FROM KNOWING TO DOING: A MEMORY-CONTROLLED BENCHMARK FOR LLM TRADING AGENTS ON STOCK MARKETS

Anonymous authors
Paper under double-blind review

ABSTRACT

Evaluating whether large language model (LLM) agents can profit in capital markets is increasingly framed as end-to-end trading: place an agent in a historical market, let it trade, and measure portfolio returns. This setup is vulnerable to two evaluation failures. First, long backtests often overlap with the knowledge cutoffs of frontier LLMs, allowing memorized tickers, dates, prices, and market narratives to substitute for investment reasoning. Second, raw returns are a noisy proxy for the ability to select stocks, since positive performance may come from market beta, style exposure, or favorable regimes rather than stock-selection alpha.

We introduce KTD-FIN (**K**nowing-**T**o-**D**oing **Fin**ancial Benchmark), an end-to-end trading benchmark based on price data that addresses both issues. To mitigate knowledge leakage when backtests cover periods before LLMs' knowledge cutoffs, KTD-FIN applies masking at the data layer to anonymize key identifiers and calendar information consistently across prompts and tools, limiting the cues agents can use to retrieve memorized market information while enabling controlled comparisons of identifier access. It also incorporates a Barra-style performance attribution framework that decomposes portfolio returns into market, style, and stock-selection alpha components. This finer-grained decomposition allows stock-selection ability to be assessed separately from market and style exposures that can confound evaluations based on raw returns.

Across ten frontier LLM agents evaluated on the Chinese A-share CSI300 over a 2024–2026 trading window, masking changes agent rationales, pushing them away from narratives about specific companies and toward factor reasoning under anonymized identifiers. Attribution analysis further shows that LLM agents' cumulative returns under identifier masking are largely explained by market and style exposure, with negative stock-selection alpha for nine of ten agents under the full nine-factor specification. Cross-model replications reproduce behavioral differences; leave-one-factor-out attribution preserves closely aligned rankings. These findings suggest that financial LLM benchmarks should evaluate not only whether an agent makes money, but also which sources drive those returns. We provide KTD-FIN as a reproducible template for evaluating LLM trading agents with leakage controls and return attribution.

1 INTRODUCTION

A growing number of recent systems treat the LLM not as a system that answers financial questions but as a trading agent: something that watches the market, decides what to look at next, and commits capital (Yu et al., 2023; Zhang et al., 2024; Li et al., 2024a). The benchmarks built around this shift fall into two groups. One keeps the model outside the market and tests how well it can read filings, answer questions, or extract structured information (Islam et al., 2023; Guo et al., 2023; Xie et al., 2023; 2024). The other puts the model inside a historical simulator, lets it trade, and reports portfolio return. As this end-to-end framing becomes the standard way to ask whether an LLM can "profit" in capital markets, two evaluation failures emerge. First, long backtests routinely overlap with the knowledge cutoffs of frontier models, so memorized tickers, prices, dates, and macro narratives can substitute for the data the benchmark thinks it is showing. Second, even when leakage

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

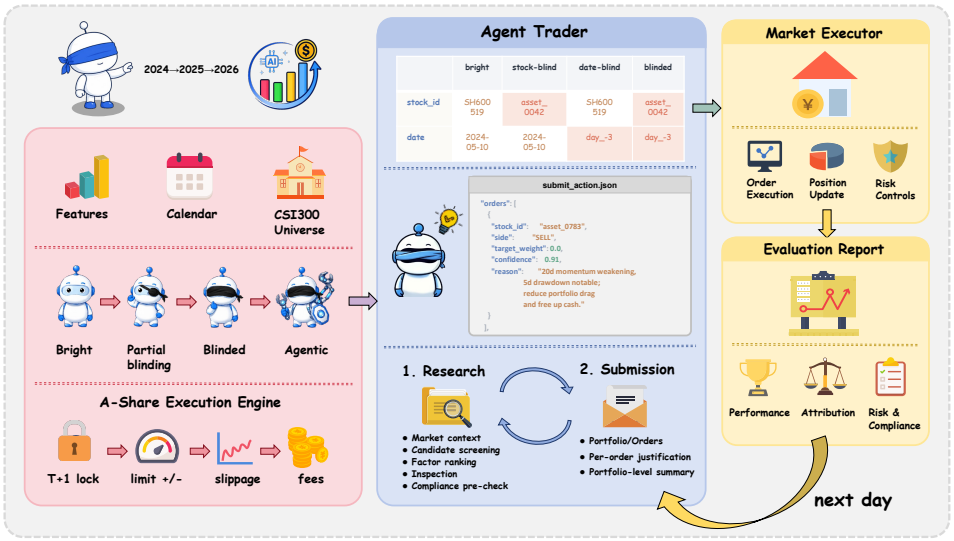

Figure 1: Overview of KTD-FIN. An LLM agent receives CSI300 features under one of four masking conditions, executes a *research*-then-*submission* loop, and submits a JSON action to a Qlib-based executor with realistic trading constraints.

is controlled, cumulative return is a noisy proxy for skill in stock selection: positive performance may reflect market beta, persistent style exposure, or a favorable regime rather than stock-specific residual performance.

Real stock tickers and dates can cue a model to recall price commentary, company descriptions, and macroeconomic news encountered during pretraining (Li et al., 2025; Ma, 2026). In our trading environment, for example, market observations consist of numerical features derived from OHLCV. Yet when real tickers are visible, one model writes *"*SH601857*—defensive blue-chip with strong downside protection, increase position"*, even though the inputs contain no such company description. We also examined trading explanations in a test covering a period after the evaluated model's knowledge cutoff. Company, industry, or business information absent from the numerical inputs appeared when stock identities were visible, but not when they were hidden. DeepFund (Li et al., 2025) evaluates models after their knowledge cutoffs, retaining identity information for realistic trading evaluation when a sufficiently long common window is available. OracleProto (Ma, 2026) shows that instructions alone do not prevent memory use in static forecasting, motivating controls over the data supplied to the model. We extend this approach to sequential trading.

Even with leakage controlled, a profitable backtest does not tell us where the gains came from. Market movements and persistent style exposures can generate positive returns even when stock selection contributes little. Empirical asset pricing separates these sources by decomposing returns into market, style, and stock-specific residual components. Market timing and style allocation can themselves reflect useful decisions; this breakdown measures their realized contributions alongside those of stock selection.

