



Explainable zero-shot trading using multi-agent LLM architecture: A backtested approach for Bitcoin price

Hae Sun Jung^a, Haein Lee^{b,*}

^a Department of Applied Artificial Intelligence, Sungkyunkwan University, 25-2 Sungkyunkwan-Ro, Jongno-Gu, Seoul, 03063, South Korea

^b School of Interdisciplinary Studies, Dongguk University-Seoul, 30 Phildong-ro 1-gil, Jung-gu, Seoul, 04620, South Korea

ARTICLE INFO

Keywords:

Large language models
Multi-agent systems
Zero-shot prompting
Cryptocurrency trading
Natural language reasoning

ABSTRACT

This study introduces a zero-shot, reasoning-based multi-agent trading framework utilizing large language models (LLMs) to integrate heterogeneous signals for Bitcoin trading over a 1400-day period. The framework combines specialized agents, each dedicated to a modality such as technical indicators, on-chain metrics, macroeconomic signals, and textual sentiment, with a meta-agent that synthesizes their rationales into coherent trading decisions without task-specific fine-tuning. Empirical evaluations using Bitcoin market data reveal that the proposed framework outperforms conventional time-series models over a short-horizon (three-day) period, achieving a 21.75 % total return (29.30 % annualized) and a Sharpe ratio of 1.08, surpassing the Long Short-Term Memory (LSTM) baseline by 1.70 percentage points in total return and 0.003 in Sharpe ratio. Ablation results reveal that Reddit-based sentiment enhances profitability (23.30 % total return), while news-based sentiment introduces semantic noise that degrades performance. All strategies are rigorously evaluated under realistic backtesting conditions, explicitly considering slippage and transaction costs to ensure reproducibility and fair comparisons. Beyond raw returns, systematic evaluation through an LLM-based evaluation protocol (G-EVAL) validates the consistency and interpretability of agent rationales, reinforcing model transparency. The proposed framework's modularity, interpretability, and robust empirical performance highlight its potential as an interpretable, scalable, and transparent approach to financial decision-making, aligning with the broader goals of explainable artificial intelligence in risk-sensitive financial systems.

1. Introduction

1.1. Motivation

Cryptocurrency markets are characterized by increasing volatility, heterogeneity, and information overload (Woebbeking, 2021). These dynamics necessitate trading systems that can integrate diverse data sources and dynamically adapt to changing conditions. Traditional algorithmic trading strategies primarily rely on time-series models that forecast future price movements based on historical market data such as price and volume. A wide range of time-series models, including recurrent neural networks (RNNs) such as Long Short-Term Memory (LSTM) and Gated Recurrent Unit (GRU), as well as more recent Transformer-based and linear architecture,

* Corresponding author.

E-mail addresses: jestiriel@g.skku.edu (H.S. Jung), lhi00034@dongguk.edu (H. Lee).

effectively capture temporal dependencies for price prediction (Jung et al., 2024; Chatterjee et al., 2024). However, these models remain fundamentally constrained in integrating and analyzing heterogeneous data sources such as on-chain activity, macroeconomic indicators, and investor sentiment in a unified and context-aware manner.

To overcome these limitations, previous studies enriched time-series prediction models by incorporating sentiment analysis outputs as auxiliary features. For instance, polarity scores derived from Financial Bidirectional Encoder Representations from Transformers (FinBERT) or rule-based methods such as VADER have been used to supplement predictive models with textual sentiment information (Gadi & Sicilia, 2024; Zou & Herremans, 2023). Although such hybrid approaches enhance predictive performance, they typically treat sentiment as a static numerical input, and do not support dynamic inference or transparent decision-making. Furthermore, they cannot infer coherent relationships among heterogeneous data sources in an interpretable manner.

In contrast, human traders often synthesize insights across multiple domains including technical indicators, market signals, blockchain activity, and textual data to make informed, context-aware decisions (Delfabbro et al., 2021). With the advent of large language models (LLMs), such integrative reasoning capabilities can be increasingly replicated within computational systems (Wei et al., 2022). LLMs can process both structured data and unstructured text, generate financially contextualized interpretations, and articulate their reasoning in natural language (Wu et al., 2023). This enables the development of agent-based architectures that not only make trading decisions, but also produce transparent, human-interpretable justifications (Park et al., 2023).

While LLMs are increasingly being explored for real-time decision-making in traditional financial markets, their application to cryptocurrency trading is still at an early stage (Singh & Bhat, 2024). Existing studies have primarily focused on isolated tasks such as sentiment classification, entity recognition, or news summarization (Kulbhaskar & Subramaniam, 2023), rather than constructing end-to-end trading agents. Even recent LLM-based trading approaches, including those by Kumar et al. (2024), Makri et al. (2025), and Wang et al. (2024), are often constrained by restrictive setups. These models typically rely on univariate price inputs, short-horizon prediction windows, or fine-tuning strategies with limited generalizability. They also lack modular role separation, interpretability mechanisms, and architectural flexibility. Furthermore, evaluations are frequently conducted under loosely specified or incomparable backtesting conditions, undermining the reproducibility and fairness of performance comparisons (Brauneis & Sahiner, 2024).

To address these limitations, this study proposes a novel multi-agent framework that integrates heterogeneous signals through LLMs for interpretable financial decision-making.

1.2. Research objectives

This study addresses the limitations of conventional time-series forecasting models and existing LLM-based financial applications by designing and validating a modular, interpretable, zero-shot reasoning framework for cryptocurrency trading. The primary research objectives are as follows:

- To develop a role-specific multi-agent architecture where each agent uses an LLM to process a distinct class of market signals, such as technical indicators, on-chain metrics, macroeconomic trends, and sentiment data.
- To examine the effectiveness of zero-shot LLMs in generating accurate, interpretable trading decisions without task-specific fine-tuning.
- To benchmark the proposed framework against deep learning-based time-series models across multiple trading horizons, using unified backtesting environment that reflects realistic execution conditions.
- To evaluate the contribution of LLM-based agents in enhancing transparency, semantic consistency and explanatory robustness are assessed through ablation testing and Generative Pretrained Transformers (GPT)-based evaluation (G-EVAL), with particular focus on agents handling unstructured text.

By pursuing these objectives, the study explores the potential of LLMs as dynamic, interpretable components in automated financial decision-making pipelines.

1.3. Contributions

The key contributions of this research are as follows:

- A novel multi-agent architecture that enables zero-shot reasoning by assigning LLMs to role-specific agents, each interpreting a distinct market signal. This design supports modular reasoning and transparent decision-making across heterogeneous data sources.
- A GPT-based meta-agent that synthesizes the natural language rationales of individual agents into unified trading actions with interpretable justifications, enhancing auditability and regulatory transparency.
- Textual agents that process textual data from Reddit and news articles, enabling comparative analysis of semantic coherence and marginal utility in trading. Experimental results show that Reddit-based sentiment improves short-horizon returns, while news signals introduce decision conflict.
- A rigorous backtesting protocol that evaluates the framework against LSTM, GRU, Decomposition Linear (DLinear), and Patch Time Series Transformer (PatchTST) under standardized execution timing, transaction cost assumptions, and prediction intervals.
- Integration of G-EVAL into the multi-agent trading framework to diagnose reasoning consistency and epistemic reliability, providing a structured approach to interpretability assessment in LLM-based financial applications.

Collectively, these contributions demonstrate that LLM-based multi-agent frameworks can achieve both quantitative competitiveness and interpretability, offering promising architecture for next-generation financial artificial intelligence (AI) systems in high-volatile domains such as cryptocurrency markets. In particular, the framework's primary value lies in its interpretability, offering decision processes that can be examined, critiqued, and trusted under realistic market conditions.

2. Background

The increasing complexity and volatility of cryptocurrency markets has prompted a growing body of research focused on predictive modeling and decision-making systems. This trend has also gained significant attention within the information and computing science communities, where cryptocurrency prediction is approached not only as a financial forecasting task but also as a challenge involving time-series analysis, unstructured data processing, and explainable AI (XAI) design.

Early approaches to cryptocurrency market prediction relied primarily on numerical and statistical methods, adapting classical time-series models such as AutoRegressive Integrated Moving Average (ARIMA), Generalized Autoregressive Conditional Heteroskedasticity (GARCH), and their variants to handle price volatility and non-stationarity. For instance, Abu Bakar and Rosbi (2017) used the ARIMA model to examine the predictability of Bitcoin exchange rates under high volatility. The model showed an explanatory power of approximately 44.44 % and a prediction error rate of 5.36 %. Cheikh et al. (2020) applied a smooth transition GARCH model and found a positive asymmetric response between returns and volatility in major cryptocurrencies. This pattern differs from traditional assets and suggests that cryptocurrencies exhibit safe-haven characteristics. Almansour et al. (2021) utilized the ARCH and GARCH models. Their analysis confirmed that past volatility significantly influences current volatility in the cryptocurrency markets. These early findings validate the application of quantitative modeling techniques in cryptocurrency markets, particularly in addressing volatility and trend estimation. While these models are effective in leveraging historical data such as price, volume, and volatility, they often fail to capture the nonlinear dynamics and sudden regime changes that characterize these markets.

Building on the limitations of traditional statistical models, recent studies leveraged advanced machine learning algorithms to classify market directions and predict short-horizon trends based on market features. For instance, Jung et al. (2023a) predicted Bitcoin price trends using 11 technical indicators with six machine learning models. The Extreme Gradient Boosting (XGBoost) model achieved an accuracy of 90.57 % and an area under the curve (AUC) of 97.48 %, demonstrating competitive predictive potential. Akyildirim et al. (2021) analyzed the predictability of machine learning classification algorithms using daily and minute-level data from 12 major cryptocurrencies. By incorporating historical prices and technical indicators as model features, they found that all algorithms consistently achieved 55–65 % accuracy, with Support Vector Machines (SVM) delivering the most consistent performance. Chowdhury et al. (2020) applied various ML techniques to predict the closing prices of cryptocurrency indices and their components, highlighting the practical value of ensemble learning. Lahmiri and Bekiros (2019) utilized deep learning models to forecast the prices of Bitcoin, Digital Cash, and Ripple, showing that LSTM networks capture both short- and long-term dependencies more effectively than conventional neural networks. Similarly, Patel et al. (2020) and Tanwar et al. (2021) developed hybrid LSTM-GRU models that outperformed single-model baselines in terms of predictive accuracy. Although these structured approaches demonstrate clear advantages for trend prediction, they often fail to account for market fluctuations driven by investor sentiment and information shocks, which are particularly influential in cryptocurrency markets.

To address this gap, recent studies explored the use of sentiment and attention signals derived from unstructured textual sources. For instance, Lamon et al. (2017) proposed a text-based model that effectively predicted major price surges and drops in Bitcoin and Ethereum by labeling news and social media data based on actual price movements. Wolk (2020) utilized Twitter-based sentiment analysis and a hybrid model to predict cryptocurrency prices at 10-minute intervals. A one-month trading experiment demonstrated that this approach outperforms existing automated trading tools in terms of profitability. Jain et al. (2024) combined Robustly Optimized BERT Approach (RoBERTa)-based sentiment analysis with Twitter metadata to predict the prices of major cryptocurrencies such as Solana and Avalanche. By integrating LSTM and regression models, they successfully captured the price fluctuation trends.

