories, ranking them based on retrieval scores from each long-term memory layer. This procedure enables FINMEM to produce comprehensive reflections that provide a well-founded rationale and in-depth inference of the correlation between market ground labels and the memories retrieved. Through repeated trading operations, reflections, and memory events with significant impact, transition to a deeper memory processing layer, getting preserved for guiding future investment decisions during the testing phase.

In the testing phase, where FINMEM cannot access future price data, it relies on the Cumulative Return over the previous M trading days as market trends. To compensate for the absence of future market price information, FINMEM utilizes enhanced reflections derived from immediate reflections spanning a M-trading-day period as supplementary references. When faced with a specific trading inquiry, FINMEM integrates insights from various sources, including historical Cumulative Return, outcomes from extended reflection, and the Top-K retrieved memories. This comprehensive approach enables FINMEM to execute well-informed trading decisions.

It should be noted that FINMEM generates executable actions exclusively in the immediate reflection operation of the testing phase. Since the trading direction is guided by the actual price trend, the training phase of FINMEM does not make investment decisions. Instead, this phase is dedicated to accumulating trading experience through comparing market trends with incoming multi-source financial messages. Additionally, during this phase, FINMEM develops a memory module enriched with a comprehensive knowledge base, thereby evolving its capability for independent decision-making in future trading activities.

4 EXPERIMENTS:

Our objective in this section is to assess the trading performance of FINMEM and underscore its distinctive benefits by addressing the following research questions (RQs): **RQ1**: Does FINMEM outperform contemporary leading algorithmic trading agents? **RQ2**: Are there tasks that challenge other trading algorithms but are manageable by FINMEM? **RQ3**: Does equipping FINMEM with different risk inclination choices truly differentiate its trading performance?

[2]Guardrails AI GitHub: https://github.com/guardrails-ai/guardrails

Ticker	Model	Cumulative Return (%)	Sharpe Ratio	Daily Volatility (%)	Annualized Volatility (%)	Max Drawdown (%)
TSLA	Buy and Hold	-18.6312	-0.5410	4.4084	69.9818	55.3208
	FINMEM	**61.7758***	**2.6789**	2.9522	46.8649	**10.7996**
	Generative Agents	13.4636	0.5990	**2.8774**	**45.6774**	24.3177
	FinGPT	-7.4554	-0.2795	3.4145	54.2027	42.3993
	A2C	13.7067	0.3979	4.4096	70.0009	52.3308
	PPO	1.2877	0.0374	4.4110	70.0232	54.3264
	DQN	33.3393	0.9694	4.4027	69.8900	52.0033
NFLX	Buy and Hold	35.5111	1.4109	3.1964	50.7410	20.9263
	FINMEM	**36.4485***	**2.0168**	**2.2951**	**36.4342**	**15.8495**
	Generative Agents	32.0058	1.5965	2.5460	40.4168	16.9893
	FinGPT	9.0090	0.4266	2.6819	42.5732	28.2705
	A2C	14.6155	0.5788	3.2071	50.9112	25.0184
	PPO	8.4121	0.3330	3.2086	50.9344	25.0184
	DQN	-12.2067	-0.4833	3.2078	50.9217	28.7017
AMZN	Buy and Hold	-10.7739	-0.4980	2.7697	43.9674	33.6828
	FINMEM	**4.8850***	**0.2327**	2.6872	42.6576	**22.9294**
	Generative Agents	-13.9271	-0.9981	1.7864	28.3576	27.7334
	FinGPT	-29.6781	-2.1756	**1.7464**	**27.7225**	28.4838
	A2C	-6.3591	-0.2938	2.7706	43.9819	26.1275
	PPO	-8.4194	-0.3891	2.7702	43.9761	33.6828
	DQN	-29.9820	-1.3906	2.7603	43.8177	38.3740
MSFT	Buy and Hold	14.6949	0.8359	2.2326	35.4411	15.0097
	FINMEM	**23.2613***	**1.4402**	2.0512	32.5617	14.9889
	Generative Agents	-18.1031	-1.6057	**1.4318**	**22.7285**	24.2074
	FinGPT	5.7356	0.4430	1.6442	26.1008	**12.8459**
	A2C	0.4598	0.0261	2.2357	35.4913	23.6781
	PPO	12.8067	0.7282	2.2333	35.4532	19.5355
	DQN	14.7397	0.8385	2.2326	35.4408	25.1845
COIN	Buy and Hold	-30.0071	-0.5150	6.7517	107.1795	60.5084
	FINMEM	**34.9832***	**0.7170**	5.6538	89.7515	35.7526
	Generative Agents	3.4627	0.0896	**4.4783**	**71.0908**	**32.0957**
	FinGPT	-88.7805	-1.9507	5.2736	83.7153	73.5774

Table 1: Overall trading performance comparison during testing period between FINMEM and other algorithmic agents.* *indicates that the result of Wilcoxon signed-rank test is statistically significant.*[3]

4.1 DATASET AND IMPLEMENTATION DETAILS:

We assessed FINMEM's performance using multi-source financial data from 2021-08-15, to 2023-04-25, sourced from reputable financial databases and APIs like Yahoo Finance (yfinance) and Alpaca News API, detailed explained in Figure 3 of Appendix B. The stock tickers for our experiments are detailed in Table 5 of Appendix D. These were selected because they are among those with the highest volumes of accessible news text data, and they are spread across various trading sectors.

In the Trading Agents Comparison, FINMEM employs GPT-4-Turbo as its backbone LLM, with the temperature parameter set at 0.7 to balance content consistency and creativity. Training occurred on financial data spanning 2021-08-17 to 2022-10-05, followed by testing from 2022-10-06 to 2023-04-10. The training period was chosen to account for the seasonal nature of corporate financial reporting and the duration of data retention in FINMEM's memory module. The selected training duration ensures to include at least one publication cycle of either Form 10-Q (classified as intermediate memory) or Form 10-K (regarded as deep memory), occasionally encompassing both. This strategy ensures that the experiences retained in FINMEM are still influential during the testing phase for a significant period. We limited the retrieval from each layer to the top 5 memory events. FINMEM operated under three distinct risk inclination settings, and we report performance based on the setting yielding the highest cumulative return in tests. To maintain consistency in the comparison, both training and testing phases for the other two LLM-based agents were synchronized with FINMEM's train-test split. Parameters unique to these agents, not covered by FINMEM's configuration, remained as defined in their original source codes.