We introduce KTD-FIN (**K**nowing-**T**o-**D**oing **Fin**ancial Benchmark; Figure 1) to address these two evaluation problems. Built on Qlib, the benchmark simulates trading in the CSI300 universe of the Chinese A-share market using OHLCV histories and derived technical factors. Four masking settings either retain both stock identities and dates (*bright*), hide stock identities (*stock-blind*), hide dates (*date-blind*), or hide both (*blinded*). The same masking rules apply to prompts and tool outputs. In identity recovery tests, we ask ten models to infer the hidden stock identities and dates from the masked observations, checking whether the remaining data still reveal this information. To distinguish sources of profit, KTD-FIN uses daily Barra-style cross-sectional attribution to separate each agent's realized return into market, style, and stock selection alpha components. Three decision modes vary the information available to the agent: identifiers without price features (*memory-only*), supplied candidate stocks and signals (*fixed-candidate*), or market information retrieved through tools (*open-research*). Ten metrics summarize returns, trading behavior, and reliability.

We evaluate ten frontier LLMs under *blinded* × *open-research* over a continuous trading window from 2024-01-01 to 2026-04-10. We compare them with eighteen machine learning baselines using Alpha9 features, trained on 2008–2022 data and validated on 2023 data, as well as CSI300 buy-and-hold. In selected *bright* examples, the agent describes stocks as "defensive blue-chip" or "liquor leader"; under *blinded*, the same model instead refers to factor ranks for anonymized stocks. The full nine-factor Barra attribution shows that the agents' returns are largely explained by market and style exposure, with stock selection alpha near zero for one model and negative for the other nine. Tests on additional models reproduce the behavioral differences across masking settings and the higher aggregate returns of *open-research* relative to *fixed-candidate*. For one model evaluated over a trading period after its knowledge cutoff, median excess returns are positive while mean selection alpha is negative. Rankings by selection alpha also remain closely aligned when factors are removed from the attribution model one at a time.

Contributions. (1) KTD-FIN, an end-to-end trading benchmark that combines consistent masking of stock identities and dates across prompts and tools with Barra-style return attribution. (2) A comparative evaluation of ten LLMs, eighteen machine learning baselines using Alpha9 features, and CSI300 buy-and-hold, assessing trading performance, behavior, reliability, and selection alpha, with additional tests across models, on a trading period after a model's knowledge cutoff, and under alternative factor specifications. (3) A reproducible evaluation specification covering the trading environment, execution interface, masking rules, identity recovery tests, and dataset configuration, providing a foundation for future extensions beyond price data to financial news and other market information.

2 RELATED WORK

Financial benchmarks and trading agents. Financial LLM evaluation has expanded from testing domain knowledge and interpreting documents to assessing investment decisions. Knowledge and language understanding benchmarks measure familiarity with financial concepts (Guo et al., 2023; Zhu et al., 2024; Xie et al., 2023), while financial question answering tests reasoning over evidence in reports and tables (Islam et al., 2023; Chen et al., 2021; 2022; Zhu et al., 2021). Trading evaluation brings these capabilities into a sequential setting, where decisions change portfolio holdings and affect subsequent outcomes (Xie et al., 2024; Li et al., 2024a). Research on trading agents explores how memory and reflection support these decisions (Yu et al., 2023; Zhang et al., 2024; Li et al., 2024b), and how multiple agents can coordinate financial analysis (Li et al., 2023; Yu et al., 2024). As the evaluation moves from answers to investment outcomes, the information available to the agent and the sources of its returns become central to interpreting performance. KTD-FIN incorporates both into the evaluation protocol through controlled access to stock identities and dates and a decomposition of returns into market, style, and stock selection contributions.

Contamination in historical evaluation. Historical evaluation must account for information that a model may have encountered during pretraining. Evaluating on newly released problems or market observations provides a temporal separation between training and testing (Jain et al., 2024; Li et al., 2025). Retrospective evaluation instead requires examining what information remains accessible to the model. Membership inference methods estimate whether particular texts appeared in pretraining (Shi et al., 2024), while OracleProto (Ma, 2026) combines model cutoff constraints with temporal filtering and content checks to control information exposure in historical forecasting. For sequential trading, this control must persist as the agent requests data and receives tool outputs throughout a backtest. KTD-FIN applies consistent masking of stock identities and dates across these interactions, allowing comparisons over the same market period under different information conditions. Identity recovery tests further assess whether models can infer the original identities and dates from the masked observations.

3 BENCHMARK DESIGN

KTD-FIN evaluates daily trading in the CSI300 universe using a shared Qlib execution engine (Figure 1). We vary two aspects of the agent's input: whether real stock identifiers and calendar dates are visible, and how the agent obtains market information. Four masking settings control

identifier visibility, while three decision modes determine access to market data and tools. The underlying market data and execution rules remain the same across settings.

3.1 TRADING ENVIRONMENT AND EXECUTION

At each trading step, the agent receives its account state, current positions, execution results from the previous step, performance relative to the benchmark, and trading constraints. It then submits a JSON action specifying its orders. All market observations come from open, high, low, close, and volume (OHLCV) histories and technical features derived from them. Using numerical market inputs also allows us to check whether the agent's trading explanations refer to information absent from those inputs.

Each account starts with CNY $1{,}000{,}000$ and can hold long positions only. Shares purchased on one trading day become available for sale on the next trading day under the T+1 rule. Trading costs are 5 basis points for buys and 15 basis points for sells, with a minimum charge of CNY 5 per order. Orders execute at the next trading day's opening price.

The simulator excludes stocks at their upper price limit from `BUY` candidates and stocks at their lower price limit from `SELL` candidates, using the board-specific thresholds in Appendix H. Constraint violations and execution outcomes are logged without silently correcting invalid actions.

3.2 THREE DECISION MODES

The three modes differ in the market information available to the agent and whether it can request additional data (Table 1). All modes provide the same account information and use the same execution rules and masking options.

Table 1: Information access and decision tasks in the three modes.

Mode	Market information	Agent's task
memory-only	Stock identifiers; no price features	Trade using identifiers and account information
fixed-candidate	OHLCV and factor values for supplied candidates	Select trades from the candidate table
open-research	Market data obtained through six read-only tools	Gather information and choose trades

In *fixed-candidate*, the candidate set is the union of the top-k stocks under each factor. Comparing this mode with *open-research* examines decisions based on supplied signals and decisions based on information the agent retrieves for itself.