Given these strengths, recent research has increasingly explored how LLMs can be embedded within broader financial decision-making pipelines, not merely as sentiment analyzers but as core components of forecasting and trading systems. Recent advancements in LLMs offer a promising avenue for addressing key limitations of sentiment-based cryptocurrency forecasting. While prior approaches have incorporated sentiment signals by quantifying textual data into numerical scores, often through lexicon-based tools or pre-trained classifiers, these methods tend to reduce complex market narratives into static features, limiting both contextual sensitivity and interpretability. By contrast, LLMs can directly process unstructured financial texts, capture subtle semantic cues, and produce explicit reasoning traces without extensive feature engineering. These capabilities enable a more context-aware and explainable integration of textual information into financial prediction pipelines, supporting diverse applications such as environmental, social, and governance (ESG) evaluation, market trend explanation, and automated reporting (Lee et al., 2025a, 2025b; Le, 2024).

Kumar et al. (2024) utilized pre-trained language models in a neural architecture designed to jointly predict the cryptocurrency price direction and volatility. Their model combined LSTM-based temporal encoders with Transformer-based decoders, leveraging LLMs to enhance contextual feature extraction. Makri et al. (2025) conducted short-horizon and few-shot forecasting by partially freezing the layers of pre-trained LLMs such as GPT-2 and Large Language Model Meta AI version 2 (LLaMA-2) and fine-tuning the remaining layers on Ethereum time-series data. The input structure was univariate, utilizing only price information based on datasets from Kaggle and self-collected node-level data. Their results demonstrated superior prediction accuracy in terms of mean absolute error (MAE) and mean squared error (MSE) compared to conventional models such as LSTM, and PatchTST. Notably, LLaMA-3 achieved the highest precision in both short-horizon and few-shot settings. Luo et al. (2025) proposed an LLM-powered multi-agent framework for cryptocurrency portfolio management. Their system consisted of specialized agents responsible for tasks such as news

summarization, technical analysis, and risk assessment. The framework employed intra- and inter-module coordination mechanisms to exchange information and execute confidence-weighted ensemble strategies. The framework was primarily designed for investment decision making across the top 30 cryptocurrencies by market capitalization, and empirical results using real market data showed performance gains over single-agent baselines. Wang et al. (2024) introduced the multi-agent framework designed to enhance the trading performance of LLMs by separating reasoning processes into factual and subjective pathways. This system incorporates specialized agents for processing statistics, factual news, sentiments, and self-reflective reasoning, which are then integrated to generate final trading decisions. A key feature of this framework is the explicit separation of objective facts and subjective opinions in market news and adjust the influence of each type of information depending on the market regime, leading to improved trading outcomes. Although recent studies have made meaningful progress in applying LLMs to cryptocurrency forecasting and trading, each exhibits key limitations in terms of strategy design, signal integration, and evaluation rigor.

While prior research has made significant progress in applying LLMs to cryptocurrency forecasting, key limitations remain in signal modularity, interpretability, and evaluation rigor. Many existing frameworks lack explicit agent roles or rely on narrow input types, making them less adaptable and obscure. Additionally, inconsistent backtesting setups make it difficult to ensure fair performance comparison and reproducibility.

To address these limitations, this study presents a modular, interpretable, and zero-shot framework where role-specific LLM agents handle diverse market signals, and a GPT-based meta-agent synthesizes their output into coherent trading decisions. All strategies were rigorously evaluated under standardized backtesting conditions, including consistent prediction windows, execution count, and cost modeling, to ensure empirical robustness. Furthermore, agent-level ablation tests and G-EVAL assessments provide in-depth insight into reasoning transparency and semantic coherence, aligning the system with XAI principles crucial to real-world financial applications. Accordingly, the framework is designed with interpretability at its core, balancing transparency and semantic coherence with empirical performance to align with the practical requirements of explainable AI in finance.

3. Materials and methods

The overall workflow of the proposed trading system is shown in Fig. 1. The following subsections detail the data sources and modeling procedures that support each component of this architecture.

3.1. Data collection

This study utilizes five primary categories of data to support both traditional time-series models and LLM-based reasoning agents: (1) market price with volume, (2) on-chain indicators, (3) macro-financial variables, (4) social media posts, and (5) financial news articles. The structured datasets (market, on-chain, macro-financial) span four years (March 1, 2021, to February 28, 2025), whereas the unstructured textual datasets (Reddit and news) were collected only for 304 trading days (May 1, 2024, to Feb 28, 2025).

Market Data. Daily market data for both Bitcoin including open, high, low, and close (OHLC) prices and trading volumes, were

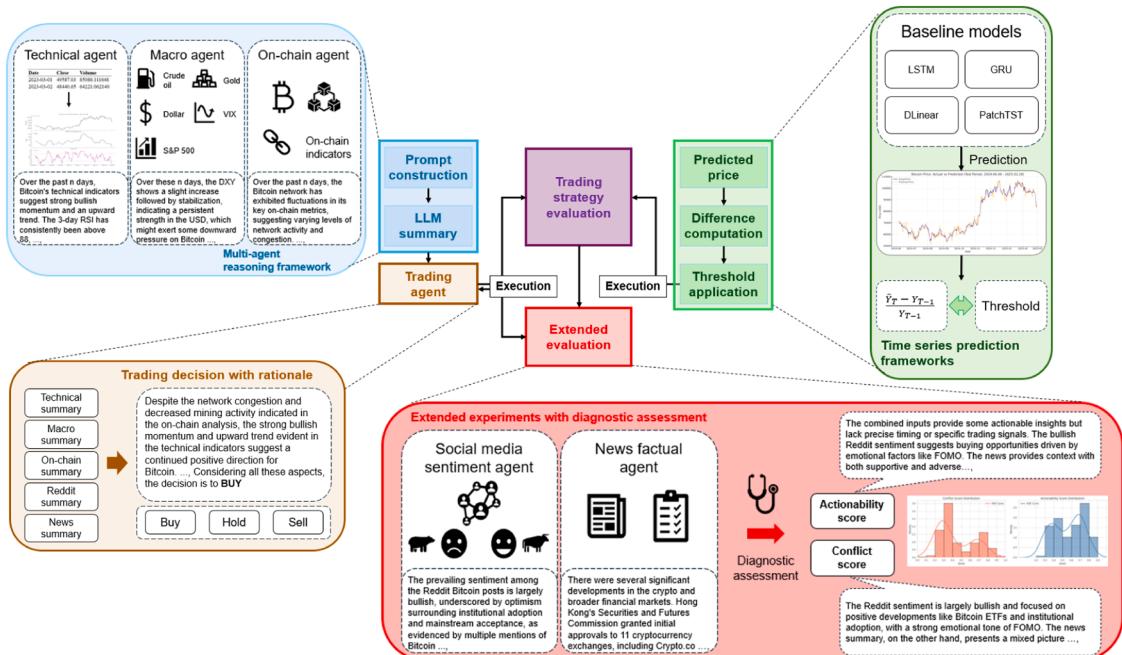


Fig. 1. Multi-agent framework for LLM-guided trading.

collected using the ccxt Python library, which aggregates price feeds from major cryptocurrency exchanges. Binance was chosen as the primary data source because of its high liquidity and trading volume, which ensure consistency and market representativeness (Jung et al., 2023b).

On-chain Indicators. Bitcoin on-chain data includes five metrics: confirmation time, hash rate, transaction count, transaction fees, and number of active addresses. All data were obtained from Bitinfocharts.com and aligned with the corresponding market series. These indicators capture network activity, congestion, and user participation, and were selected based on their predictive relevance in prior studies (Jung et al., 2024).

Macro-financial Variables. To capture global risk sentiment, we included daily closing prices for the U.S. Dollar Index (DXY), gold, the S&P 500 Index, the Volatility Index (VIX), and West Texas Intermediate (WTI) crude oil from Yahoo Finance. Since traditional markets are closed on weekends, values were forward-filled from the previous Friday through the weekend to maintain a strictly lag-aware alignment with the 7-day cryptocurrency calendar, thereby extending the raw 1007 days of macro-financial coverage to 1461 aligned records.

Textual Data: Reddit and News. Reddit posts were collected from cryptocurrency-related subreddits (e.g., r/Bitcoin) using Reddit Archive, and financial news articles mentioning Bitcoin were retrieved from LexisNexis. The Bitcoin corpus includes 67,075 Reddit posts and 8657 news articles. All entries were timestamped and aligned by date with structured series. Because textual data were incorporated only during the LLM evaluation phase, their effective temporal coverage was limited to 304 trading days within the four-year horizon.

Table 1 provides a summary of the datasets used in this study, including 1461 consecutive daily records for structured variables and a shorter, 304-day window for textual sources.

To integrate these heterogeneous sources, we merged all structured data using an inner-join strategy based on daily timestamps.

Unstructured text data were not integrated directly into the unified structured dataset but were instead aligned by date to enable later contextual associations. As a preliminary step, standard preprocessing was applied, including the removal of non-English entries, duplicate documents, and embedded Uniform Resource Locators (URLs). Further stratified sampling and input constraints for text-based modeling were applied during an extended evaluation phase ([Section 4.3](#)).

The resulting structured dataset spanned 1461 aligned daily records, each containing heterogeneous information suitable for time-series forecasting or reasoning-based decision making. The transformation into supervised model inputs (e.g., sliding windows and training/test splits) is detailed in [Section 4](#).

3.2. Technical indicator computation

To represent the technical patterns commonly exploited in algorithmic trading, a comprehensive set of technical indicators was computed from the Bitcoin price time-series using rolling statistics and exponential smoothing techniques implemented in Python (Peng et al., 2021). The indicators were categorized into the following three groups:

Trend-following indicators. These include the Simple Moving Average (SMA) and Exponential Moving Average (EMA), computed over 3-, 7-, 14-, 30-, 60-, and 120-day windows. The Moving Average Convergence Divergence (MACD) was also calculated using the 12- and 26-day EMAs, along with a nine-day signal line.

Momentum indicators. These include the Relative Strength Index (RSI), raw Momentum (defined as the absolute price change over time), and the Rate of Change (ROC) over a 10-day window.

Oscillators. These include the Stochastic RSI (featuring %K and %D lines), Stochastic Oscillator (based on a five-day range with three-day smoothing), and Williams %R calculated over a 14-day window.

The resulting indicators were utilized as input variables in the time-series prediction models and as input features for the technical indicator agent within the LLM-based decision framework. Due to the use of rolling windows, particularly the 120-day SMA/EMA, the effective analysis horizon was shortened from 1461 to 1341 days. This reduction reflects the initial data points lost during indicator initialization, and all reported results in [Section 4](#) are based on this 1341-day effective sample.

3.3. Baseline time-series prediction models

To establish quantitative benchmarks for predictive performance, four time-series regression models were implemented, each representing a distinct model architecture: LSTM, GRU, DLinear, and PatchTST (Cho et al., 2014; Hochreiter & Schmidhuber, 1997; Nie et al., 2022; Zeng et al., 2023). These models were selected to cover a methodological spectrum ranging from traditional recurrent networks to recent linear and Transformer-based architectures.

Table 1

Overview of data categories and coverage.

Data type	Data category	Source	Daily fields	Total days
Structured	Market data	Binance	OHLC	1461
	On-chain indicators	Bitinfocharts	5 fields	1461
	Macro-financial variables	Yahoo Finance	5 fields	1461 (Imputed)
Unstructured	Reddit posts	Reddit Archive	Posts	304
	News articles	LexisNexis	Articles	304

To account for the varying temporal dependencies in cryptocurrency markets, the input features were structured using sliding windows of 3, 14, and 30 days. The shortest window aims to capture short-horizon market fluctuations, while extended windows enable exploration of medium- and long-term contextual effects. All the models were trained in a supervised regression setting, with the objective of predicting a single scalar value corresponding to the next day's closing price of Bitcoin.