We also comprehensively evaluate FINMEM's trading efficacy against a selection of leading DRL trading agents, A2C, PPO, and DQN. Additionally, we compare FINMEM with two SOTA LLM-based financial agents: FinGPT (Yang et al. (2023)) and Generative Agents(GA) (Park et al. (2022)) and the Buy and Hold (B&H) benchmark strategy. Considering that DRL algorithms need extensive training data for stable and converged results, and given our daily evaluation of trading performance, we extended the DRL agents' training period to roughly a 10-year span, from 2012-01-01 to 2022-10-05, for a fair comparison. The testing period was kept consistent with the other models. The DRL algorithms were implemented using Stable Baselines 3 (Raffin et al. (2021)). Our evaluation of FINMEM's investment performance versus other algorithmic trading agents uses five financial metrics: Cumulative Return, Sharpe Ratio, Annualized Volatility, Daily Volatility, and Max Drawdown. We averaged metrics across five trials in experiments and ablation studies. Metric details and

[3] The bold numbers in this and subsequent tables signify the best performance for the respective metrics.

configurations are in Appendix E.

4.2 Algorithmic Trading Agents Comparison (RQ1 & RQ2)

In this experiment, we evaluate the stock trading effectiveness of FinMem against competing models, focusing on equities from five diverse trading sector companies: Tesla, Inc. (TSLA), Netflix, Inc. (NFLX), Amazon.com, Inc. (AMZN), Microsoft Corporation (MSFT), and Coinbase Global, Inc. (COIN). The consolidated performance of various algorithmic trading agents, based on five main metrics, is presented in Table 1. Considering the critical role of Cumulative Return in measuring trading success over time, detailed time series plots are provided in Appendix H. Notably, FinMem's trading performance for COIN was solely compared with LLM trading agents and the baseline model. This is attributed to COIN's recent IPO in April 2021, which led to a lack of sufficient trading history for generating stable results with DRL algorithms. These plots reveal the progression of Cumulative Return for each company during the test phase, providing a comprehensive performance comparison.

In response to **RQ1**, the trading outcomes presented in Table 1 reveal that FinMem outperforms all other algorithmic trading agents and the B&H baseline strategy in terms of Cumulative Return and Sharpe Ratio. FinMem's superiority is statistically significant when compared to the second-best trading strategy. Specifically, for TSLA and NFLX, FinMem's strategy achieves Sharpe Ratios exceeding 2.0 and Cumulative Returns surpassing 35% while maintaining the lowest Volatility and Max Drawdown. These indicators underscore FinMem's ability to generate higher returns per unit of risk. In the case of MSFT, FinMem also records a Sharpe Ratio above 1.0 and a Cumulative Return over 20%, coupled with relatively low Volatility and Max Drawdown, demonstrating its impressive trading performance. For AMZN and COIN, FinMem consistently delivers positive Cumulative Returns and superior Sharpe Ratios, outperforming other strategies that yield negative values for these metrics. Additionally, its Volatility and Max Drawdown are on the lower end. Hence, these results collectively demonstrate FinMem's robust trading performance across a diverse range of trading sectors. Specifically, FinMem exhibits superior performance compared to the two other LLM agents in our study, FinGPT and GA. This underscores the effectiveness of FinMem's unique profiling and memory structure, which are particularly tailored for LLM agents dealing with financial data, significantly enhancing their investment decision-making capabilities.

In response to **RQ2**, the main challenge for DRL trading agents is their need for extensive training data over long periods, a requirement not met by stocks with brief historical records. As Table 1 indicates, FinMem outperforms DRL agents, achieving superior results in shorter training times, even when DRL agents are trained on data spanning nearly a decade. This attribute renders FinMem particularly beneficial for companies like COIN. with limited trading history, where DRL agents often struggle with data insufficiency. Furthermore, FinMem distinguishes itself among LLM-based agents designed for brief training, as detailed in Appendix H.

To further evaluate FinMem's performance with limited training data, we compressed the training period to 2021-08-17 to 2022-02-10, extending the testing phase to 2023-04-25, focusing on TSLA due to its extensive news volume. FinMem's trading outcomes are detailed in Figure 7 of Appendix H. Notably, with less than six months of daily data, including one Form 10-K and one Form 10-Q, FinMem achieved consistently high gains and the highest cumulative return on 2022-12-28.

The notable trading efficacy of FinMem is largely due to its advanced profiling and memory module design. This innovative architecture allows FinMem to seamlessly assimilate and prioritize crucial information from diverse data types, including textual and numerical inputs. The adaptability of its profiling module in adjusting to different risk preferences is key to its success in both capitalizing on market upswings and protecting investments during declines. A representative case is observed with TSLA, where FinMem's optimal trading outcomes were achieved through a self-adaptive risk mechanism. This mechanism endows FinMem with the capability to adopt a cautious approach in response to negative short-term cumulative returns, while switching to a more aggressive stance during periods of positive returns, thereby avoiding excessive passivity. This self-adaptive risk incliation has been largely effective across various stocks, with the exception of MSFT, as detailed in Appendix F. For MSFT, embracing a risk-seeking approach aligned better with its overall bullish market trend. Furthermore, the distinct functionalities of the memory module, such as differentiated retention durations for various types of information and key memory transitions, empower FinMem to discern and retain critical data for informed investment strategies.

Metric	B&H	Self-Adaptive	Risk Seeking	Risk Averse
Cumulative Return (%)	-66.9497	**54.6958**	-19.4132	-12.4679
Sharpe Ratio	-2.0845	**2.4960**	-0.7866	-1.5783
Daily Volatility (%)	3.9527	2.7419	3.2722	**1.7744**
Annualized Volatility (%)	3.8050	2.5960	2.9236	**0.9358**
Max-Drawdown (%)	67.3269	**12.5734**	45.0001	15.9882

Table 2: Comparison of overall trading performance during the testing period with different risk inclinations setting.

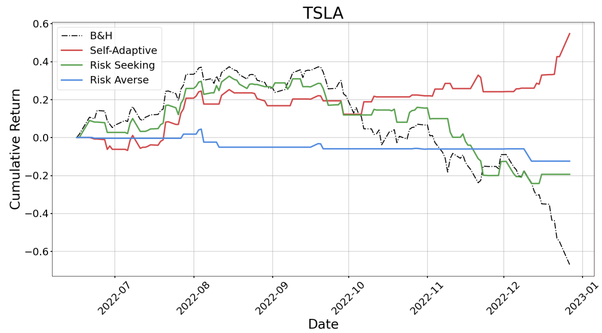

Figure 2: Comparative Cumulative Return by different risk inclination in FINMEM's profiling module.