3.3 FOUR MASKING SETTINGS

The masking settings determine whether the agent sees real stock identifiers and calendar dates: *bright* retains both; *stock-blind* replaces stock identifiers with aliases but retains dates; *date-blind* retains stock identifiers but replaces dates with relative day indices; and *blinded* replaces both. The numerical market observations remain unchanged. These settings allow us to examine stock identity and date information separately over the same market period. Aliases stay fixed within a trading episode and are reassigned across episodes to prevent a persistent association with the same stock.

Masking applies to both prompts and tool outputs. Each episode has a mapping between real identifiers and their replacements. When the agent calls a tool, the system converts its aliases and relative dates back to the values needed to query the underlying data. Before returning the result, it applies the masking rules again. Under *blinded*, neither prompts nor tool results expose real stock identifiers or calendar dates. Returns, volatilities, drawdowns, and other numerical features are computed from the original time series; masking changes their labels, not their values.

Identity recovery tests. We ask ten models to infer the original stock identities and dates from exactly the inputs shown under *blinded* (Section 4.3). This separate test checks whether numerical price patterns or board information retained for trading rules reveal the hidden identities or dates.

3.4 RESEARCH AND ORDER SUBMISSION

In *open-research*, the agent first gathers information through six read-only tools and then submits its orders, following a ReAct-style loop (Yao et al., 2023). The initial prompt contains only account information and trading constraints. Within the configured call limit, the tools provide market and stock data, candidate screening and comparison, portfolio state, and risk checks (Appendix B).

All modes submit a JSON action containing orders and an overall rationale. Each order specifies a stock, buy or sell direction, a target weight or share count, confidence in $[0, 1]$, and a reason. Confidence is used to evaluate calibration, and reasons support the analysis of trading explanations. Appendix D gives the full schema.

Unparseable tool calls receive structured error feedback so research can continue. Submissions that violate formatting or trading constraints are retried up to the configured limit; if all retries fail, no trade is made. At execution, orders that violate universe membership, position limits, T+1, or formatting requirements are rejected and logged without aborting the step.

3.5 EVALUATION METRICS

We report ten metrics: total return, Sharpe ratio, maximum drawdown, and information ratio for performance; annualized turnover, portfolio concentration (Herfindahl–Hirschman index, HHI), cash ratio, and abstention rate for trading behavior; and parse-failure rate and expected calibration error (ECE) for reliability and confidence calibration. Tool-validity rate and the Brier score provide supplementary diagnostics. Appendix G gives their definitions.

4 EXPERIMENTS

4.1 SETUP

Evaluation windows and models. We first use Step-3.5-Flash to examine how masking and research access affect trading. The experimental design covers four masking settings, three decision modes, ten short windows (W1–W10), and five seeds, with some experimental combinations omitted. Each window spans approximately 43 trading days, and the ten windows cover different market conditions (Appendix L). W10 falls entirely after this model's knowledge cutoff.

We then compare ten frontier LLMs under the same *blinded* $\times$ *open-research* setting over a continuous window from 2024-01-01 to 2026-04-10, covering 548 trading days. This evaluation compares models over a longer trading period with the same information access and execution rules.

Additional experiments examine whether the findings from Step-3.5-Flash extend to other models. Three other models from the long-window evaluation are tested under bright and blinded open-research across all ten short windows (Appendix J). Two further models are evaluated across all three decision modes.

Eighteen ML baselines. We compare the ten LLM agents with eighteen Qlib baselines on the long window. All baselines use the same nine Alpha9 features selected through variance inflation factor (VIF) screening. We train them on 2008–2022 data and use 2023 for validation. They include linear models, gradient-boosted trees, tabular neural networks, RNNs, Transformers, and specialized time-series models. All use the same trading execution layer as the LLM agents. Table 5 presents eight representative baselines, and Appendix H reports all eighteen. CSI300 buy-and-hold serves as the market reference.

Reporting and implementation. For account metrics, we report the median across seeds, together with the first and third quartiles $[Q_1, Q_3]$ where available. To summarize results across short windows, we first take the median across seeds within each window and then the median across windows. Account returns compound over trading days. For attribution, we sum daily return contributions over the evaluation period and average these contributions across seeds.

Within each experimental setting, all LLMs use the same prompts, tools, masking rules, and execution rules through vendor APIs. The Qlib baselines are trained on one H800. Appendix I gives the evaluation configuration.

4.2 Research access and trading behavior

For Step-3.5-Flash, open-research produces higher aggregate returns than fixed-candidate under both bright and blinded observations (Table 2). The memory-only condition examines trading without price features. When real tickers are visible, the model places trades even without those features; under blinded, its median return is 0.00%.

The same advantage appears for `gpt-5.4-mini` and `hy3`: open-research has higher aggregate returns than fixed-candidate for both models under both masking settings (Table 3). This experiment covers bright and blinded, all three decision modes, five windows spanning 2024–2026 including W10, and three seeds, for a total of 180 runs.

Masking also changes trading behavior. Across 30 comparisons matched by model, decision mode, and window, the mean absolute difference between bright and blinded is 18.7 percentage points in capital utilization, 15.7 percentage points in abstention rate, and 11.41 in annualized turnover.

Table 2: Step-3.5-Flash total return (%) over the short windows, with five seeds.

Mode	Bright	Blinded
Open-research	$\mathbf{-2.11}$	$\mathbf{-1.83}$
Fixed-candidate	-3.61	-3.76
Memory-only	-0.16	0.00

Table 3: Total return (%) for two additional models over five windows, with three seeds.

Model	Mask	Open-research	Fixed-candidate
`gpt-5.4-mini`	Bright	$+0.10$	-3.80
	Blinded	-1.99	-2.74
`hy3`	Bright	-1.37	-1.76
	Blinded	-1.57	-3.91

Returns are medians across seeds within each window, then across windows. Bold marks the higher return between open-research and fixed-candidate.

4.3 Recovering identities from masked inputs

We use identity recovery tests to assess whether models can infer the information hidden by the blinded setting. Ten LLMs receive the same masked inputs available to the trading agent for 200 stock/date cases. They attempt to identify the original stock ticker, date, sector, and trading board.

Ticker top-5 recovery counts a success when the true ticker appears among a model's five guesses. The highest recovery rate across the ten models is 10.2%, compared with 1.4% for random guessing. Joint recovery requires both a correct ticker among the five guesses and a date within ± 7 trading days of the true date. The highest joint recovery rate is 1.5%, compared with 0.0% for random guessing. When only the first ticker guess counts, the highest recovery rate is 3.0%. Appendices K and N provide the full results and visualization.