The dataset was chronologically split into training, validation, and test sets in a 64:16:20 ratio. Prior to model training, all the input features were normalized using MinMaxScaler fitted exclusively on the training set, and the resulting scaling parameters were applied to the validation and test sets. Preprocessing was conducted separately for each window size to ensure numerical stability and fair comparability across models.

The models are briefly described as follows:

- LSTM. The RNN architecture designed to capture long-term dependencies in sequential data using gated memory cells. It mitigates vanishing gradient problems and is widely adopted in the time-series prediction.
- GRU. A simplified variant of LSTM that combines forget and input gates into a single update gate, offering computational efficiency while retaining the temporal modeling capacity.
- DLinear. A lightweight model that decomposes a time-series into trend and seasonal components and fits them using linear regressors. Despite its simplicity, it performs competitively on various benchmarks.
- PatchTST. A Transformer-based model designed for long-term prediction. It uses non-overlapping patches and self-attention mechanisms to model both local and global dependencies.

These models provide price-centric prediction baselines, enabling an objective evaluation of the contribution of natural language reasoning within the proposed multi-agent framework.

3.4. Multi-agent reasoning framework

To enable reasoning across diverse data modalities, we implemented a multi-agent framework using GPT-4o. The architecture comprises multiple specialized agents, each responsible for processing a specific type of input, and a centralized trading agent that integrates their output into a final decision.

Alternative LLMs such as Claude, Gemini, and Financial GPT (FinGPT) were initially considered for the implementation of all agents (Luukkonen et al., 2023). GPT-4o was ultimately selected because recent independent evaluation demonstrated its superior ability to maintain consistency under hierarchical or conflicting instructions (Zhang et al., 2025), a capability critical for multi-agent coordination where diverse agent outputs must be aggregated coherently. Based on this foundation, the proposed framework assigns distinct reasoning roles to individual agents that specialize in different market modalities while maintaining centralized coordination through the GPT-4o-based trading agent.

Three domain-specific agents were employed:

Technical indicator agent. Analyzes time-series patterns in the computed technical indicators and price data using sliding windows of 3, 14, and 30 days. It produces natural language summaries that describe recent momentum dynamics and notable technical signals.

On-chain agent. Interprets key on-chain features to assess network-level activities and participation trends. Through variable-length sliding windows, it highlights behavioral patterns and shifts in on-chain activity.

Macro-financial agent. Processes macroeconomic indicators to capture shifts in global risk sentiment and economic outlook. It summarizes the potential implications of cryptocurrency market dynamics, including volatility spillovers.

Each agent independently generates an interpretable summary of recent market conditions based on its respective modality, without making explicit trading predictions. Subsequently, the final trading decisions are made by a centralized meta-agent that synthesizes outputs from upstream agents and integrates reasoning across diverse informational modalities to formulate a unified trading recommendation. At each decision point, the trading agent selects one of three actions ("BUY," "SELL," or "HOLD") and generates a textual explanation based on the summaries provided by the domain-specific agents.

To minimize hallucination risk, the framework was designed to operate based on fixed, domain-specific prompt templates rather than open-ended generation. Each agent was structured to produce concise, data-grounded summaries directly derived from structured numerical inputs, thereby suppressing any tendency toward speculative or unverifiable statements.

The meta-agent was designed to integrate these summaries without altering their factual content or introducing unsupported information, ensuring that synthesis was confined to the explicit reasoning provided by the agents. Additionally, 20 repeated experiments were conducted, and averaged results were reported to stabilize stochastic variation. This design ensured that final trading decisions reflected consistent and traceable multi-agent reasoning rather than uncontrolled text generation artifacts.

Additionally, two text-based agents were incorporated into extended experiments to capture the influence of unstructured textual information from different sources.

Social media sentiment agent. Extracts community sentiment from Bitcoin-related Reddit posts using a rolling window. It reflects informal investor opinions and the emotional tone of retail discourse.

News summarization agent. Processes Bitcoin-related news articles through a factual summarization pipeline to extract key developments and macroeconomic signals relevant to market behavior.

These two agents enable ablation studies to isolate the marginal contributions of factual news content and community-driven sentiment to overall trading performance. To enhance transparency, Appendix A provides representative prompt templates, Appendix B presents illustrative agent outputs. These materials collectively demonstrate the zero-shot reasoning process underlying the framework and ensure the reproducibility of results.

3.5. Trading strategy evaluation setup

To ensure a fair and causally aligned comparison between time-series models and LLM-based agent strategies, all approaches were evaluated within a unified backtesting framework under consistent market assumptions. This framework standardized the capital base, execution timing, transaction costs, and performance evaluation, thereby isolating the differences in reasoning mechanisms as the primary source of performance variation.

To examine temporal adaptability, three sliding window lengths (3, 14, and 30 days) were selected to represent short-, medium-, and long-term market horizons (Jung et al., 2024). The 3-day window captures high-frequency sentiment and liquidity shocks, the 14-day horizon reflects typical biweekly trading cycles observed in swing trading, and the 30-day setting represents monthly momentum and macroeconomic adjustment patterns. Evaluating these different horizons enables an interpretation of performance differences based on the trade-off between responsiveness and stability across market regimes.

The initial capital was set at 100,000 United States Dollar (USD), and only one trading decision was allowed per day. No leverage was employed. Each trade incurred a transaction fee of 0.1 %, which reflects the average maker-taker fee on major cryptocurrency exchanges such as Binance, thereby providing a realistic cost assumption. In addition, a slippage rate of 0.1 % was applied to all buy and sell operations to conservatively account for execution price deviations owing to market liquidity and order book depth. All strategies were tested over a common evaluation period using the same test dataset.

Decision Timing. All models generated trading signals using only the information available up to and including the close of day $t - 1$. For time-series prediction models, the predicted closing price of day t was compared with the observed close of day $t - 1$. A Buy signal was issued if the predicted change exceeded a positive threshold, and a sell signal was issued if the predicted change fell below a negative threshold. If the predicted change lay within the threshold band, no trade was executed and the previous position was held. This directional approach was chosen because relative price changes are more robust and realistic indicators than exact level forecasts.

Threshold Calibration. To mitigate overtrading driven by minor fluctuations, dynamic thresholds were applied based on the window size. A wider threshold of 1.5 % was used for short-horizon windows, reflecting higher noise levels, whereas a narrower threshold of 1.0 % was applied for longer horizons, where predictions tend to be more stable.

Execution and Profit Evaluation. Trading actions were executed at the opening price of day t , after accounting for transaction costs and slippage. Portfolio values were then updated at the closing price of the same day t , which was used to calculate realized profits and losses. This procedure ensured that no information from day t was available at the time of decision making, while performance was evaluated only with prices revealed after execution. The identical process was applied to LLM-based agent strategies: at each decision

Algorithm 1

Simplified trading algorithm.

Input:

- Initial capital C_0
- Daily OHLC market data
- On-chain, technical, macro-financial, Reddit, and news features
- Transaction fee rate f and slippage rate s
- Window size w for feature aggregation

Procedure:

1. For each day $t = 1 \dots T$:
 - a. Construct input features using data available up to day $t - 1$
 - b. For baseline models:
 - i. Predict closing price of day t
 - ii. Compute predicted return Δp relative to close of day $t - 1$
 - iii. Compare Δp with dynamic threshold θ (calibrated by window size)
 - iv. If $\Delta p > +\theta$: issue BUY
 - v. If $\Delta p < -\theta$: issue SELL
 - vi. Otherwise: HOLD previous position
 - c. For LLM-based agents:
 - i. Generate natural language summaries of each modality
 - ii. Provide summaries to the trading agent
 - iii. Trading agent outputs decision (BUY / SELL / HOLD) with rationale
 - d. Execute trade at the opening price of day t
 - Apply fee f and slippage s
 - e. Update portfolio value at the closing price of day t
 - Calculate realized profit and loss
2. Record daily portfolio values and returns

Output:

- Time-series of portfolio values
- Performance metrics: Total return, Annualized return, Sharpe ratio ($r_f = 0$, annualized $\sqrt{365}$), Sortino ratio, Maximum Drawdown, Calmar ratio

point, structured prompts were constructed using data available up to day t-1, and the agent produced both a natural-language rationale and a final trading decision. A simplified pseudocode is provided in [Algorithm 1](#) to clarify the timing.

Performance Evaluation. Daily portfolio values were recorded throughout the test period using this unified procedure. These values served as the basis for computing a set of key performance metrics that quantitatively evaluated both the profitability and risk profile of each strategy. The selected metrics captured not only the absolute return but also the risk-adjusted performance and downside exposure, enabling robust and interpretable comparisons across different modeling approaches. The performances of all strategies were assessed using the following indicators:

- Total return: The percentage increase in total portfolio value from the initial capital to the end of the test period.
- Annualized return: The geometric mean return scaled to a 365-day horizon, computed as $(1 + \text{Total Return})^{365/\text{days}} - 1$
- Sharpe ratio: The mean of daily returns divided by their standard deviation, annualized using a 365-day factor, and computed with a risk-free rate of $r_f = 0$ ([Sharpe, 1966](#)).
- Sortino ratio: Similar to the Sharpe ratio but penalizing only downside volatility, providing a more targeted measure of risk-adjusted performance ([Sortino & Price, 1994](#)).
- Maximum Drawdown (MDD): The maximum observed loss from a peak to a trough during the test period, capturing worst-case performance.
- Calmar ratio: The annualized return divided by the MDD, reflecting return per unit of downside risk over the evaluation period ([Young, 1991](#)).

Collectively, these indicators provide a balanced evaluation of profitability, risk-adjusted performance, downside protection, and trading consistency, enabling robust and interpretable comparisons across strategies. In addition to profitability and risk-adjusted measures, three supplementary diagnostics were computed to capture trading intensity and position management.

- Exposure time: Exposure time was defined as the proportion of days in which the portfolio held a non-zero position relative to the total number of days in the evaluation horizon. This measure reflects the degree of continuous market participation.
- Average holding period: The average holding period was computed as the mean duration (in consecutive trading days) of uninterrupted non-zero position intervals. This diagnostic distinguishes between persistent trading stances and frequent, short-lived position changes.
- Turnover: Turnover was calculated as the sum of absolute daily position changes multiplied by the contemporaneous closing price, normalized by the initial capital base. This measure reflects capital reallocation intensity, providing insight into whether excess performance arises from genuine reasoning advantages or from opportunistic trading frequency.

These diagnostics were applied consistently across all strategies, ensuring that differences in profitability could be evaluated considering trading intensity and position stability.

4. Results

4.1. Experimental setup

The experiments were carried out on Google Colaboratory using an NVIDIA Tesla T4 GPU, paired with an Intel(R) Xeon(R) CPU @ 2.00 GHz and 53 GB of system memory.

To ensure both reproducibility and realistic language generation, we adopted a decoding setup that balances stability with natural linguistic diversity. Specifically, we used a non-zero temperature of 0.7, which introduces moderate variability into the outputs. To evaluate robustness, each configuration was repeated across 20 different random seeds, and reported results represent averages over these runs. This design ensures that findings are not artifacts of a particular seed choice while still reflecting the naturalistic variety characteristic of large language models.

All role-specific agents and the meta-agent were executed under the following decoding parameters unless otherwise specified

Table 2
Decoding parameters for all agents.