4.3 DYNAMIC CHARACTER DESIGN EVALUATION (RQ 3)

In the secone experiment, we focused on evaluating the influence of FINMEM's profiling module on its trading effectiveness. As depicted in Appendix F, we equipped FINMEM with three distinct risk profiles: risk-seeking, risk-averse, and a self-adaptive character. We executed a comparative analysis of FINMEM's performance across these risk profiles, maintaining consistency in all other settings as outlined in Section 4.1. We chose TSLA to conduct this assessment because it possesses the largest amount of textual data, offering sufficient information to assess performance differences among varying types of character design. This study was run with a more compact training period from 2022-03-14 to 2022-06-15 and a testing period from 2022-06-16 to 2022-12-28. This shorter duration was chosen for budgetary efficiency.

In response to **RQ3**, Table 2 delineates the varied trading performance across different risk profiles. The self-adaptive profile enabled FINMEM to achieve the most favorable trading performance, as it was the only one to secure a positive Cumulative Return and a Sharpe Ratio exceeding 2.0, along with the least Max Drawdown.In contrast, the risk-seeking profile, while beneficial during a stable or bullish market as evidenced by MSFT's performance in Figure 8 of Appendix H, exhibited increased volatility and a decline in the face of a market downturn. Under the self-adaptive risk option, FINMEM shifts to a risk-averse setting. When the cumulative return over the past 3 days turned negative, it often chose to maintain existing positions. This strategy led to a Cumulative Return trajectory that typically trailed the market baseline, indicating a tendency towards excessive caution, which restrained trading actions in a bullish market. Overall, the dynamic character options feature, especially the risk profiles, equips FINMEM with flexible decision support in the varied market landscape.

5 CONCLUSION AND FUTURE WORK

In this paper, we introduce FINMEM, an innovative trading agent with an adjustable cognitive memory and dynamic character design. This framework enhances trading performance using real-world financial datasets, as evidenced by our experiments. FINMEM stands out with its human-like memory and dynamic character, enabling effective handling of complex financial data and adaptability to new scenarios. Its memory module surpasses other LLM trading agents in processing and organizing financial data into an evolving long-term memory. The dynamic character design, coupled with multiple risk profiles, empowers FINMEM to filter impactful financial data efficiently.

FINMEM excels in converting diverse financial data into effective investment strategies, benefiting from reduced training times. This is particularly advantageous for trading with new companies. Although currently using general-purpose LLMs, FINMEM is fully compatible with LLMs specifically fine-tuned for financial applications. It is expected that FINMEM's trading effectiveness will be further enhanced through the use of a more extensive and higher-quality dataset, in conjunction with LLMs that are specially designed for financial scenarios.

REFERENCES

Ang, A. and Chen, J. (2003). Downside risk. *Journal of Portfolio Management*, 29(4):103–112.

Avramelou, L., Nousi, P., Passalis, N., and Tefas, A. (2023). Deep reinforcement learning for financial trading using multi-modal features. *Expert Systems with Applications*, page 121849.

Balhara, S., Gupta, N., Alkhayyat, A., Bharti, I., Malik, R. Q., Mahmood, S. N., and Abedi, F. (2022). A survey on deep reinforcement learning architectures, applications and emerging trends. *IET Communications*.

Chen, T.-L. (2012). Forecasting the taiwan stock market with a novel momentum-based fuzzy time-series. *Review of Economics & Finance*, 2:38–50.

Chen, Y.-F. and Huang, S.-H. (2021). Sentiment-influenced trading system based on multimodal deep reinforcement learning. *Applied Soft Computing*, 112:107788.

Cochrane, J. H. (1988). Volatility tests and efficient markets: A review essay. *Journal of Monetary Economics*, 22(3):463–485.

Craik, F. I. and Lockhart, R. S. (1972). Levels of processing: A framework for memory research. *Journal of verbal learning and verbal behavior*, 11(6):671–684.

Dang, Q.-V. (2019). Reinforcement learning in stock trading. In *International conference on computer science, applied mathematics and applications*, pages 311–322. Springer.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Ethayarajh, K. (2019). How contextual are contextualized word representations? comparing the geometry of bert, elmo, and gpt-2 embeddings. *arXiv preprint arXiv:1909.00512*.

Fang, B. and Zhang, P. (2016). *Big Data in Finance*, pages 391–412. Springer International Publishing, Cham.

Gershman, S. J. and Ölveczky, B. P. (2020). The neurobiology of deep reinforcement learning. *Current Biology*, 30(11):R629–R632.

Huang, Z., Gutierrez, S., Kamana, H., and MacNeil, S. (2023). Memory sandbox: Transparent and interactive memory management for conversational agents. *arXiv preprint arXiv:2308.01542*.

Hull, J. (2007). *Risk Management and Financial Institutions*. John Wiley & Sons.

Jangmin, O., Lee, J., Lee, J. W., and Zhang, B.-T. (2006). Adaptive stock trading with dynamic asset allocation using reinforcement learning. *Information Sciences*, 176(15):2121–2147.

Liffiton, M., Sheese, B., Savelka, J., and Denny, P. (2023). Codehelp: Using large language models with guardrails for scalable support in programming classes. *arXiv preprint arXiv:2308.06921*.

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

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

Liu, X.-Y., Yang, H., Gao, J., and Wang, C. D. (2021). FinRL: Deep reinforcement learning framework to automate trading in quantitative finance. *ACM International Conference on AI in Finance (ICAIF)*.

Millea, A. (2021). Deep reinforcement learning for trading—a critical survey. *Data*, 6(11):119.

Miller, G. A. (1956). The magical number seven, plus or minus two: Some limits on our capacity for processing information. *Psychological review*, 63(2):81.

Mnih, V., Badia, A. P., Mirza, M., Graves, A., Lillicrap, T., Harley, T., Silver, D., and Kavukcuoglu, K. (2016). Asynchronous methods for deep reinforcement learning. In *International conference on machine learning*, pages 1928–1937. PMLR.

Mnih, V., Kavukcuoglu, K., Silver, D., Graves, A., Antonoglou, I., Wierstra, D., and Riedmiller, M. (2013). Playing atari with deep reinforcement learning. *arXiv preprint arXiv:1312.5602*.

OpenAI (2023). Gpt-4 technical report.

Park, J. S., O'Brien, J., Cai, C. J., Morris, M. R., Liang, P., and Bernstein, M. S. (2023). Generative agents: Interactive simulacra of human behavior. In *Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology*, UIST '23, New York, NY, USA. Association for Computing Machinery.