4.4 Trading after the model's knowledge cutoff

W10 evaluates Step-3.5-Flash over 43 trading days from 2026-02-02 to 2026-04-10, entirely after its knowledge cutoff. Both bright and blinded produce positive median returns in excess of CSI300, while selection alpha is negative in all five seeds (Table 4). In this period, outperforming the index therefore coexists with a negative contribution from stock selection under the attribution model.

We also inspect the explanations attached to the final trading actions. The audit counts explicit company, sector, or business descriptions that are absent from the numerical inputs. Each masking condition has 215 trading steps across the five seeds. These descriptions appear in 14 steps under bright and 9 under date-blind, compared with none under either stock-blind or blinded. With the underlying market data held fixed, such references appear only when real stock identities are visible, even in this period after the model's cutoff. Appendix M reports the full counts and trading behavior.

Table 4: Step-3.5-Flash results on W10 with five seeds. Account returns are medians across seeds; selection alpha is the mean additive contribution across seeds.

Statistic	Bright	Blinded
Account total return (median)	$+1.01\%$	$+4.67\%$
CSI300 return	$+0.66\%$	$+0.66\%$
Account excess return (median)	$+0.35\%$	$+4.01\%$
Selection alpha (mean)	-7.70%	-5.30%
Seeds with negative selection alpha	$5/5$	$5/5$

Table 5: Account performance over the long window, sorted by total return separately for LLM agents and Qlib baselines. Eight of the eighteen Qlib baselines are shown; Table 10 reports all eighteen. MDD denotes maximum drawdown. α rank compares LLMs by selection alpha under the full nine-factor model.

Model	Total return	Sharpe	MDD	α rank
Ten LLM agents: blinded × open-research				
`qwen3.6-plus`	$+85.29\%$	1.13	-24.49%	2
`gpt-5.5`	$+61.26\%$	0.98	-24.63%	6
`doubao-seed-2.0`	$+61.01\%$	1.15	-15.62%	3
`claude-opus-4-7`	$+58.80\%$	0.98	-26.69%	1
`minimax-m2.7`	$+56.31\%$	0.89	-29.43%	4
CSI300 buy-and-hold	$+36.92\%$	—	—	—
`step-3.5-flash`	$+21.86\%$	0.49	-28.99%	5
`gemini-3.1-pro`	$+8.25\%$	0.29	-49.67%	7
`deepseek-v4-pro`	$+1.39\%$	0.19	-42.35%	8
`glm-5.1`	-3.55%	0.10	-44.88%	9
`kimi-k2.6`	-24.23%	-0.34	-45.49%	10
Qlib: Alpha9, 2008–2022 train, 2023 validation				
`SFM`	$+86.58\%$	2.02	-7.41%	—
`DoubleEnsemble`	$+73.66\%$	1.41	-18.24%	—
`CatBoost`	$+72.97\%$	1.43	-16.74%	—
`LightGBM`	$+66.22\%$	1.29	-15.59%	—
`Transformer`	$+60.35\%$	1.61	-11.29%	—
`LSTM`	$+43.92\%$	1.12	-11.76%	—
CSI300 buy-and-hold	$+36.92\%$	—	—	—
`MLP`	$+34.08\%$	0.69	-21.75%	—
`Linear`	$+29.60\%$	0.74	-13.88%	—

4.5 Cross-model leaderboard

Five of the ten LLM agents earn more than the CSI300 buy-and-hold return of $+36.92\%$. Qwen3.6-Plus has the highest total return among the LLMs at $+85.29\%$, with a Sharpe ratio of 1.13. Among the ML baselines, SFM has the highest total return at $+86.58\%$, with a Sharpe ratio of 2.02 (Table 5). Figure 2 shows the LLM equity curves, and Table 15 reports additional behavioral and calibration metrics. The following attribution analysis examines the sources of returns within the LLM group.

4.6 Barra-style return attribution

To examine where returns come from, we fit a weighted least-squares cross-sectional regression over investable stocks each day:

$$r_{i,t} = f_{0,t} + \sum_{k=1}^{K} x_{i,k,t-1}\lambda_{k,t} + \varepsilon_{i,t}. \tag{1}$$

Here $r_{i,t}$ is the stock's close-to-close return. The intercept $f_{0,t}$ captures the common return across stocks and is estimated by the regression; it is not the CSI300 index return. Each $x_{i,k,t-1}$ is a style exposure measured before day t, winsorized, and standardized across stocks. The coefficient $\lambda_{k,t}$ is the estimated return to that factor, and $\varepsilon_{i,t}$ is the stock's residual return.

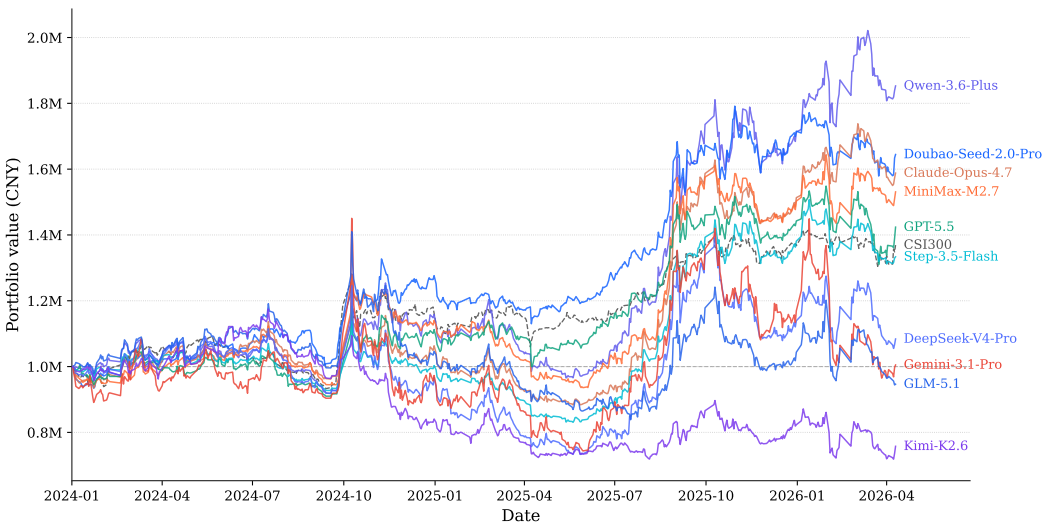

Figure 2: Daily equity (median across seeds) under blinded × open-research on the long window, starting from CNY 1,000,000. Dashed grey: CSI300.