Agent	Model	Temperature	Top-p	Max tokens	Seed
On-chain	GPT-4o	0.7	1.0	1024	20 runs with varied seed
Technical indicator	GPT-4o	0.7	1.0	1024	20 runs with varied seed
Macro-financial	GPT-4o	0.7	1.0	1024	20 runs with varied seed
News	GPT-4o	0.7	1.0	1024	20 runs with varied seed
Social media	GPT-4o	0.7	1.0	1024	20 runs with varied seed
Trading	GPT-4o	0.7	1.0	1024	20 runs with varied seed

Note. We deliberately avoided fully deterministic decoding (temperature = 0.0). While such settings guarantee identical outputs, they may produce overly rigid and unnatural responses that underestimate the expressive range of LLM reasoning. By choosing temperature = 0.7 and averaging across varied seeds, we ensured (i) robustness of performance across stochastic runs and (ii) naturalistic decision rationales that better reflect real-world usage scenarios.

(Table 2):

4.2. Trading strategy evaluation results

To ensure a fair comparison of profitability and risk-adjusted performance across models, we calibrated key trading parameters (e.g., threshold levels) in the baseline models so that their average trade counts fell within the 70–110 range. This range corresponds to the typical trading frequency observed in the LLM agent during preliminary experiments, thereby minimizing confounding effects from trading intensity.

Baseline experiments followed a comprehensive evaluation design that included four time-series models, three input window sizes (3, 14, and 30 days), a grid search over trading thresholds, and a hyperparameter sweep over hidden dimensions (32, 64, and 128). Each configuration was repeated across 20 random seeds to ensure robustness. All models were trained with the Adam optimizer, up to 100 epochs with early stopping (patience=10) on validation loss. The LLM-based agent was evaluated under the same window settings and repetition counts for consistency. This rigorous setup ensured that observed performance differences reflected the models themselves rather than inconsistencies in experimental design, providing a reliable basis for comparison between prediction-based and reasoning-based strategies. The test evaluation period was common to all window configurations. It spanned from June 6, 2024 to February 28, 2025 and covered a total of 268 days.

4.2.1. Window size 3

Under the three-day window setting, the optimal configuration across the time-series models included a hidden dimension of 32, learning rate of 0.0005 (0.05 for DLinear), batch size of 64, and trading threshold of 0.015. A summary of the performance results for this configuration is presented in Table 3.

The LLM-based multi-agent strategy outperformed all baselines across both profitability and risk-adjusted categories. It achieved the highest total return ($21.75\% \pm 3.03$) and annualized return ($29.30\% \pm 3.30$), while maintaining competitive downside protection with an MDD of $14.69\% \pm 0.87$. In terms of risk-adjusted metrics, it recorded a Sharpe ratio of 1.08 ± 0.11 , a Sortino ratio of 1.71 ± 0.19 , and a Calmar ratio of 1.99 ± 0.25 , reflecting a balanced profile of profitability and risk control.

Among the baselines, LSTM and DLinear delivered the strongest results, with total returns of $20.05\% \pm 3.74$ and $19.79\% \pm 3.69$, respectively. However, their risk-adjusted metrics were consistently lower than those of the GPT-based strategy, with Sharpe ratios of 1.05 ± 0.17 and 1.03 ± 0.15 , and Calmar ratios of 2.07 ± 0.41 and 2.00 ± 0.36 . While these baselines demonstrated solid profitability, they exhibited relatively higher sensitivity to drawdowns and volatility. By contrast, PatchTST underperformed in this short-horizon configuration, with the lowest Total return and weakest Calmar ratio. This suggests that Transformer-based models may be less effective in capturing short-term, fine-grained temporal dependencies.

Overall, these findings indicate that the proposed GPT-based multi-agent framework offers the most favorable risk-return trade-off under short-horizon conditions, successfully balancing profitability with robustness against drawdowns (Fig. 2).

4.2.2. Window size 14

Under the 14-day input window setting, the optimal configuration included a hidden dimension of 64, learning rate of 0.0005 (0.01 for DLinear), batch size of 64, and trading threshold of 0.015. A summary of the performance results for this configuration is presented in Table 4.

The GRU achieved the highest Total return at $21.65\% \pm 4.06$, alongside a solid Calmar ratio of 2.10 ± 0.34 , highlighting its advantage in balancing profitability and downside risk. The GPT-4o multi-agent framework delivered a Total return of $20.34\% \pm 3.24$ and a Sharpe ratio of 1.02 ± 0.15 , confirming its robustness across mid-range horizons despite slightly higher drawdowns (MDD = $15.88\% \pm 1.08$).

PatchTST showed the most pronounced improvement compared to the 3-day setting, reaching a Total return of $20.75\% \pm 5.91$ and the highest Calmar ratio of 2.37 ± 0.49 across all models. This indicates that Transformer-based architectures benefit substantially from longer temporal contexts, enabling them to capture trend dynamics more effectively. By contrast, LSTM and DLinear maintained stable performance, with Total returns of $20.09\% \pm 4.60$ and $20.29\% \pm 3.85$, respectively, and Sharpe ratios near unity, underscoring their adaptability across window lengths.

Overall, these findings suggest that under mid-range prediction conditions, GRU and PatchTST exhibit strong profitability and risk-adjusted outcomes, while the GPT-4o multi-agent strategy remains a competitive and interpretable alternative that integrates

Table 3

Trading performance under three-day input window.

Model	Total return (%)	Annualized return (%)	MDD (%)	Sharpe ratio	Sortino ratio	Calmar ratio
Buy&Hold	18.90	26.70	22.52	0.80	1.32	1.19
GRU	18.05 ± 4.55	24.40 ± 6.20	13.70 ± 1.23	0.97 ± 0.19	1.51 ± 0.30	1.78 ± 0.48
LSTM	20.05 ± 3.74	27.10 ± 4.10	13.10 ± 1.64	1.05 ± 0.17	1.59 ± 0.27	2.07 ± 0.41
DLinear	19.79 ± 3.69	26.80 ± 4.11	13.37 ± 1.29	1.03 ± 0.15	1.55 ± 0.23	2.00 ± 0.36
PatchTST	6.90 ± 4.71	9.30 ± 6.60	14.45 ± 2.03	0.48 ± 0.23	0.75 ± 0.36	0.64 ± 0.47
Ours (GPT-4o)	21.75 ± 3.03	29.30 ± 3.30	14.69 ± 0.87	1.08 ± 0.11	1.71 ± 0.19	1.99 ± 0.25

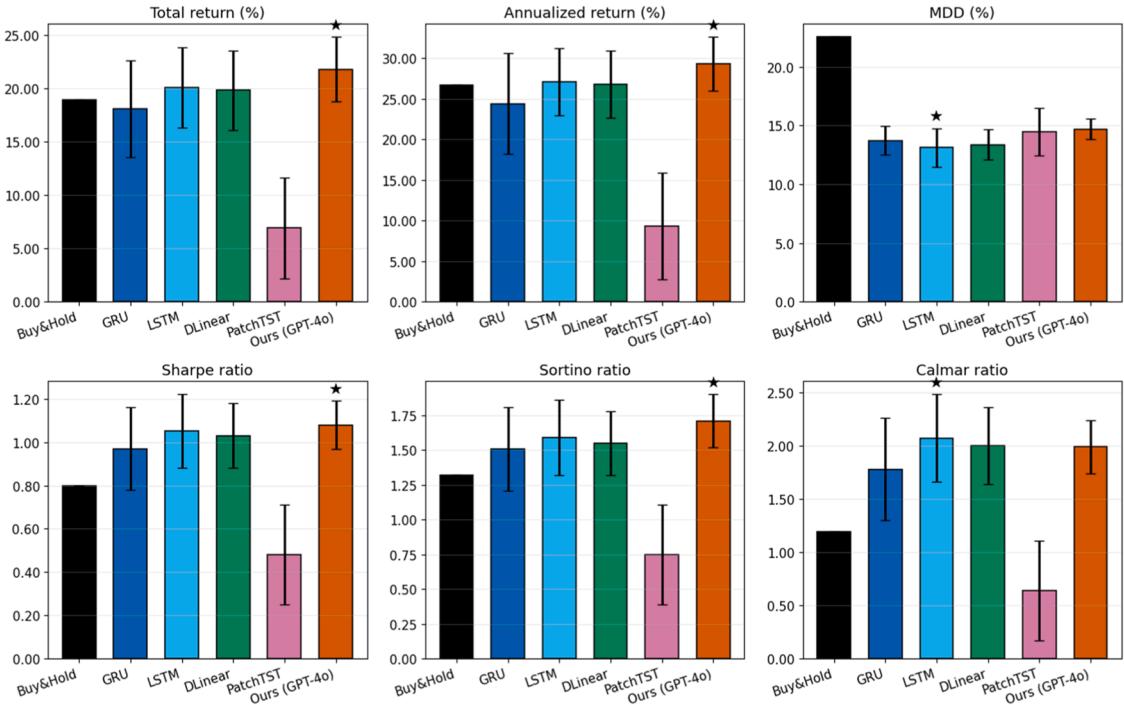


Fig. 2. Performance summary across trading models (3-day window, \$100k capital, fee/slippage 0.1 %).

Table 4

Trading performance under 14-day input window.

Model	Total return (%)	Annualized return (%)	MDD (%)	Sharpe ratio	Sortino ratio	Calmar ratio
Buy&Hold	18.90	26.70	22.52	0.80	1.32	1.19
GRU	21.65 ± 4.06	28.32 ± 5.32	13.46 ± 1.62	1.05 ± 0.20	1.55 ± 0.30	2.10 ± 0.34
LSTM	20.09 ± 4.60	25.96 ± 5.94	13.47 ± 2.51	1.00 ± 0.22	1.50 ± 0.33	1.93 ± 0.44
DLinear	20.29 ± 3.85	26.34 ± 4.99	16.19 ± 2.28	0.95 ± 0.18	1.45 ± 0.28	1.63 ± 0.29
PatchTST	20.75 ± 5.91	27.04 ± 7.71	11.39 ± 1.22	1.15 ± 0.25	1.70 ± 0.37	2.37 ± 0.49
Ours (GPT-4o)	20.34 ± 3.24	26.29 ± 4.19	15.88 ± 1.08	1.02 ± 0.15	1.55 ± 0.23	1.66 ± 0.20

heterogeneous signals through reasoning (Fig. 3).

4.2.3. Window size 30

Under the 30-day input window setting, the optimal configuration employed a hidden dimension of 128, a learning rate of 0.0005 (0.01 for DLinear), a batch size of 64, and a trading threshold of 0.010. This long-term configuration allowed models to aggregate extended temporal information, potentially favoring architectures with stable sequence modeling capabilities. The performance results are summarized in Table 5.

Among all models, GRU delivered the strongest overall performance, recording the highest Total return of $22.14\% \pm 3.81$ and the most favorable risk-adjusted profile, with a Sharpe ratio of 1.55 ± 0.27 and Sortino ratio of 2.56 ± 0.44 . Its Calmar ratio of 2.31 ± 0.40 also indicates efficient downside risk management despite longer horizon.

PatchTST also performed competitively in this setting, achieving a Total return of $20.07\% \pm 4.41$ and a Calmar ratio of 1.85 ± 0.36 , showing that Transformer-based architectures can capitalize on longer input contexts to capture trend dynamics. DLinear and LSTM delivered moderate performance, with Total returns of $18.12\% \pm 6.01$ and $16.03\% \pm 4.65$, respectively, alongside Sharpe ratios around 1.0.