Park, J. S., Popowski, L., Cai, C., Morris, M. R., Liang, P., and Bernstein, M. S. (2022). Social simulacra: Creating populated prototypes for social computing systems. In *Proceedings of the 35th Annual ACM Symposium on User Interface Software and Technology*, pages 1–18.

Pätäri, E. and Vilska, M. (2014). Performance of moving average trading strategies over varying stock market conditions: the finnish evidence. *Applied Economics*, 46(24):2851–2872.

Pricope, T.-V. (2021). Deep reinforcement learning in quantitative algorithmic trading: A review. *arXiv preprint arXiv:2106.00123*.

Radford, A., Narasimhan, K., Salimans, T., Sutskever, I., et al. (2018). Improving language understanding by generative pre-training.

Raffin, A., Hill, A., Gleave, A., Kanervisto, A., Ernestus, M., and Dormann, N. (2021). Stable-baselines3: Reliable reinforcement learning implementations. *Journal of Machine Learning Research*, 22(268):1–8.

Schulman, J., Wolski, F., Dhariwal, P., Radford, A., and Klimov, O. (2017). Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*.

Sharpe, W. F. (1994). The sharpe ratio. *The Journal of Portfolio Management*, 21(1):49–58.

Shi, Y., Li, W., Zhu, L., Guo, K., and Cambria, E. (2021). Stock trading rule discovery with double deep q-network. *Applied Soft Computing*, 107:107320.

Sumers, T., Yao, S., Narasimhan, K., and Griffiths, T. L. (2023). Cognitive architectures for language agents. *arXiv preprint arXiv:2309.02427*.

Sun, R. (2004). Desiderata for cognitive architectures. *Philosophical psychology*, 17(3):341–373.

Sweller, J. (2012). Human cognitive architecture: Why some instructional procedures work and others do not.

Touvron, H., Lavril, T., Izacard, G., Martinet, X., Lachaux, M.-A., Lacroix, T., Rozière, B., Goyal, N., Hambro, E., Azhar, F., et al. (2023). Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*.

Vaidya, R. (2020). Moving average convergence-divergence (macd) trading rule: An application in nepalese stock market" nepse". *Quantitative Economics and Management Studies*, 1(6):366–374.

Wang, L., Ma, C., Feng, X., Zhang, Z., Yang, H., Zhang, J., Chen, Z., Tang, J., Chen, X., Lin, Y., et al. (2023). A survey on large language model-based autonomous agents. *arXiv preprint arXiv:2308.11432*.

Wu, S., Irsoy, O., Lu, S., Dabravolski, V., Dredze, M., Gehrmann, S., Kambadur, P., Rosenberg, D., and Mann, G. (2023). Bloomberggpt: A large language model for finance. *arXiv preprint arXiv:2303.17564*.

Wu, X., Chen, H., Wang, J., Troiano, L., Loia, V., and Fujita, H. (2020). Adaptive stock trading strategies with deep reinforcement learning methods. *Information Sciences*, 538:142–158.

Xiong, Z., Liu, X.-Y., Zhong, S., Yang, H., and Walid, A. (2018). Practical deep reinforcement learning approach for stock trading. *arXiv preprint arXiv:1811.07522*, pages 1–7.

Yang, H., Liu, X.-Y., and Wang, C. D. (2023). Fingpt: Open-source financial large language models. *arXiv preprint arXiv:2306.06031*.

Yang, H., Liu, X.-Y., Zhong, S., and Walid, A. (2020). Deep reinforcement learning for automated stock trading: An ensemble strategy. In *Proceedings of the first ACM international conference on AI in finance*, pages 1–8.

Zhao, W. X., Zhou, K., Li, J., Tang, T., Wang, X., Hou, Y., Min, Y., Zhang, B., Zhang, J., Dong, Z., et al. (2023). A survey of large language models. *arXiv preprint arXiv:2303.18223*.

A Decision-making Module Workflow and LLM Prompt Template

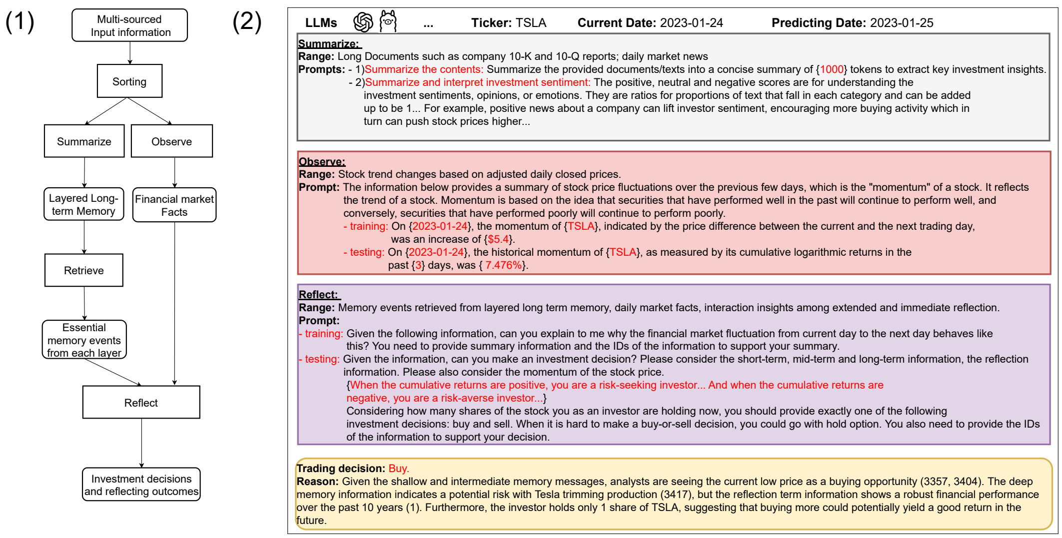

Figure 3: (1) The decision-making module workflow of the FinMem trading agent retrieves critical memory events to inform specific decisions. (2) LLM prompt template used by FinMem to interact with incoming financial information.

B Raw data sources

Data Sources
News data associated with ticker: News data is sourced from the Alpaca News API, which utilizes Benzinga as its backend provider.
Form 10-Q, Part 1 Item 2 (Management's Discussion and Analysis of Financial Condition and Results of Operations): Quarterly reports (Form 10-Q) are required by the U.S. Securities and Exchange Commission (SEC).
Form 10-k, Section 7 (Management's Discussion and Analysis of Financial Condition and Results of Operations): Annual reports (Form 10-K) are required by the U.S. Securities and Exchange Commission (SEC), sourced from EDGAR, and downloaded via SEC API.
Historical stock price: Daily open price, high price, close price, adjusted close price, and volume data from Yahoo Finance.