Table 6: Seed-averaged additive Barra contributions (%) under the full nine-factor model. Com. is the common component, Sty. is style, and Port is their sum with selection alpha. Port uses the attribution path and differs from compounded, seed-median account returns in Table 5.

Model	Com.	Sty.	Sel. α	Port
Claude Opus 4.7	+35.0	+16.6	+0.2	+51.8
Qwen3.6-Plus	+41.8	+29.2	−0.7	+70.3
Doubao-Seed-2.0	+42.5	+15.5	−4.7	+53.3
MiniMax-M2.7	+29.6	+26.4	−7.4	+48.6
Step-3.5-Flash	+29.5	+17.5	−13.7	+33.3
GPT-5.5	+40.0	+11.8	−14.9	+36.9
Gemini-3.1-Pro	+38.3	+23.6	−38.5	+23.4
DeepSeek-V4-Pro	+38.5	+23.1	−43.0	+18.6
GLM-5.1	+39.5	+27.5	−61.9	+5.1
Kimi-K2.6	+32.1	+22.0	−77.8	−23.7

We aggregate these components using the portfolio weights in place before the day's return. This gives the common, style, and stock selection contributions for each agent. The stock selection component is the portfolio-weighted residual:

$$\alpha_t^p = \sum_i w_{i,t-1}^p \varepsilon_{i,t}.$$

The model uses nine factors derived from price data, covering momentum, volatility, liquidity, reversal, skewness, and volume–price interaction (Table 8). VIF screening uses data from before the evaluation period.

Under the full nine-factor model, nine of the ten agents have negative selection alpha, while all ten have positive common and style contributions (Table 6). Rankings by total return and selection alpha also differ. GPT-5.5 ranks second by account return but sixth by selection alpha; Claude Opus 4.7 ranks fourth by return and first by selection alpha.

4.7 Ranking stability across factor specifications

We repeat the attribution nine times, removing one factor at a time from the full model. Each specification uses the same trading trajectories and averaging across seeds. Across these specifications, the number of models with negative selection alpha ranges from six to ten, compared with nine

under the full model. Rankings remain largely unchanged, with Spearman correlations of 0.964–1.000 relative to the full model. Omitting intraday momentum (MOM_ID) raises alpha by 10.16 percentage points on average, the largest shift among the nine specifications. Relative rankings are therefore more stable than alpha magnitudes and signs. Appendix O reports the summary table and the comparison without style factors.

5 Discussion

5.1 How stock identities affect trading

In W10, company or business descriptions absent from the numerical inputs appear in *bright* explanations but not under *blinded*, even after Step-3.5-Flash's knowledge cutoff. Evaluation dates and identifier access therefore require separate controls. Identity recovery tests further assess whether models can infer the hidden stock identities and dates from masked inputs.

The effects also appear in portfolio decisions. The W10 and cross-model comparisons show differences in capital utilization, turnover, and abstention across masking settings. Comparisons across decision modes further show how access to price features and research tools affects investment and trading activity. Holding the market period and underlying price data fixed allows us to examine these changes under different information conditions.

5.2 Interpreting trading contributions

The LLM agents' returns are largely explained by market and style exposure. Most models have negative selection alpha under the full nine-factor model, while all ten receive positive common and style contributions.

Agent rankings remain largely unchanged when individual factors are removed, while alpha magnitudes and signs depend on which exposures are accounted for.

5.3 Portfolio exposures

Average factor exposures on days with stock holdings show distinct portfolio preferences: LLMs favor intraday momentum and higher realized volatility, while Qlib models generally favor lower volatility (Appendix F). These differences show how the two groups construct portfolios under the same execution rules, helping interpret their returns and risk.

6 Conclusion

We introduce KTD-Fin, a benchmark for LLM trading agents that combines masking of stock identities and dates with return attribution. Our evaluation compares ten LLMs with eighteen trained ML baselines. Controlled masking experiments show changes in trading explanations and capital allocation. Market and style exposures explain much of the LLM agents' returns, with nine of ten models showing negative stock selection alpha under the full nine-factor model. Rankings by selection alpha remain largely unchanged when individual factors are removed. These findings show why evaluating trading agents requires examining the information behind their decisions and the sources of their returns alongside overall profitability. KTD-Fin provides a reproducible setting for this evaluation.

7 Limitations

Our experiments focus on price-based trading in the CSI300 universe. Evaluation in other markets remains future work. We are exploring ways to incorporate news and financial reports, using their original publication times to determine when agents can access them. Selection alpha depends on the nine price-derived factors used for attribution. Broader factor specifications and return attribution for the ML baselines would extend the comparison.

AI USE STATEMENT

An LLM assistant was used to help convert the manuscript to the ICLR format, edit prose, and incorporate analyses from author-provided experimental summaries. This assistance did not include running new trading experiments or generating new numerical results. The reported experimental values were taken from the existing manuscript and author-provided analyses. The authors are responsible for the final text, claims, and artifacts.

ETHICS STATEMENT

This benchmark studies historical trading simulations. Its results do not establish the suitability of any agent for live trading. Financial deployment can expose users to losses, and evaluation should account for transaction costs, execution constraints, information timing, and changing market conditions. The price-only setting excludes private personal information and does not evaluate human subjects.

REPRODUCIBILITY STATEMENT

Section 3 specifies the observation channel, masking conditions, decision modes, and execution rules. Section 4 describes evaluation windows, model coverage, aggregation, and baselines. Appendices B–I provide tool schemas, prompts, a masked trace, metric definitions, factor descriptions, baseline configuration, and harness settings. Additional masking, post-cutoff, and factor-sensitivity results are reported in Section 4.2, Section 4.4, Section 4.7, and the corresponding appendices.

REFERENCES

Zhiyu Chen, Wenhu Chen, Charese Smiley, Sameena Shah, Iana Borova, Dylan Langdon, Reema Moussa, Matt Beane, Ting-Hao Huang, Bryan R. Routledge, and William Yang Wang. FinQA: A dataset of numerical reasoning over financial data. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2021. URL https://aclanthology.org/2021.emnlp-main.300/.

Zhiyu Chen, Shiyang Li, Charese Smiley, Zhiqiang Ma, Sameena Shah, and William Yang Wang. ConvFinQA: Exploring the chain of numerical reasoning in conversational finance question answering. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2022. URL https://aclanthology.org/2022.emnlp-main.421/.