By contrast, the GPT-4o multi-agent framework showed a relative decline in both profitability and stability under long-horizon conditions. It achieved a Total return of $17.40\% \pm 4.82$ with the highest MDD ($20.49\% \pm 1.07$), resulting in the weakest Calmar ratio (1.14 ± 0.33). While short- and mid-horizon settings demonstrated the competitiveness of the framework, its weaker performance under 30-day windows highlights a key limitation of zero-shot reasoning. The agent's reliance on recent textual and structured cues may have led to an overemphasis on short-lived market dynamics, reducing its ability to abstract persistent trends. Unlike recurrent or Transformer-based architectures that explicitly encode temporal dependencies, the GPT-based agent lacks mechanisms for memory consolidation or temporal smoothing, which likely contributed to its higher drawdown and weaker risk-adjusted profile.

Overall, the results highlight that RNN-based models, particularly GRU, retain a significant advantage in long-horizon contexts,

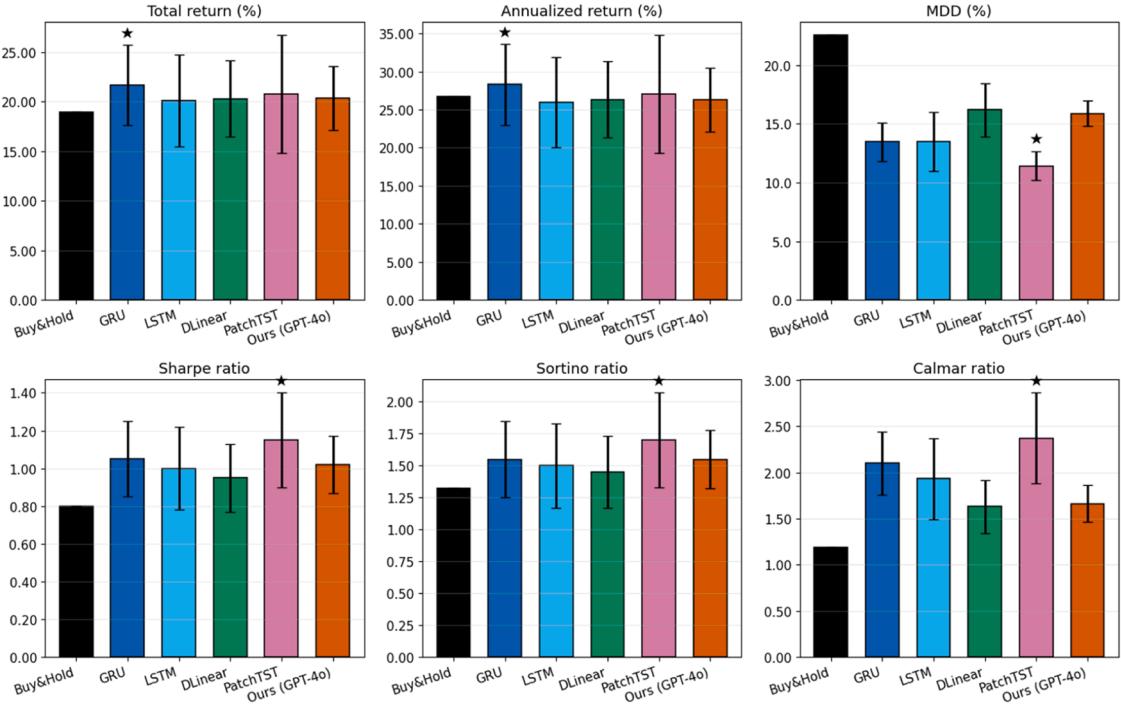


Fig. 3. Performance summary across trading models (14-day window, \$100k capital, fee/slippage 0.1 %).

Table 5

Trading performance under 30-day input window.

Model	Total return (%)	Annualized return (%)	MDD (%)	Sharpe ratio	Sortino ratio	Calmar ratio
Buy&Hold	18.90	26.70	22.52	0.80	1.32	1.19
GRU	22.14 ± 3.81	31.74 ± 5.19	13.73 ± 1.87	1.55 ± 0.27	2.56 ± 0.44	2.31 ± 0.40
LSTM	16.03 ± 4.65	21.42 ± 6.20	14.41 ± 1.36	1.00 ± 0.32	1.65 ± 0.52	1.49 ± 0.47
DLinear	18.12 ± 6.01	24.48 ± 8.11	15.07 ± 2.38	1.10 ± 0.28	1.81 ± 0.47	1.63 ± 0.42
PatchTST	20.07 ± 4.41	27.13 ± 5.99	14.68 ± 1.63	1.24 ± 0.24	2.05 ± 0.40	1.85 ± 0.36
Ours (GPT-4o)	17.40 ± 4.82	23.39 ± 6.52	20.49 ± 1.07	0.77 ± 0.22	1.26 ± 0.37	1.14 ± 0.33

while Transformer-based models like PatchTST show improved adaptability. The GPT-based agent, although competitive in short- to mid-range settings, appears less effective at longer horizons (Fig. 4). These results align with the temporal rationale outlined in Section 3.5, indicating that short- and medium-horizon settings favor reasoning-based adaptability, whereas extended horizons increasingly reward architectures with explicit temporal encoding mechanisms.

4.3. Robustness diagnostics

To further validate the reliability of our findings, we conducted a series of robustness diagnostics. Since performing these checks across all window sizes was impractical, we focus on the three-day input window, which provides the most representative case given the agent's strong performance and clear contrasts with baselines.

4.3.1. Diagnostics of trading intensity and position management

While profitability and risk-adjusted metrics provide a high-level view of model performance, it is also important to assess whether the reported gains are driven by excessive trading or unstable position management. To address this concern, we report additional diagnostics, including the realized trade count, turnover, exposure time, and average holding period (Table 6). These diagnostics provide insights into the trading style of each model and clarify whether observed advantages caused from robust reasoning rather than opportunistic frequency effects.

As shown, all baseline models were calibrated to fall within the 70–110 trade range, ensuring comparability. The GPT-4o multi-agent executed approximately 84 trades on average, closely aligned with RNN and linear baselines. Its turnover (7.60 ± 1.12) was comparable to GRU and LSTM, indicating that outperformance was not driven by disproportionately higher trading activity. The exposure time (93.15 %) suggests that the agent maintained a market position for the majority of the test period, while the average holding period (22 days) was slightly longer than that of the baselines, reflecting a tendency toward more persistent positions rather

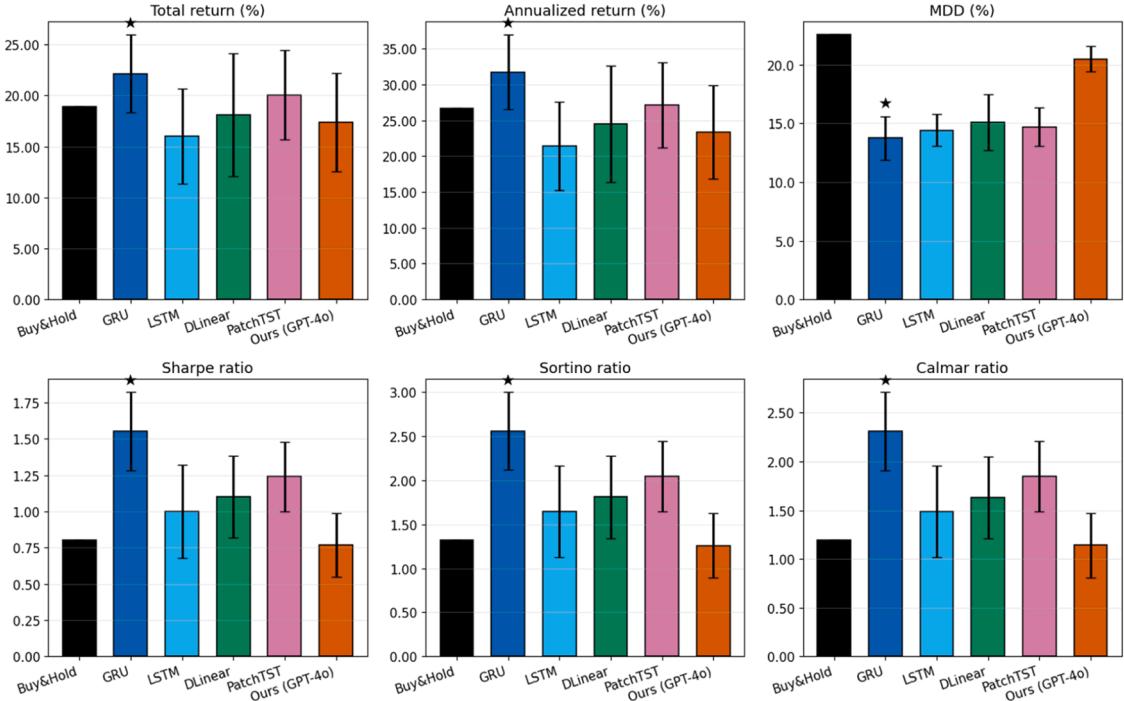


Fig. 4. Performance summary (30-day window, \$100k capital, fee/slippage 0.1 %).

Table 6

Trading behavior diagnostics under the three-day input window.

Model	Trade count	Turnover	Exposure time (%)	Avg holding period (days)
Buy & Hold	1	0.00	100.0	267.0
GRU	74.35 ± 21.19	6.90 ± 1.05	92.51 ± 2.41	18.35 ± 3.12
LSTM	90.12 ± 18.47	6.75 ± 0.98	91.74 ± 2.77	19.02 ± 2.91
DLinear	87.28 ± 17.05	6.85 ± 1.10	90.93 ± 2.85	17.84 ± 3.05
PatchTST	72.44 ± 15.36	5.20 ± 0.82	88.18 ± 3.20	20.21 ± 2.76
Ours (GPT-4o)	83.67 ± 11.22	7.60 ± 1.12	93.15 ± 2.51	22.00 ± 2.63

Table 7

Sensitivity of performance metrics to fee/slippage assumptions.

Model	Metric	Fees and slippage		
		0.05 %	0.10 %	0.15 %
Buy&Hold	Total return (%)	19.12	18.93	18.74
	Sharpe ratio	0.81	0.80	0.79
	Calmar ratio	1.20	1.19	1.18
GRU	Total return (%)	18.32	18.10	17.87
	Sharpe ratio	0.99	0.97	0.96
	Calmar ratio	1.82	1.78	1.75
LSTM	Total return (%)	20.27	20.08	19.86
	Sharpe ratio	1.07	1.05	1.04
	Calmar ratio	2.10	2.07	2.04
DLinear	Total return (%)	20.02	19.83	19.64
	Sharpe ratio	1.05	1.03	1.02
	Calmar ratio	2.04	2.00	1.97
PatchTST	Total return (%)	7.08	6.92	6.71
	Sharpe ratio	0.50	0.48	0.47
	Calmar ratio	0.65	0.64	0.63
Ours (GPT-4o)	Total return (%)	22.03	21.82	21.61
	Sharpe ratio	1.10	1.08	1.06
	Calmar ratio	2.05	1.99	1.97

than excessive turnover of positions.

These diagnostics demonstrate that the GPT-4o strategy achieved higher returns while operating under similar trading intensity as conventional baselines, reinforcing the robustness of its interpretability-driven decision process.