Table 3: Raw data and memory warehouses of FinMem

C Prompt Template for FinMem's Profiling Module

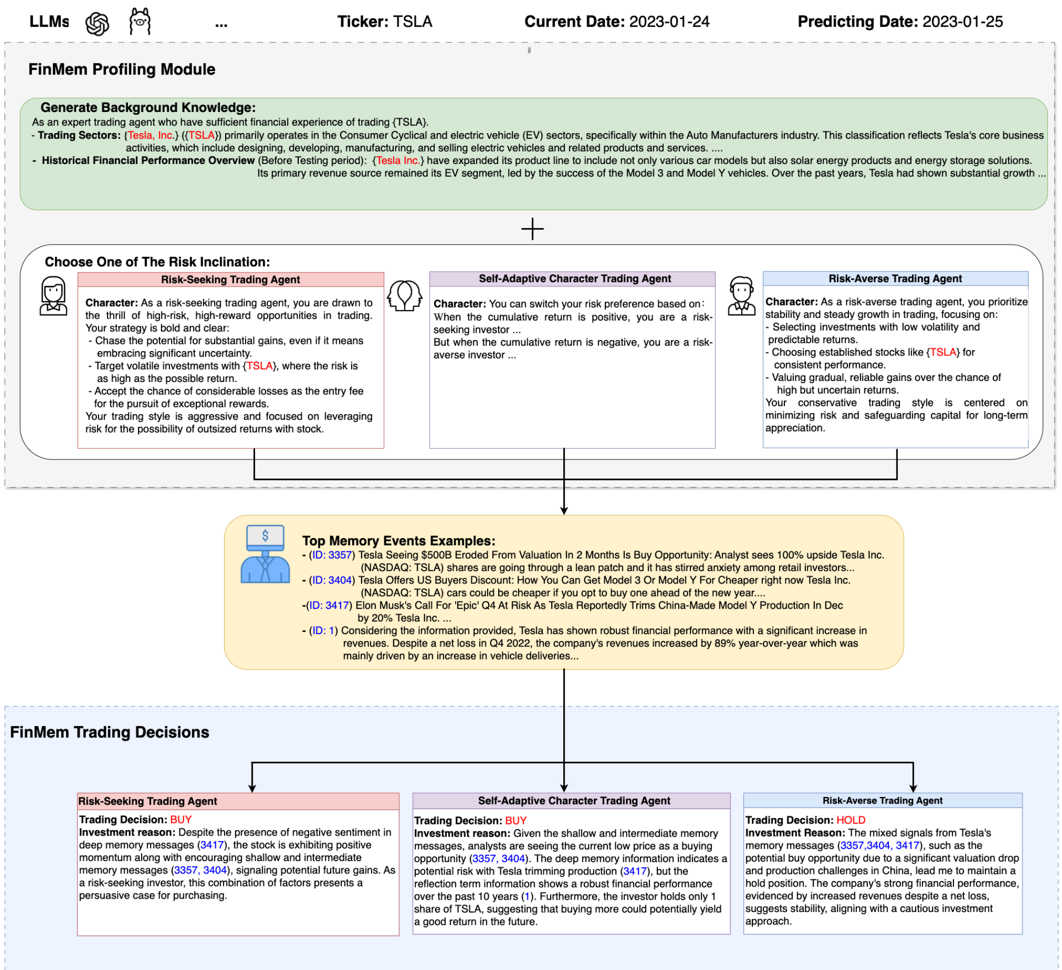

Figure 4: The prompt template for FinMem's profiling module. It includes two key elements of its character setting: professional background knowledge and three distinct investment risk inclinations. In the self-adaptive risk inclination option, the omitted texts align with the detailed descriptions provided for the risk-seeking and risk-averse inclinations.

D Distribution of News

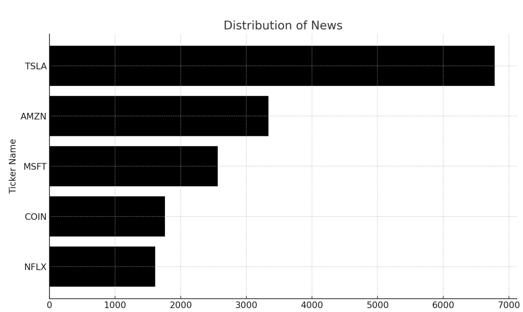

Figure 5: The distribution of news in scraped from Alpaca News API for the five stocks in the experiments

E FORMULA OF CLASSIC FINANCIAL METRICS FOR EVALUATING TRADING PERFORMANCE

Cumulative Return (Hull (2007)): Cumulative Return is a key trading performance metric because it provides a comprehensive insight into investment performance, especially for strategies that emphasize long-term growth and reinvestment. The effectiveness of different investment strategies is evaluated based on their Cumulative Returns, which reflect the total change in value over time. In this study, we compute Cumulative Returns over the specified period by summing daily logarithmic returns, as outlined in Equation 7. This method is widely accepted in the finance area due to its ability to precisely capture minor price fluctuations and symmetrically address gains and losses. In essence, a higher Cumulative Return typically indicates a more effective strategy.

$$\begin{aligned}\textbf{Cumulative Return} &= \sum_{t=1}^{n} r_i \\ &= \sum_{t=1}^{n} \left[\ln \left(\frac{p_{t+1}}{p_t} \right) \cdot \text{action}_t \right], \end{aligned} \tag{7}$$

where r_i represents the logarithmic return for day $t+1$, p_t is the closing price on day t, p_{t+1} is the closing price on day $t+1$, and action_t denotes the trading decision made by the model for that day.