Xin Guo, Haotian Xia, Zhaowei Liu, Hanyang Cao, Zhi Yang, Zhiqiang Liu, Sizhe Wang, Jinyi Niu, Chuqi Wang, Yanhui Wang, Xiaolong Liang, Xiaoming Huang, Bing Zhu, Zhongyu Wei, Yun Chen, Weining Shen, and Liwen Zhang. Fineval: A chinese financial domain knowledge evaluation benchmark for large language models, 2023.

Pranab Islam, Anand Kannappan, Douwe Kiela, Rebecca Qian, Nicolas Scherrer, and Bertie Vidgen. Financebench: A new benchmark for financial question answering, 2023.

Naman Jain, King Han, Alex Gu, Wen-Ding Li, Fanjia Yan, Tianjun Zhang, Sida Wang, Armando Solar-Lezama, Koushik Sen, and Ion Stoica. LiveCodeBench: Holistic and contamination free evaluation of large language models for code, 2024.

Changlun Li, Yao Shi, Chen Wang, Lingling Zhao, Bo Zheng, Minyi Qin, Jiayi Sun, Wentao Zhang, Xiang Liu, and Tie-Yan Liu. Time travel is cheating: Going live with deepfund for real-time fund investment benchmarking, 2025.

Haohang Li, Yupeng Cao, Yangyang Yu, Xiaomin Zhou, and Xiao-Yang Liu. Investorbench: A benchmark for financial decision-making tasks with llm-based agent, 2024a.

Yang Li, Yangyang Yu, Haohang Li, Zhi Chen, and Khaldoun Khashanah. TradingGPT: Multi-agent system with layered memory and distinct characters for enhanced financial trading performance, 2023.

Yuan Li, Bingqiao Luo, Qian Wang, Nuo Chen, Xu Liu, and Bingsheng He. CryptoTrade: A reflective LLM-based agent to guide zero-shot cryptocurrency trading. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2024b. URL https://aclanthology.org/2024.emnlp-main.63/.

Yiding Ma. Oracleproto: A reproducible framework for benchmarking llm native forecasting via knowledge cutoff and temporal masking, 2026. https://github.com/MaYiding/OracleProto.

Weijia Shi, Anirudh Ajith, Mengzhou Xia, Yangsibo Huang, Daogao Liu, Terra Blevins, Danqi Chen, and Luke Zettlemoyer. Detecting pretraining data from large language models. In *International Conference on Learning Representations (ICLR)*, 2024. URL https://openreview.net/forum?id=zWqr3MQuNs.

Qianqian Xie, Weiguang Han, Zhengyu Zhang, Yanzhao Lai, Min Peng, Alejandro Lopez-Lira, and Jimin Huang. Pixiu: A large language model, instruction data and evaluation benchmark for finance, 2023.

Qianqian Xie, Weiguang Han, Yanzhao Lai, Min Peng, and Jimin Huang. The finben: An holistic financial benchmark for large language models, 2024.

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik Narasimhan, and Yuan Cao. React: Synergizing reasoning and acting in language models. In *International Conference on Learning Representations*, 2023.

Yangyang Yu, Haohang Li, Zhi Chen, Yuechen Jiang, Yang Li, Denghui Zhang, Rong Liu, Jordan W. Suchow, and Khaldoun Khashanah. FinMem: A performance-enhanced LLM trading agent with layered memory and character design, 2023. URL https://arxiv.org/abs/2311.13743.

Yangyang Yu, Zhiyuan Yao, Haohang Li, Zhiyang Deng, Yupeng Cao, Zhi Chen, Jordan W. Suchow, Rong Liu, Zhenyu Cui, Zhaozhuo Xu, Denghui Zhang, Khaldoun Khashanah, Linyi Yang, Yi Yang, Yinheng Xiao, Jianyun Wang, Yangyang Yang, Mengyou Wang, Daniel Wei, and Xiao-Yang Liu. FinCon: A synthesized LLM multi-agent system with conceptual verbal reinforcement for enhanced financial decision making. In *Advances in Neural Information Processing Systems 37 (NeurIPS)*, 2024. URL https://arxiv.org/abs/2407.06567.

Wentao Zhang, Lingxuan Zhao, Haochong Xia, Shuo Sun, Jiayi Sun, Minyi Qin, Xinyi Li, Xue Zhao, Jun Cai, Bo Zheng, Lingling Zhao, Xiang Liu, Bo An, and Tie-Yan Liu. A multimodal foundation agent for financial trading: Tool-augmented, diversified, and generalist, 2024.

Fengbin Zhu, Wenqiang Lei, Youcheng Huang, Chao Wang, Shuo Zhang, Jiancheng Lv, Fuli Feng, and Tat-Seng Chua. TAT-QA: A question answering benchmark on a hybrid of tabular and textual content in finance. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics (ACL-IJCNLP)*, 2021. URL https://aclanthology.org/2021.acl-long.254/.

Jie Zhu, Junhui Li, Yalong Wen, and Lifan Guo. Benchmarking large language models on CFLUE – a Chinese financial language understanding evaluation dataset. In *Findings of the Association for Computational Linguistics: ACL 2024*, pp. 5673–5693, 2024. URL https://aclanthology.org/2024.findings-acl.337/.

A TRADING RATIONALES UNDER FOUR MASKING SETTINGS

The following examples show the explanations agents give for their trading decisions under the four masking settings. *Bright* retains real stock tickers and calendar dates; *stock-blind* hides tickers; *date-blind* hides dates; and *blinded* hides both. We reproduce the selected explanations verbatim so that readers can compare the company information, numerical features, and portfolio considerations cited in each setting.

BRIGHT **real ticker IDs visible** **(4 verbatim rationales)**

[SH600519 BUY] "**Kweichow Moutai**, *liquor leader*; low volatility (0.0234), drawdown well controlled (-7.3%), 5-day return $+15.2\%$ ranks among the top—suitable as a steady allocation."

[SH600900 BUY] "**Yangtze Power**, low volatility 0.01, 5-day $+3.1\%$, 20-day $+6.8\%$; in a weak market, add a new 5% position as a *defensive utility* core holding."

[SH601939 BUY] "**CCB** 20-day momentum $+11\%$ tops the screen, low volatility, shallow drawdown—fits a *large-cap low-vol high-momentum* style."

[SH688981 BUY] "**Semiconductor manufacturing leader**, beneficiary of an *industry-recovery expectation*; 20-day return $+108\%$, the Oct-18 limit-up move shows strong drive, volatility 8.31% relatively contained—suitable for allocation."