4.3.2. Sensitivity of performance metrics to fee/slippage assumptions

Transaction fees and slippage can substantially affect the net profitability of high-frequency strategies. In our main experiments, each trade incurred a fee of 0.1 % and slippage of 0.1 % on each side of the trade, which reflects the average maker–taker fee structure observed on major cryptocurrency exchanges such as Binance and Coinbase. To evaluate robustness, we performed a deterministic sensitivity analysis under alternative assumptions of ± 0.05 % around the baseline values. Table 7 reports results for three representative indicators, which jointly capture profitability, risk-adjusted performance, and downside risk. Other metrics exhibited qualitatively similar patterns and are omitted.

Across all specifications, the relative ordering of performance remained stable, and the GPT-4o multi-agent strategy consistently outperformed baselines in both profitability and risk-adjusted terms. The modest declines observed under higher fee/slippage assumptions confirm that the reported advantages are not artifacts of cost calibration but reflect genuine robustness.

4.3.3. Temporal dynamics analysis

While summary metrics capture overall profitability and risk-adjusted outcomes, they do not fully reflect the time-varying dynamics of portfolio evolution. To complement the quantitative results, Figs. 5 and 6 present the cumulative return trajectories and drawdown profiles of the proposed GPT-based multi-agent strategy compared with a Buy-and-Hold benchmark under the three-day input window, where the agent achieved its strongest performance.

Fig. 5 shows that the cumulative return of the proposed strategy tracked closely with Buy-and-Hold during test period while demonstrating more stable behavior during periods of volatility. Although headline returns appeared similar, Our framework achieved these outcomes with a smoother trajectory, highlighting its ability to mitigate drawdowns during volatile periods.

Fig. 6 illustrates the drawdown comparison between the two strategies. Our approach delivered equivalent or stronger profitability while materially constraining downside exposure, with a maximum drawdown of about -15 % versus -20 % for Buy-and-Hold. This resilience translated into markedly stronger Calmar and Sortino ratios.

These diagnostics confirm that the proposed strategy converted nominally similar headline returns into demonstrably more robust, risk-adjusted performance by limiting deep losses while capturing meaningful upward trends. These temporal dynamics underscore the interpretability-driven advantages of the framework and its practical value for risk-sensitive trading contexts.

4.4. Extended evaluation with text-based agents

4.4.1. Quantitative performance evaluation through agent ablation

To evaluate the contribution of unstructured text-based information to the trading performance, a series of ablation experiments were conducted by incrementally integrating two text-based agents. All experiments were performed in the three-day window setting, which yielded the strongest baseline results.

To comply with the input limitations of GPT-4o while preserving temporal diversity and representativeness, Reddit and news content were capped at 90,000 words per three-day decision window, with a daily limit of 30,000 words per source. When daily volumes exceeded this threshold, documents were stratified-sampled in proportion to their counts, ensuring fair coverage across time

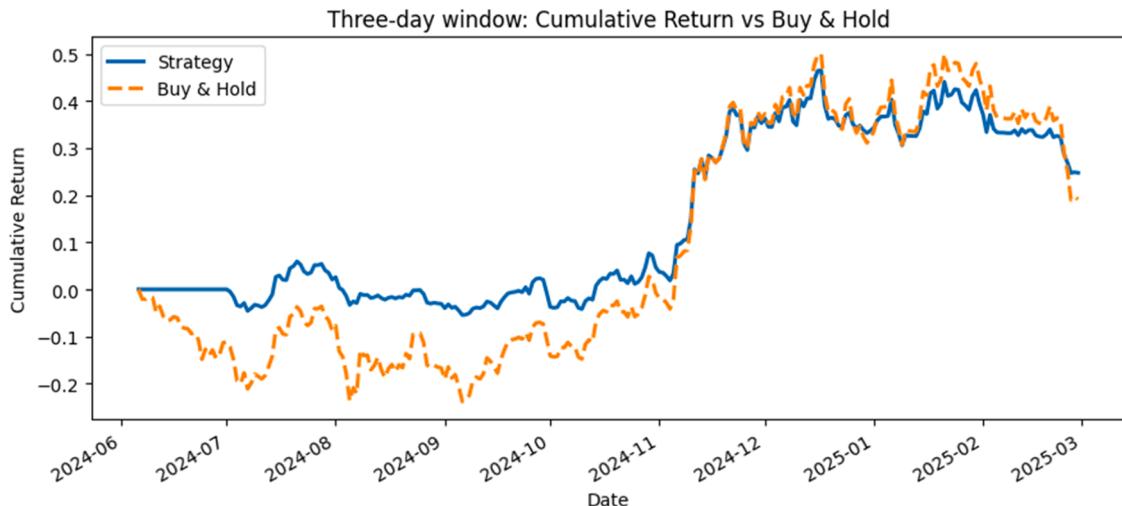


Fig. 5. Cumulative return trajectories of the GPT-based multi-agent strategy versus the Buy-and-Hold benchmark under the three-day input window (fee = 0.1 %, slippage = 0.1 %).

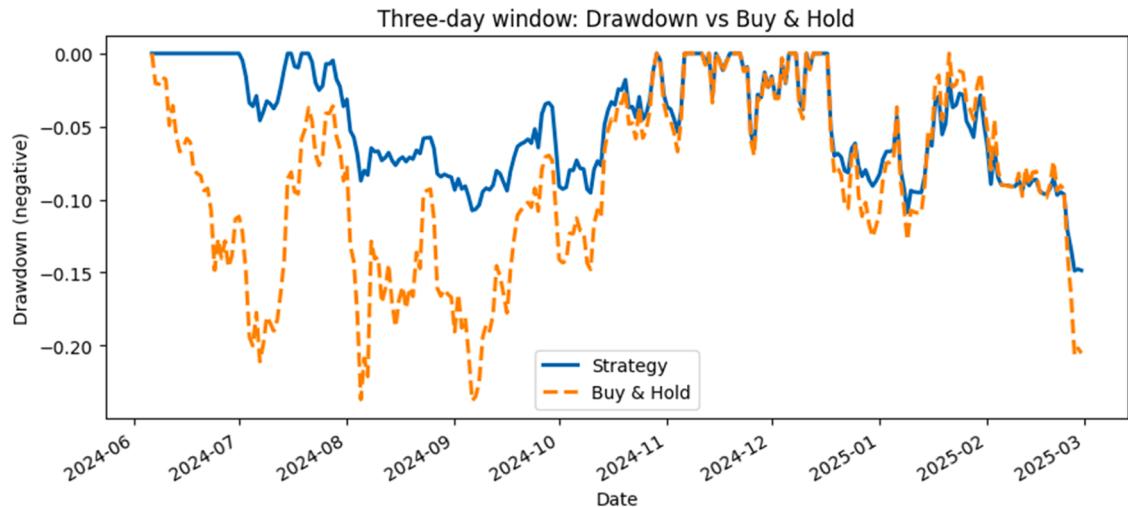


Fig. 6. Drawdown profiles of the GPT-based multi-agent strategy and Buy-and-Hold under the three-day input window (fee = 0.1 %, slippage = 0.1 %).

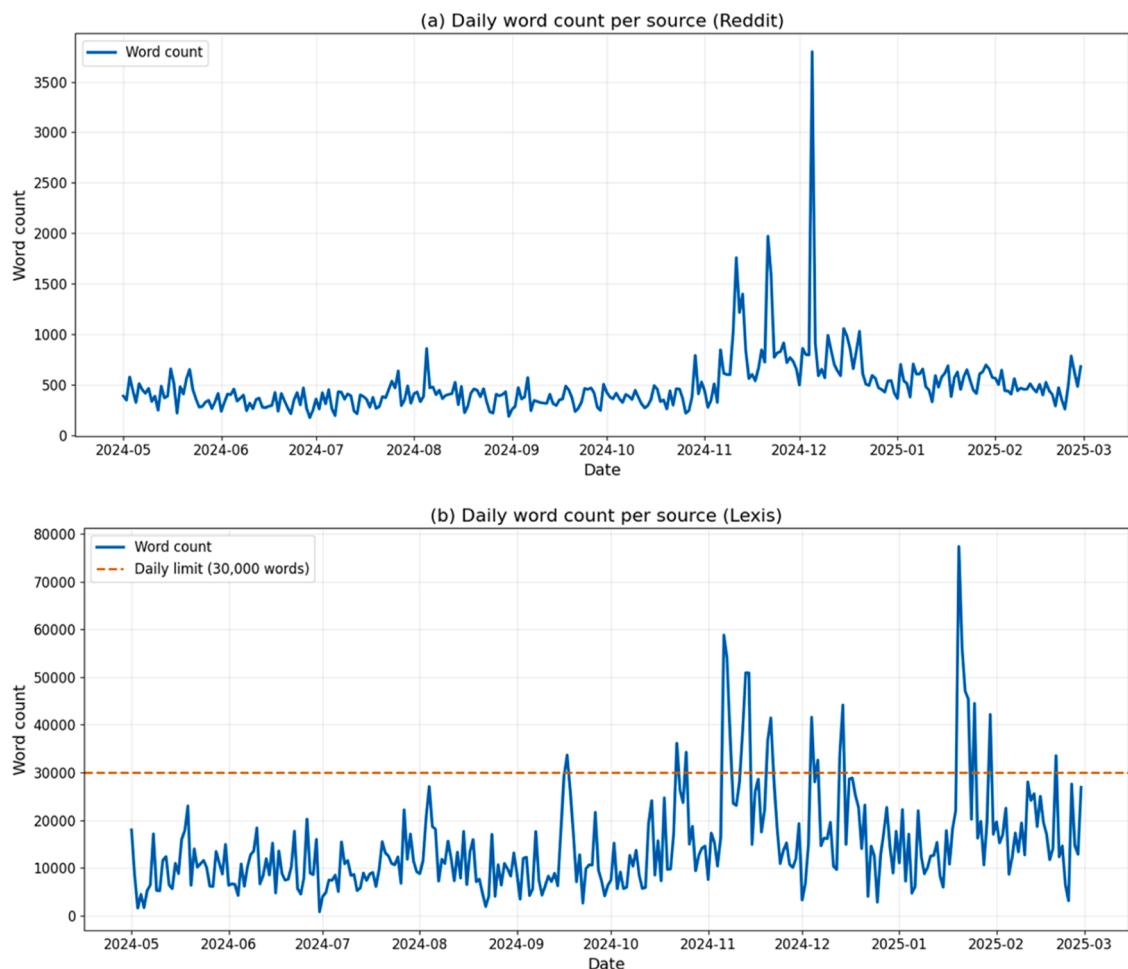


Fig. 7. Daily word counts distribution of textual datasets. (a) Reddit posts. (b) News articles from LexisNexis. The red dashed line indicates the 30,000-word daily limit.

and sources without surpassing the token limit. To validate this procedure, we examined realized daily word counts relative to the cap. As shown in Fig. 7, only a few days exceeded 30,000 words, while most remained well below, confirming that the sampling preserved content coverage, temporal balance, and reproducibility.

To evaluate the contribution of each text-based agent, four configurations were tested under a three-day input window setting: (1) a structured-only baseline; (2) structured inputs with the factual news agent; (3) structured inputs with the social media sentiment agent; and (4) structured inputs with both agents combined. The comparative performance results are summarized in Table 8.

As shown in Table 8, the baseline model, which utilized only structured inputs achieved a Total return of $21.75\% \pm 3.03$, with a Sharpe ratio of 1.08 ± 0.11 , a Calmar ratio of 1.99 ± 0.25 , and an MDD of $14.69\% \pm 0.87$.