Sharpe Ratio (Sharpe (1994)): Sharpe Ratio is another core metric for evaluating investment performance and adjusting returns for risk. It is calculated by dividing the portfolio's average excess return (R_p) over the risk-free rate (R_f) by its volatility (σ_p), as shown in Equation 8. This metric adjusts returns for risk, with a higher ratio indicating better risk-adjusted performance. Essential in comparing different portfolios or strategies, it contextualizes performance against similar investments. Although a Sharpe Ratio above 1 is typically considered favorable and above 2 as excellent, these benchmarks can vary depending on the context of comparison.

$$\textbf{Sharpe Ratio} = \frac{R_p - R_f}{\sigma_p} \tag{8}$$

Annualized Volatility and Daily Volatility(Cochrane (1988)): Annualized Volatility (Annum-Volatility) is calculated as the **Daily Volatility** (standard deviation of daily logarithmic returns) multiplied by the square root of the typical number of trading days in a year (252) as outlined in Equation 9, is vital for assessing investment risk. This measure reflects the extent of fluctuation in a security or market index's returns over a year, indicating potential deviations from average returns. It's especially relevant for investors with specific risk profiles, such as those who are risk-averse, who may prefer portfolios demonstrating lower annualized volatility.

$$\textbf{Annum-Volatility} = \textbf{Daily Volatility} \times \sqrt{252} \tag{9}$$

Max Drawdown (Ang and Chen (2003)): Max Drawdown is a metric for assessing risk. It represents the most significant decrease in a portfolio's value, from its highest (P_{peak}) to its lowest point (P_{trough}) until a new peak emerges, detailed in Equation 10. Indicative of investment strategy robustness, a smaller Max Drawdown suggests reduced risk.

$$\textbf{Max Drawdown} = \max\left(\frac{P_{\text{peak}} - P_{\text{trough}}}{P_{\text{peak}}}\right) \tag{10}$$

F OPTIMAL RISK INCLINATION WITH DIFFERENT STOCKS

	Risk-Averse	Self-Adaptive	Risk-Seeking
TSLA		√	
NFLX		√	
AMZN		√	
MSFT			√
COIN		√	

Figure 6: The optimal risk inclination for FINMEM when trading different stocks.

G BASELINE AND COMPARATIVE MODELS:

We assess FINMEM's trading performance in comparison to five advanced algorithmic agents and a commonly accepted baseline trading strategy. Among these, three models employ Deep Reinforcement Learning (DRL) approaches, while the remaining two are based on Large Language LLMs. Brief descriptions of each are provided below:
Buy-and-Hold strategy (B&H):
A passive investment approach, where an investor purchases stocks and holds onto them for an extended period regardless of market fluctuations, is commonly used as a baseline for comparison of stock trading strategies.
DRL trading agents:
As the FINMEM is practiced and examined on the basis of single stock trading and discrete trading actions, we choose three advanced DRL algorithms fitting into the same scenarios according to the previous and shown expressive performance in the work of Liu et al. (2021; 2022). The DRL training agents only take numeric features as inputs.

- **Proximal Policy Optimization (PPO):** PPO (Schulman et al. (2017)) is employed in stock trading due to its stability and efficiency. One salient advantage of PPO is that it maintains a balance between exploration and exploitation by bounding the policy update, preventing drastic policy changes.

- **Deep Q-Network (DQN):** DQN (Mnih et al. (2013)) is an adaptation of Q-learning, that can be used to optimize investment strategies. Unlike traditional Q-learning that relies on a tabular approach for storing Q-values, DQN generalizes Q-value estimation across states using deep learning, making it more scalable for complex trading environments.

- **Advantage Actor-Critic (A2C):** A2C (Mnih et al. (2016)) is applied to optimize trading actions in the financial environment. It operates by simultaneously updating both the policy (actor) and the value (critic) functions, providing a balance between exploration and exploitation.

LLM trading agents:
We evaluate FINMEM against two LLM agents in the context of stock trading. The first LLM agent, known for its proficiency in general-purpose tasks, serves as a baseline. The second agent, a leading-edge LLM in trading, has been acclaimed for its promising performance in stock market operations.

- **General-purpose Generative Agents – GA:** The generative AI agent by Park et al. (2022), originally intended to simulate realistic human behavior and make everyday decisions, has been adapted here for specific stock trading tasks. This agent's architecture includes a memory module that employs recency, relevance, and importance metrics to extract pivotal memory events for informed decision-making. However, it does not provide a layered memory module to effectively differentiate the time sensitivities unique to various types of financial data. Additionally, although it features a profiling module to define agent attributes like professional background, the model does not include a mechanism for self-adaptive risk preference. In our experiments, we modified the original prompt template created by Park et al., which was intended for general daily tasks, to suit financial investment tasks. The textual elements of this revised template closely align with those of FINMEM, with the exception of two components that are absent in this version of general-purpose Generative Agents.

- **LLM trading agents – FINGPT:** A novel open-source LLM framework specialized for converting incoming textual and numeric information into informed financial decision-making, introduced by Yang et al. (2023). It claims superiority over the traditional buy-and-hold strategy.

H Cumulative return comparison over time of FinMem

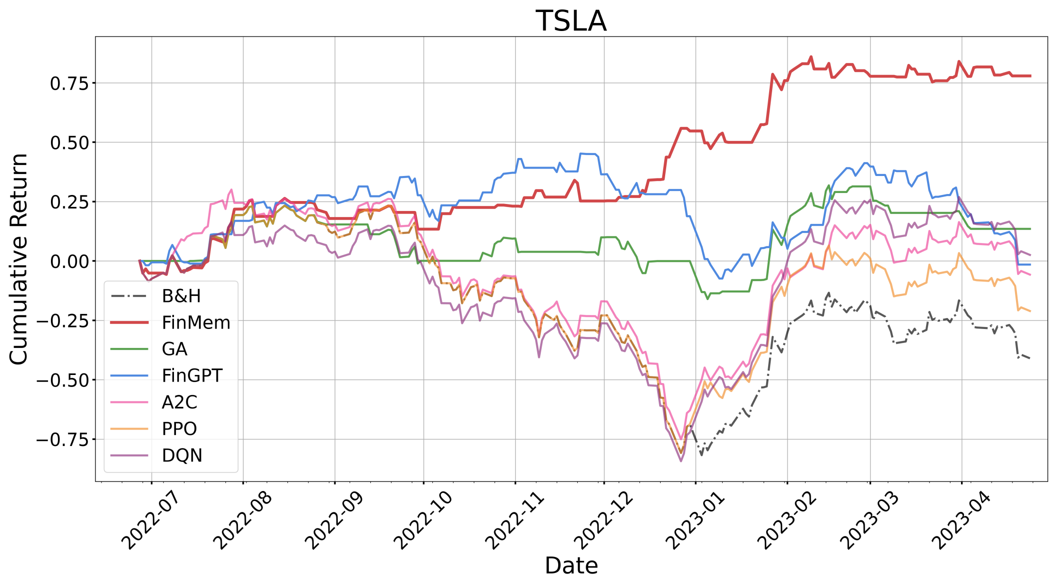

Figure 7: Cumulative Return of FinMem on trading Tesla, Inc. (TSLA) stock Over an Extended Testing Period.