STOCK-BLIND **anonymized ticker, real date** **(3 verbatim rationales)**

[asset_0004 SELL] "asset_0004 current weight 10.38% exceeds the 10% single-name cap, must reduce 160 shares to comply; the name has 20-day return $+23.7\%$ and a strong showing, but *valuation is now elevated*—take partial profit, retain the core."

[asset_0017 BUY] "20-day return tops the screen at $+33.6\%$, long-run trend is clear; despite weaker short-term momentum the *fundamental support remains strong*."

[asset_0023 BUY] "5-day return $+12.9\%$ ranks #1, short-term momentum strong; despite a sharper 20-day drawdown, the *current valuation is reasonable*—allocate moderately to catch the trend."

DATE-BLIND **real ticker, anonymized date** **(3 verbatim rationales)**

[SH600111 BUY] "**SH600111** has low volatility and *strong defensive properties*; the *rare-earth sector* carries allocation value, helps diversify portfolio risk."

[SH601857 BUY] "**SH601857** 20-day return $+21.4\%$, low volatility 0.0211, drawdown controlled at -5.45%; *low valuation, high safety margin*."

[SH600362 BUY] "**SH600362** short-term momentum strong and volatility low; 20-day return $+27.6\%$ with controlled drawdown—replace some *high-valuation positions*."

BLINDED **anonymized aliases** **(4 verbatim rationales)**

[asset_0006 BUY] "asset_0006 has 20-day return $+27.4\%$, low volatility 0.0255, and small drawdown -3.8%; risk/return ratio excellent, suitable to add up to the 10% cap."

[asset_0013 BUY] "asset_0013 5-day return $+14.5\%$ ranks #2, 20-day $+27.6\%$ ranks #1, low volatility and a maximum drawdown of only -3.87%; risk/return ratio superior, build a new 8% position."

[asset_0028 SELL] "asset_0028 20-day return only $+14.4\%$ ranks low and the unrealised loss is largest; risk/return unfavourable, close the position to release capital."

[asset_0011 SELL] "asset_0011 has the largest unrealised loss and is not in the top-15 by 20-day return; underperforms the market average, replace with a stronger candidate."

The examples with visible tickers refer to company names, business activities, or sector characteristics. The blinded examples refer to returns, volatility, factor ranks, and current holdings. Some stock-blind examples make general claims about valuation or fundamentals that are absent from the numerical inputs. The W10 audit counts explicit company, sector, and business references absent from the inputs across the four settings (Table 17).

B TOOL INTERFACE SPECIFICATION

In *open-research*, agents can request market and portfolio information through six read-only tools (Table 7). Before a data query, the system converts masked stock identifiers and dates in the tool arguments back to their original values. It then masks the results before returning them to the agent. Under *blinded*, the agent sees stock aliases and relative date labels throughout these interactions. All tools return JSON objects with predefined field types.

Table 7: Six read-only tools under *open-research*. The long-window evaluation allows up to 99 tool calls per trading step.

Tool	Main arguments	Returned fields
`get_market_context`	—	indices, calendar, sector tone
`screen_candidates`	factors, k	ranked aliases + factor values
`get_stock_snapshot`	alias, lookback	OHLCV bars, factors, previous close
`compare_candidates`	aliases, dims	metrics for the selected stocks
`portfolio_state`	—	holdings, cash, T+1 selling restrictions
`risk_check`	target weights	violations, projected limits

C PROMPTS AND AN EXAMPLE TRADING STEP

The system prompts below correspond to the three decision modes in Section 3.2. All three modes use the same user-message template.

The prompts are reproduced as used in the experiments. Although the open-research prompt mentions news, reports, and search tools, only the six tools in Table 7 are enabled for this mode. Market observations are derived from price and volume data; no news articles or financial reports are supplied.

SYSTEM PROMPT: OPEN-RESEARCH (HEADLINE CONFIGURATION)

```
You are an autonomous Chinese A-share trading agent. Within the
current backtest trading day, conduct research, judgement, order
submission, and self-constraint based on available data; emit a
single valid JSON trading decision.

## Goals
- Maximise the overall benchmark score within the hard constraints.
- Decide on your own whether to trade, how many names, concentrated
  or diversified, and whether to use `target_weight` or `shares`.
- If no identifiable edge exists, holding cash or maintaining the
  status quo is an acceptable decision.

## Degrees of freedom
- No fixed workflow is required; decide which tools to call, in what
  order and frequency, and when to stop research and submit.

## Information boundary
- Decisions may rely only on the current message, system context,
  and tool returns; do not assume any externally available
  information.
- All tools have time semantics strictly earlier than the current
  backtest day: market-data tools see the previous trading day and
  earlier; news / report / announcement / search tools have their
  returns filtered as of the current day, but you should still
  verify the date fields in returns to avoid indirectly referencing
  future information.
- The current day's open, close, and intraday data are not visible.

## Hard constraints
1. Long-only; no shorting.
2. Only stocks in today's tradable universe may be operated on.
3. A-share T+1: a stock bought today can only be sold the next day.
4. Each order must contain `stock_id`, `side`, `confidence`, and
   `reason`, with `reason` of at least 10 characters.
5. If no action is intended, return an empty `orders` list.

## Autonomous decision space
- Neither concentration nor the number of holdings has a hard upper
  bound; you may concentrate heavily (e.g., 1-3 names) or diversify
  broadly. Weigh win rate, risk budget, and candidate quality on
  your own.
- State your portfolio-construction rationale in `overall_reason`.

## Limit-board rules (engine layer)
- Price limits are board-differentiated: main / SME (6xxxxx / 00xxxx)
  +/-9.5%; ChiNext (30xxxx) and STAR (688xxx) +/-19.5%; Beijing
  Stock Exchange (8xxxxx / 4xxxxx) +/-29.5%.
- A next-day open at the upper limit rejects BUY; lower limit
  rejects SELL; one-sided days (open == high == low at a limit
  price) are unfillable.
- Tools expose `board / limit_pct / limit_up / limit_down` and
  limit-hit warnings per stock.
```

```
## Execution details
- Orders execute sequentially in the order given; on days with both
  buys and sells, list SELL before BUY to avoid running out of cash.
- `target_weight` is the final desired weight; do not re-submit BUY
  for a position whose current weight is already near or above it.
- When opening longs, retain a 1%-3% cash buffer (fees, lot rounding,
  spread).
- SELL only against holdings with `shares_available > 0`.
- When cash or sellable shares are uncertain, prefer to under-trade,
  lower the target, or do nothing.

## Decision requirements
- Emit an executable trading decision, not an analytical report.
- Order intentions must be clear, non-conflicting, and free of
  meaningless instructions.
- `confidence` in [0, 1]; `overall_reason` >= 20 characters.

## Output format
Emit a single JSON object only, starting with `{` and ending with
`}`; no markdown code-block wrappers, no extra text, no comments,
no tool-call summaries. See Appendix for the full action schema.
```