When only the news agent was added, the performance declined substantially. The Total return dropped to $11.39\% \pm 1.06$, while the Sharpe and Calmar ratios fell to 0.55 ± 0.08 and 1.04 ± 0.12 , respectively. MDD increased slightly to $15.37\% \pm 0.50$, suggesting that news-based sentiment may have diluted or interfered with the structured signals, introducing interpretive noise.

By contrast, the inclusion of the social media agent alone improved performance beyond the baseline. This configuration achieved a Total return of $23.30\% \pm 2.41$, a Sharpe ratio of 1.12 ± 0.12 , and a Calmar ratio of 2.01 ± 0.28 , with a modest increase in drawdown to $16.36\% \pm 1.05$. These results suggest that Reddit's community sentiment is more tightly associated with short-horizon price movements and retail investor behavior, thereby offering practical predictive value in high-frequency trading environments.

However, combining both social media and news agents resulted in lower performance than using social media alone. This dual-agent setup yielded a Total return of $18.98\% \pm 5.74$, a Sharpe ratio of 0.95 ± 0.15 , and a Calmar ratio of 1.52 ± 0.22 , with a drawdown of $17.67\% \pm 0.51$. This degradation suggests that news content may offset the utility of social media sentiment signals and highlights the importance of selective integration and source-specific calibration when incorporating multiple unstructured inputs.

4.4.2. Diagnostic assessment based on conflict and actionability analysis

To better understand the performance degradation observed in the dual-agent configuration relative to the Reddit-only setup, a diagnostic evaluation was conducted using a language-based interpretability protocol called G-EVAL (Liu et al., 2023). This procedure leverages GPT-4o in a zero-shot prompting framework to evaluate the semantic relationship between Reddit-derived sentiment and fact-based news summaries, which are the two unstructured inputs provided to the trading meta-agent.

Each evaluation instance consisted of a pair of daily summaries, covering the same time window. For each pair, GPT-4o was prompted to output two scores along with natural language justifications:

- Conflict score (0–1): Quantifies the degree of semantic divergence between two sources in terms of tone, outlook, and implication. Higher values indicate a more pronounced disagreement, such as one source signaling bullish momentum while the other warns of systemic risk.
- Actionability score (0–1): Assesses the extent to which the combined sentiment offers clear and temporally relevant signals for decision-making. Higher values reflect stronger utility of executing timely trades.

To maintain consistency and interpretability, structured prompts were applied to each Reddit–news pair in the test dataset. Table 9 presents detailed statistics for both evaluation metrics. Representative prompt templates and sample outputs of the G-EVAL procedure are provided in Appendix C to illustrate the evaluation process.

The average conflict score was 0.42, with the top quartile exceeding 0.60, suggesting frequent semantic clashes between Reddit and the news content. The actionability score averaged at 0.55, indicating that even when the sentiment was not in strong conflict, the combined input often failed to yield actionable trading insights. Fig. 8 presents the empirical distribution of evaluation scores.

Overall, these results reveal two key diagnostic insights. First, significant differences in meaning between sources can create confusion, making it more difficult for the model to build a clear line of reasoning from mixed signals. Second, even when disagreement is low, the relatively low average actionability indicates that simply combining sentiment sources without careful selection often fails to provide clear or useful signals for timely trading decisions.

5. Discussion

5.1. Summary of key findings

This study proposes a zero-shot LLM-based multi-agent trading framework that integrates heterogeneous data sources, including technical indicators, on-chain metrics, macroeconomic variables, and textual signals processed by role-specific agents. The empirical results indicate that the GPT-4o-based meta-agent outperformed conventional time-series baselines, particularly in short-horizon forecasting scenarios with a three-day input window. Notably, the inclusion of Reddit sentiment improves trading performance, whereas the incorporation of news has a destabilizing effect, suggesting that the degree of signal congruence plays a crucial role in decision quality. These findings were further substantiated by G-EVAL diagnostics, which demonstrated that decisions derived from conflicting inputs were less coherent and actionable, underscoring the importance of epistemic alignment in multi-source reasoning.

5.2. Theoretical implications

This study contributes to the theoretical understanding of modular reasoning in financial AI. Traditional predictive systems often rely on monolithic architectures that encode statistical correlations between the input features and target variables. By contrast, our

Table 8

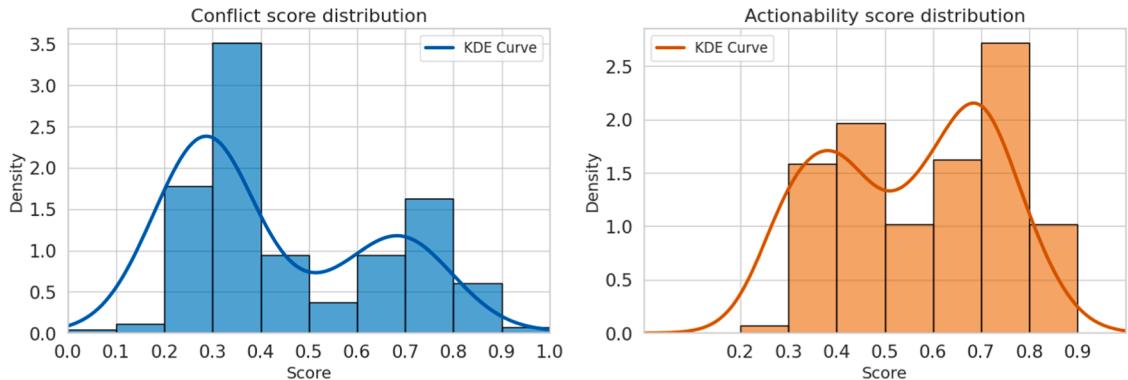
Performance comparison of GPT-4o-based trading strategies under different text agent configurations.

Model configuration	Total return (%)	Annualized return (%)	MDD (%)	Sharpe ratio	Sortino Ratio	Calmar ratio
Baseline (Structured only)	21.75 ± 3.03	29.30 ± 3.30	14.69 ± 0.87	1.08 ± 0.11	1.71 ± 0.19	1.99 ± 0.25
Baseline + News agent	11.39 ± 1.06	15.90 ± 1.50	15.37 ± 0.50	0.55 ± 0.08	0.88 ± 0.12	1.04 ± 0.12
Baseline + Social media agent	23.30 ± 2.41	32.90 ± 3.20	16.36 ± 1.05	1.12 ± 0.12	1.76 ± 0.21	2.01 ± 0.28
Baseline + News + Social media agents	18.98 ± 5.74	26.80 ± 7.20	17.67 ± 0.51	0.95 ± 0.15	1.46 ± 0.25	1.52 ± 0.22

Table 9

Descriptive statistics of conflict and actionability scores.

Metric	Mean	Median	Standard deviation	25 %	75 %	Max
Conflict score	0.42	0.30	0.21	0.30	0.60	1.00
Actionability score	0.55	0.60	0.17	0.40	0.70	0.90

**Fig. 8.** Distributions of conflict and actionability scores.

framework operationalizes distributed cognitive architecture, where each agent performs contextualized reasoning over a distinct modality and produces interpretable textual rationales. The GPT-based meta-agent synthesizes these justifications into a final decision that approximates human ensemble judgment. This design points toward a potential shift from function approximation to transparent reasoning, aligning with the growing demand for XAI (Doshi-Velez & Kim, 2017; Guidotti et al., 2018).

Furthermore, this study advances methodological approaches for sentiment integration. Rather than compressing textual evidence into scalar sentiment scores, the framework leverages generative LLMs to process full-length documents, enabling the detection of latent conflicts between sentiment and factual narratives that are often overlooked in multimodal fusion. By quantifying conflict and actionability metrics with GPT-4o, the study provides a replicable protocol for evaluating epistemic alignment across information sources in complex decision pipelines.

5.3. Practical implications for trading system design

From a practical perspective, the framework demonstrates that zero-shot prompting can deliver competitive trading performance without task-specific fine-tuning, thereby reducing system maintenance costs by avoiding repeated retraining cycles. Its transparent, text-based outputs facilitate human interpretability and regulatory auditing, both of which are increasingly crucial under evolving financial AI regulations..

The modular agent design further enhances system extensibility; new data streams or reasoning logic can be selectively integrated by adding specialized agents without retraining the entire system. This feature allows the system to evolve with minimal disruption, which is an important consideration in volatile and rapidly evolving markets such as cryptocurrencies.

However, our results also revealed the risks of semantic redundancy and noise amplification when aggregating multiple textual signals. Community-driven sentiment aligned more closely with market behavior than formal news, which occasionally introduced misaligned signals. This highlights the need for selective signal gating and agent-level confidence weighting mechanisms in real-world deployment.

5.4. Limitations and future work

Despite these strengths, the proposed framework has several limitations.

First, performance decreased under longer temporal windows, suggesting that zero-shot LLMs may lack adequate temporal

abstraction for mid- to long-term forecasting. Unlike RNNs or Transformer-based models that explicitly encode temporal dependencies, the zero-shot framework struggled to abstract persistent multi-week trends, leading to higher drawdowns and weaker risk-adjusted outcomes. Future research could address this limitation through memory-augmented prompting or time-aware reasoning strategies.

Second, the evaluation was limited to a single-asset environment. Applying the framework to multi-asset or cross-market settings requires addressing inter-asset dependencies, diversification, and portfolio-rebalancing mechanisms, each of which poses unique challenges for modular LLM-based architectures. We deliberately focused on Bitcoin because it remains the most liquid and representative cryptocurrency, with the longest and most consistent historical record of price, on-chain, and sentiment data. By contrast, other altcoins suffer from shorter data availability, structural breaks, and less reliable coverage in news and community channels, which complicates direct comparability. While this choice constrains generalizability, it ensures methodological rigor and stable evaluation. Future work could extend the framework to multi-asset portfolios once sufficiently robust and homogeneous datasets are available.

Third, although zero-shot prompting enhances generalizability, it remains sensitive to prompt phrasing and may degrade during distributional shifts. While the constrained, template-based setup mitigated hallucination risk, it may also restrict reasoning flexibility, limiting the system's adaptability to unseen contexts. Future work could explore few-shot tuning, adaptive prompt refinement, or self-verification mechanisms to enhance the robustness and generalization. Lastly, this study demonstrates that conflict and actionability scores explain the observed performance degradation when combining news with Reddit, we did not investigate their direct correlations with subsequent realized returns or error rates. Such an analysis would offer an additional validity check on the scores themselves and constitutes a valuable direction for future research. Taken together, these limitations outline promising directions for future research, from temporal abstraction and multi-asset expansion to prompt optimization, which can strengthen the applicability of LLM-based trading systems in real-world contexts.

CRediT authorship contribution statement

Hae Sun Jung: Writing – review & editing, Writing – original draft, Visualization, Validation, Resources, Project administration, Methodology, Investigation, Formal analysis, Data curation, Conceptualization. **Haein Lee:** Writing – review & editing, Writing – original draft, Visualization, Validation, Project administration, Methodology, Investigation, Formal analysis, Data curation, Conceptualization.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

The authors appreciate Editage (www.editage.co.kr) for their English editing service. This research received no external funding.

Supplementary materials

Supplementary material associated with this article can be found, in the online version, at [doi:10.1016/j.ipm.2025.104466](https://doi.org/10.1016/j.ipm.2025.104466).

Data availability

Data will be made available on request.