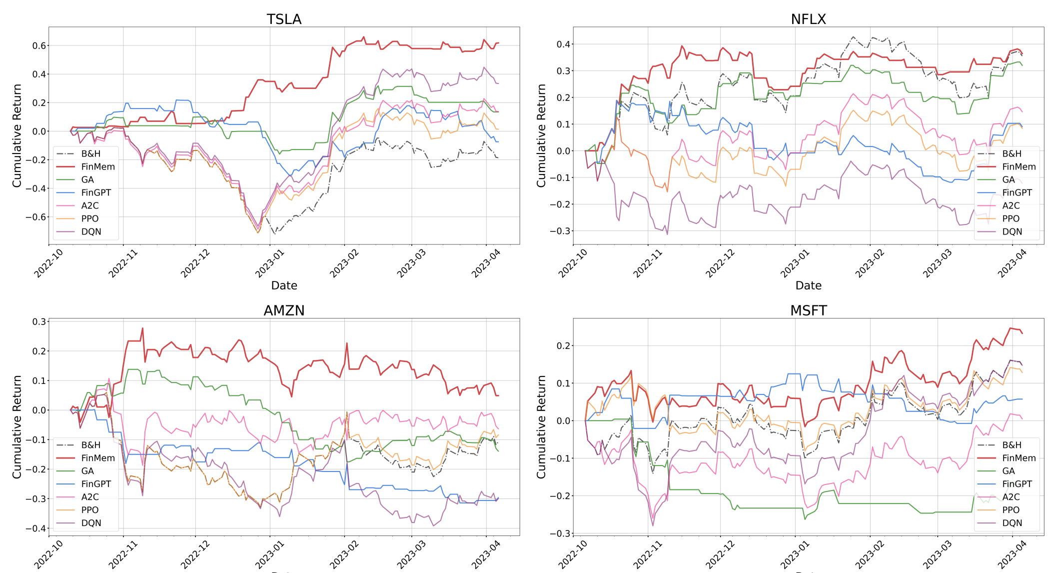

Figure 8: Cumulative return comparison over time between FinMem and other algorithmic agents across five stocks.

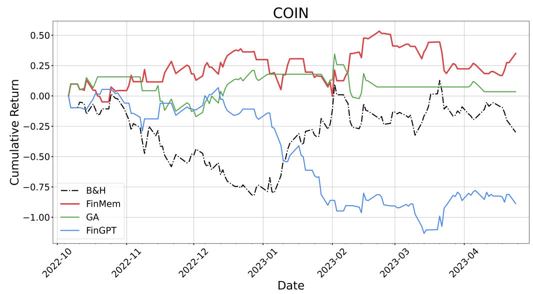

Figure 9: Cumulative Return comparison over time between FINMEM and other algorithmic agents on Coinbase Global, Inc. (COIN).

I FINMEM BACKBONE LLMS COMPARISON

Metric	B&H	GPT 3.5-Turbo	GPT4	GPT4-Turbo	davinci-003	Llama2-70b-chat
Cumulative Return (%)	-66.9497	16.1501	**62.6180**	54.6958	1.6308	-52.7233
Sharpe Ratio	-2.0845	2.1589	2.2251	**2.4960**	0.8515	-2.8532
Daily Volatility (%)	3.8050	0.8862	3.3339	2.5960	**0.2269**	2.1891
Annualized Volatility (%)	60.4020	14.0683	52.9237	41.2100	**3.6018**	34.7503
Max Drawdown (%)	67.3269	1.1073	17.4012	12.5734	**0.8408**	44.7168

Table 4: Comparison of trading performance during the testing period for FINMEM using different LLMs as backbone algorithms.

In this study, we evaluated the trading performance of FINMEM using various LLMs as its backbone algorithms. The LLMs under consideration included davinci-003, GPT 3.5-Turbo, GPT4, GPT4-Turbo, and Llama2-70b-chat. The parameter settings were consistent with its optimal performance in the comparative experiment detailed in Section 4.1, and the risk inclination was configured to be self-adaptive. The evaluated ticker is TSLA as in the Section 4.3. The results of this evaluation are compiled in Table 4.

The findings demonstrate that FINMEM, powered by GPT-4 and GPT-4 Turbo, delivered superior trading results during the test phase. Specifically, GPT-4 recorded the highest cumulative return, while GPT-4-Turbo exhibited the most favorable Sharpe Ratio. GPT 3.5-Turbo's performance was also noteworthy, following closely behind. As depicted in Figure 10, though slightly lower than market baseline (B&H), FINMEM with GPT-4-Turbo led in cumulative returns before October 2022. This period was characterized by relative stability and a modest upward trend in TSLA stock. After October 2022, with TSLA undergoing increased volatility and a notable downward trend, the cumulative return trajectory for FINMEM with GPT-4-Turbo exhibited significantly lower volatility and sustained stable returns not markedly lower than those of GPT-4. These results indicate that GPT-4 Turbo is the most suitable backbone algorithm for FINMEM.

FINMEM configured with davinci-003 and Llama2-70b-chat exhibited the lowest Annualized Volatility and Max Drawdown, yet their Cumulative Return and Sharpe Ratio were underwhelming. As illustrated in Figure 10, both models defaulted to a "Hold" strategy beyond a certain point during periods of intense fluctuation in TSLA stock. The unsatisfactory performance of davinci-003 may be attributed to its limited capability, as an earlier generation language model, to capture and understand nuanced yet decisive information.

We selected Llama2-70b-chat as it was deemed to possess stronger in-context learning and instruction-following capabilities compared to other Llama family models with fewer parameters, as noted in Zhao et al. (2023). Nonetheless, in the context of stock trading, it still demonstrated challenges in adequately comprehending key messages necessary for effective trading decisions. The comparatively poorer performance of Llama2-70b-chat can also be attributed to its shorter context window, especially when compared to the GPT models. When integrated with FINMEM, it needs to simplify prompts and shorten the length of retrieved memory insights, which could potentially result in some loss of context. The exceptional trading result demonstrated by GPT-4-Turbo across all models was a main factor in choosing it as the backbone algorithm for FINMEM in our earlier comparative analysis with other algorithmic trading agents.