References

- Abu Bakar, N., & Rosbi, S. (2017). Autoregressive integrated moving average (ARIMA) model for forecasting cryptocurrency exchange rate in high volatility environment: A new insight of bitcoin transaction. *International Journal of Advanced Engineering Research and Science*, 4(11), 130–137. <https://doi.org/10.22161/ijars.4.11.20>
- Akyildirim, E., Goncu, A., & Sensoy, A. (2021). Prediction of cryptocurrency returns using machine learning. *Annals of Operations Research*, 297(1), 3–36. <https://doi.org/10.1007/s10479-020-03575-y>
- Almansour, B. Y., Alshater, M. M., & Almansour, A. Y. (2021). Performance of ARCH and GARCH models in forecasting cryptocurrency market volatility. *Industrial Engineering & Management Systems*, 20(2), 130–139. <https://doi.org/10.7232/iems.2021.20.2.130>
- Brauneis, A., & Sahiner, M. (2024). Crypto volatility forecasting: Mounting a HAR, sentiment, and machine learning horserace. *Asia-Pacific Financial Markets*, 1–33. <https://doi.org/10.1007/s10690-024-09510-6>
- Chatterjee, I., Chakraborti, S., & Tono, T. (2024). A comparative study of bitcoin price prediction during pre-Covid19 and whilst-Covid19 period using time series and machine learning models. *Discover Analytics*, 2(1), 19. <https://doi.org/10.1007/s44257-024-00024-z>
- Cheikh, N. B., Zaiied, Y. B., & Chevallier, J. (2020). Asymmetric volatility in cryptocurrency markets: New evidence from smooth transition GARCH models. *Finance Research Letters*, 35, Article 101293. <https://doi.org/10.1016/j.frl.2019.09.008>

- Cho, K., Van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., & Bengio, Y. (2014). Learning phrase representations using RNN encoder-decoder for statistical machine translation. *arXiv preprint arXiv:1406.1078*. <https://doi.org/10.48550/arXiv.1406.1078>.
- Chowdhury, R., Rahman, M. A., Rahman, M. S., & Mahdy, M. R. C. (2020). An approach to predict and forecast the price of constituents and index of cryptocurrency using machine learning. *Physica A: Statistical Mechanics and its Applications*, 551, Article 124569. <https://doi.org/10.1016/j.physa.2020.124569>
- Delfabbro, P., King, D. L., & Williams, J. (2021). The psychology of cryptocurrency trading: Risk and protective factors. *Journal of behavioral addictions*, 10(2), 201–207. <https://doi.org/10.1556/2006.2021.00037>
- Doshi-Velez, F., & Kim, B. (2017). Towards a rigorous science of interpretable machine learning. *arXiv preprint arXiv:1702.08608*. <https://doi.org/10.48550/arXiv.1702.08608>
- Gadi, M. F. A., & Sicilia, M.Á. (2024). A sentiment corpus for the cryptocurrency financial domain: The CryptoLin corpus. *Language Resources and Evaluation*, 1–19. <https://doi.org/10.1007/s10579-024-09743-x>
- Guidotti, R., Monreale, A., Ruggieri, S., Turini, F., Giannotti, F., & Pedreschi, D. (2018). A survey of methods for explaining black box models. *ACM Computing Surveys (CSUR)*, 51(5), 1–42. <https://doi.org/10.1145/3236009>
- Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural Computation*, 9(8), 1735–1780. <https://doi.org/10.1162/neco.1997.9.8.1735>
- Jain, S., Johari, S., & Delhibabu, R. (2024). The future of cryptocurrency market analysis: Social Media data and user meta-data. *Lobachevskii Journal of Mathematics*, 45(3), 1160–1174. <https://doi.org/10.1134/S1995080224600717>
- Jung, H. S., Kim, J. H., & Lee, H. (2024). Decoding Bitcoin: Leveraging macro-and micro-factors in time series analysis for price prediction. *PeerJ Computer Science*, 10, e2314. <https://doi.org/10.7717/peerj.cs.2314>
- Jung, H. S., Lee, S. H., Lee, H., & Kim, J. H. (2023a). Predicting bitcoin trends through machine learning using sentiment analysis with technical indicators. *Computer Systems Science & Engineering*, 46(2). <https://doi.org/10.32604/csse.2023.034466>
- Jung, H. S., Lee, H., & Kim, J. H. (2023b). Unveiling cryptocurrency conversations: Insights from data mining and unsupervised learning across multiple platforms. *IEEE Access : Practical Innovations, Open Solutions*, 11, 130573–130583. <https://doi.org/10.1109/ACCESS.2023.3334617>
- Kulbhaskar, A. K., & Subramaniam, S. (2023). Breaking news headlines: Impact on trading activity in the cryptocurrency market. *Economic Modelling*, 126, Article 106397. <https://doi.org/10.1016/j.econmod.2023.106397>
- Kumar, A., & Ji, T. (2024). CryptoPulse: Short-term cryptocurrency forecasting with dual-prediction and cross-correlated market indicators. In *Proceedings of the 2024 IEEE International Conference on Big Data (BigData)* (pp. 1–8). IEEE. <https://doi.org/10.1109/BigData62323.2024.1098209>
- Lahmiri, S., & Bekiros, S. (2019). Cryptocurrency forecasting with deep learning chaotic neural networks. *Chaos, Solitons & Fractals*, 118, 35–40. <https://doi.org/10.1016/j.chaos.2018.11.014>
- Lamon, C., Nielsen, E., & Redondo, E. (2017). Cryptocurrency price prediction using news and social media sentiment. *SMU Data Sci. Rev.*, 1(3), 1–22.
- Le, V.D. (2024). Auto-generating earnings report analysis via a financial-augmented LLM. *arXiv preprint arXiv:2412.08179*. <https://doi.org/10.48550/arXiv.2412.08179>
- Lee, H., Kim, J. H., & Jung, H. S. (2025a). ESG-KIBERT: A new paradigm in ESG evaluation using NLP and industry-specific customization. *Decision Support Systems*, 193, Article 114440. <https://doi.org/10.1016/j.dss.2025.114440>
- Lee, H., Kim, J. H., & Jung, H. S. (2025b). From corporate earnings calls to social impact: Exploring ESG signals in S&P 500 ESG index companies through transformer-based models. *Journal of Cleaner Production*, 501, Article 145320. <https://doi.org/10.1016/j.jclepro.2025.145320>
- Liu, Y., Iter, D., Xu, Y., Wang, S., Xu, R., & Zhu, C. (2023). G-eval: NLG evaluation using gpt-4 with better human alignment. *arXiv preprint arXiv:2303.16634*. <https://doi.org/10.48550/arXiv.2303.16634>
- Luo, Y., Feng, Y., Xu, J., Tasca, P., & Liu, Y. (2025). LLM-powered Multi-Agent system for automated crypto portfolio management. *arXiv preprint arXiv:2501.00826*. <https://doi.org/10.48550/arXiv.2501.00826>
- Luukkonen, R., Komulainen, V., Luoma, J., Eskelinen, A., Kanerva, J., Kupari, H.M., Pyysalo, S. (2023). FinGPT: Large generative models for a small language. *arXiv preprint arXiv:2311.05640*. <https://doi.org/10.48550/arXiv.2311.05640>
- Makri, E., Palaiokrassas, G., Bouraga, S., Polychroniadou, A., & Tassiulas, L. (2025). Ethereum price prediction employing large language models for short-term and few-shot forecasting. *arXiv preprint arXiv:2503.23190*. <https://doi.org/10.48550/arXiv.2503.23190>
- Nie, Y., Nguyen, N.H., Sindhong, P., & Kalagnanam, J. (2022). A time series is worth 64 words: Long-term forecasting with transformers. *arXiv preprint arXiv:2211.14730*. <https://doi.org/10.48550/arXiv.2211.14730>
- Park, J. S., O'Brien, J., Cai, C. J., Morris, M. R., Liang, P., & 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* (pp. 1–22). ACM. <https://doi.org/10.1145/3586183.360676>
- Patel, M. M., Tanwar, S., Gupta, R., & Kumar, N. (2020). A deep learning-based cryptocurrency price prediction scheme for financial institutions. *Journal of Information Security and Applications*, 55, Article 102583.
- Peng, Y., Albuquerque, P. H. M., Kimura, H., & Saavedra, C. A. P. B. (2021). Feature selection and deep neural networks for stock price direction forecasting using technical analysis indicators. *Machine Learning with Applications*, 5, Article 100060. <https://doi.org/10.1016/j.mlwa.2021.100060>
- Sharpe, W. F. (1966). Mutual fund performance. *The Journal of business*, 39(1), 119–138.
- Singh, S., & Bhat, M. (2024). Transformer-based approach for ethereum price prediction using crosscurrency correlation and sentiment analysis. *arXiv preprint arXiv:2401.08077*. <https://doi.org/10.48550/arXiv.2401.08077>
- Sortino, F. A., & Price, L. N. (1994). Performance measurement in a downside risk framework. *The Journal of Investing*, 3(3), 59–64. <https://doi.org/10.3905/joi.3.3.59>
- Tanwar, S., Patel, N. P., Patel, S. N., Patel, J. R., Sharma, G., & Davidson, I. E. (2021). Deep learning-based cryptocurrency price prediction scheme with inter-dependent relations. *IEEE Access : Practical Innovations, Open Solutions*, 9, 138633–138646. <https://doi.org/10.1109/ACCESS.2021.3117848>
- Wang, Q., Gao, Y., Tang, Z., Luo, B., Chen, N., & He, B. (2024). Exploring LLM cryptocurrency trading through fact-subjectivity aware reasoning. *arXiv preprint arXiv:2410.12464*. <https://doi.org/10.48550/arXiv.2410.12464>
- Wei, J., Wang, X., Schuurmans, D., Bosma, M., Xia, F., Chi, E., & Zhou, D. (2022). Chain-of-thought prompting elicits reasoning in large language models. *Advances in Neural Information Processing Systems*, 35, 24824–24837. <https://doi.org/10.48550/arXiv.2201.11903>
- Woebbecking, F. (2021). Cryptocurrency volatility markets. *Digital Finance*, 3(3), 273–298. <https://doi.org/10.1007/s42521-021-00037-3>
- Wolk, K. (2020). Advanced social media sentiment analysis for short-term cryptocurrency price prediction. *Expert Systems*, 37(2), Article e12493. <https://doi.org/10.1111/esy.12493>
- Wu, S., Irsoy, O., Lu, S., Dabrowski, V., Dredze, M., Gehrmann, S., & Mann, G. (2023). Bloomberggpt: A large language model for finance. *arXiv preprint arXiv:2303.17564*. <https://doi.org/10.48550/arXiv.2303.17564>
- Young, T. W. (1991). Calmar ratio: A smoother tool. *Futures*, 20(1), 40.
- Zeng, A., Chen, M., Zhang, L., & Xu, Q. (2023). Are transformers effective for time series forecasting? *Proceedings of the AAAI Conference on Artificial Intelligence*, 37(9), 11121–11128. <https://doi.org/10.1609/aaai.v37i9.26317>
- Zhang, Z., Li, S., Zhang, Z., Liu, X., Jiang, H., Tang, X., & Jiang, M. (2025). IHEval: Evaluating language models on following the instruction hierarchy. *arXiv preprint arXiv:2502.08745*. <https://doi.org/10.48550/arXiv.2502.08745>
- Zou, Y., & Herremans, D. (2023). PreBit—A multimodal model with Twitter FinBERT embeddings for extreme price movement prediction of Bitcoin. *Expert Systems with Applications*, 233, Article 120838. <https://doi.org/10.1016/j.eswa.2023.120838>