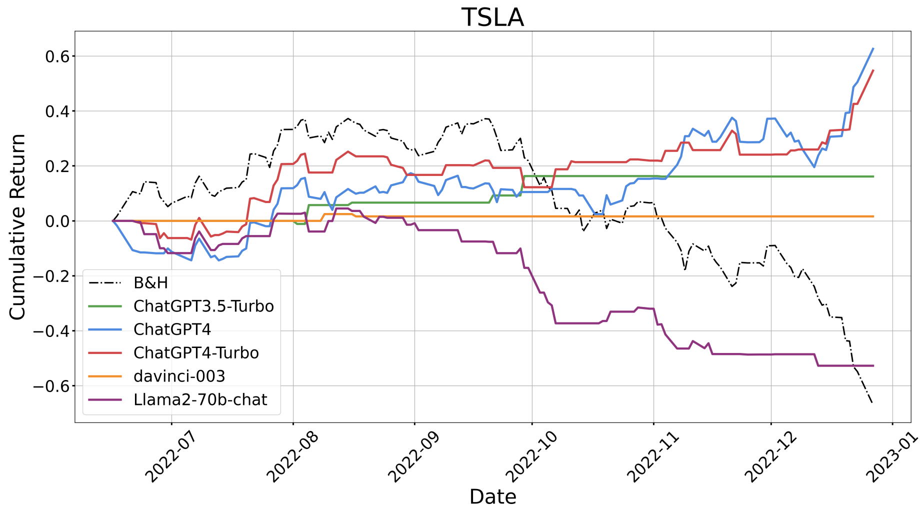

Figure 10: Comparison of overall Cumulative Returns over time for FinMem using different LLMs as backbone algorithms.

J Impact of adjusting the capacity of FinMem working memory

In this section, we explored whether appropriately tuning the memory retrieval bandwidth of FinMem could enhance its trading performance. This bandwidth is tied to the working memory's capacity within its memory module. As depicted in Figure 1, FinMem retrieves the top-K memory events from its long-term memory in response to a trading inquiry. By varying the K hyperparameter, FinMem can expand this capacity far beyond the human cognitive scope. We aimed to determine whether such flexibility in adjusting memory bandwidth translates to improvements in FinMem's performance.

Metric	B&H	Top 1	Top 3	Top 5	Top 10
Cumulative Return (%)	-66.9497	52.0936	29.4430	54.6958	**79.4448**
Sharpe Ratio	-2.0845	1.8642	1.1214	2.4960	**2.7469**
Daily Volatility (%)	3.8050	3.3105	3.1105	**2.5960**	3.4262
Annualized Volatility (%)	60.4020	52.5529	49.3779	**41.2100**	54.3891
Max Drawdown (%)	67.3269	25.2355	27.0972	**12.5734**	17.1360

Table 5: Comparison of overall trading performance during the testing period with different configurations of working memory capacity.

As demonstrated in Table 5, we adjusted the hyperparameter K to alter the number of memory events retrieved from shallow, intermediate, and deep long-term memory layers in FinMem. We tested K values of 1, 3, 5, and 10, exploring FinMem's working memory capabilities at levels below, near, and above the human cognitive limit. For all these K settings, we maintained a self-adaptive risk inclination, while other settings were consistent with those described in Section 4.1.
Across all K configurations, FinMem outperformed the Buy & Hold baseline, indicating the effectiveness of its memory module in processing diverse information and capturing critical events, which subsequently enhanced its trading performance, as evidenced by positive Cumulative Returns and Sharpe Ratios. Notably, higher K values, like 5 and 10, enabled FinMem to achieve the best Cumulative Returns and Sharpe Ratios exceeding 2.0. With K set to 1, FinMem still performed moderately well by capturing the most critical memory events of each layer.

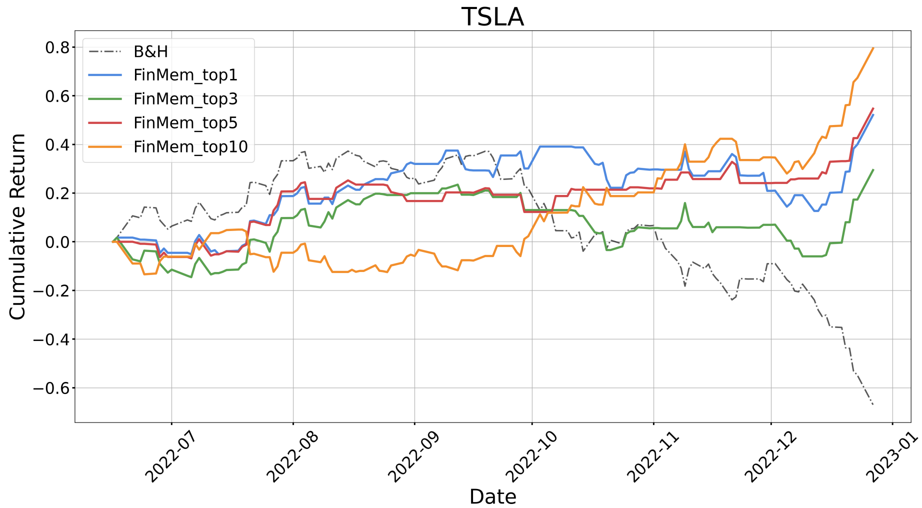

Figure 11: Cumulative Return over time for with different FinMem working memory capacity.

An in-depth analysis in Figure 11, which shows the Cumulative Return over time for various K settings, reveals that a K value of 5 is optimal for trading TSLA stock, consistently delivering robust performance with the lowest Volatility and Max-Drawdown. Before mid-October 2022, when the stock market was relatively stable and slightly upward, FinMem's trading actions aligned well with market trends (referring to B&H) and avoided significant losses. During periods of high volatility and continuous downturns (post-mid-October 2022), it maintained earnings by reducing "Buy" actions and favoring more "Hold" and "Sell" strategies. However, setting K to 10, while effective during market volatility, resulted in significant losses in stable market conditions. The issue may stem from the disproportionately loose capacity constraints on incoming information relative to the volume of incoming data. The broad memory retrieval bandwidth might have mixed trivial messages with critical ones, hampering FinMem's decision precision. This challenge becomes especially evident in neutral market conditions, where the influx of information includes a mix of varying market sentiments and trends.

Appropriately tuning the number of memory events (Top-K in the FinMem memory module can significantly enhance its trading performance. The aforementioned study illustrates that FinMem can achieve optimal results by effectively assimilating key signals from a sufficient quantity of filtered memories across each layer. However, the optimal value for K may vary depending on the volume and quality of incoming information.