SYSTEM PROMPT: FIXED-CANDIDATE

```
You are a professional Chinese A-share quantitative analyst. The
current experimental mode is fixed-candidate: a fixed candidate
pool is provided; you cannot call tools and cannot extend the
candidate set on your own.

Important notes
- Stock identifiers appear in the form provided in the input
  (e.g., asset_0042); use the input identifiers verbatim.
- The current step identifier may be a real date or a relative
  index. Do not use external memory, news, real-stock common
  knowledge, or any data not provided.
- BUY only from the fixed candidate pool; SELL only against current
  holdings; an empty `orders` list is acceptable.
- Long-only; A-share T+1 (sell the next day at the earliest);
  concentration and number of holdings have no hard upper bound,
  decide on your own.
- Each order must contain `stock_id`, `side`, `confidence`, and
  `reason`; `target_weight` and `shares` are mutually exclusive.

The output must be a single pure JSON object (no markdown, no
extra text).
```

SYSTEM PROMPT: MEMORY-ONLY

```
You are a professional Chinese A-share quantitative analyst. The
current experimental mode is memory-only: the system provides only
today's tradable universe (identifiers), your account state, and
constraints. No price, volume, return, volatility, factor, or
indicator data is provided, and no tool calls are permitted.

Important notes
- The input contains only stock identifiers. An identifier may be
  a real ticker (e.g., SH600519) or an anonymous alias (e.g.,
  asset_0042); use the input identifiers verbatim.
- The current step identifier may be a real date or a relative
  index.
- Your decision may rely only on the identifiers, account state,
```

```
  and constraints in the input. If you have a view on a given
  identifier, you may act on it; if you have no basis for a
  decision, you may return an empty `orders` list.
- BUY only from today's tradable universe (identifiers present in
  the input); SELL only against current holdings; no action is
  always allowed.
- Long-only; A-share T+1; concentration and number of holdings have
  no hard upper bound.
- Each order must contain `stock_id`, `side`, `confidence`, and
  `reason`; `target_weight` and `shares` are mutually exclusive.

The output must be a single pure JSON object (no markdown, no
extra text).
```

USER-MESSAGE TEMPLATE

The placeholders in `{...}` are filled with the current trading step's data. Under *blinded*, `date_label` is a relative day index, such as `day_+12`. Stock identifiers in `positions_table`, `universe_preview`, and `features_table` use `asset_NNNN` aliases.

```
## Backtest step: {date_label}

## Constraint summary
- Concentration and number of holdings have no hard upper bound;
  decide on your own.
- Short selling: {allow_short}
- T+1 settlement: yes

## Current holdings and account
- Total value: {total_value} CNY
- Cash: {cash} CNY ({cash_weight} of NAV)
- {num_positions} positions:
{positions_table}

## Today's tradable universe ({universe_size} names)
{universe_preview}

## Market summary over the last {obs_window} observations
## ({features_count} names; data cutoff T-1 close)
{features_table}

## Previous-step execution outcome
{prev_execution_summary}

## Portfolio performance
- Cumulative return since the start of the backtest:
  {portfolio_ret_cum}
- CSI300 cumulative return over the same window:
  {benchmark_ret_cum}

Emit today's trading decision based on the above (pure JSON, no
markdown).
```

EXAMPLE TRADING STEP: BLINDED × OPEN-RESEARCH

Box C shows excerpts from Step-3.5-Flash's first trading step in W1 under *blinded* × *open-research*. The model makes four tool calls and then submits an action. We show the calls to `get_market_context`, `screen_candidates`, and `risk_check`, followed by the final action. The call to `compare_candidates` is omitted.

BLINDED × OPEN-RESEARCH **anchor model, W1 step 0** **agent's view**

[1] tool_call `get_market_context()`

[1] tool_result `{as_of_date: day_+0, data_cutoff: day_-3, contains_current_day_market_data: false, market: csi300, universe_size: 300, ...}`

[2] tool_call `screen_candidates(sort_by=ret_20d, top_k=15)`

[2] tool_result `{as_of_date: day_+0, data_cutoff: day_-3, candidates: [{stock_id: asset_0001, prev_close: 3.034, ret_1d: 0.0185, ret_5d: 0.1123, ret_20d: 0.3355, vol_20d: 0.0302, drawdown_20d: −0.055, rank: 1}, {stock_id: asset_0002, prev_close: 0.567, ret_1d: 0.0996, ret_5d: 0.2213, ret_20d: 0.3104, vol_20d: 0.0272, drawdown_20d: −0.045, rank: 2}, ...}`

[3] tool_call `risk_check(draft_orders=[{stock_id: asset_0001, side: BUY, target_weight: 0.10, confidence: 0.85, reason: "20-day return ranks #1 (+33.6%); short-term momentum strong; core allocation"}, {stock_id: asset_0002, side: BUY, target_weight: 0.10, confidence: 0.82, reason: "1-day +9.96% and 5-day +22.1% both rank #1; explosive short-term strength"}, ...])`

[3] tool_result `{valid: true, violations: [], projected_weights: {asset_0001: 0.10, asset_0002: 0.10, ...}}`

[final] submit_action `{orders: [{stock_id: asset_0001, side: BUY, target_weight: 0.10, confidence: 0.85, reason: "..."}, {stock_id: asset_0002, side: BUY, target_weight: 0.10, confidence: 0.82, reason: "..."}, ...], overall_reason: "High-momentum, low-volatility names dominate; build five-name core positions and retain cash buffer."}`

The example shows how masking is applied throughout a trading step (Section 3.3). Tool arguments, tool results, and the final action use stock aliases. The date fields `as_of_date` and `data_cutoff` contain relative day indices. The model's explanations refer to numerical features, including ranks, returns, and volatility. The execution engine maps aliases back to real tickers internally when processing orders.

D ACTION SCHEMA

At the end of each trading step, the agent submits one JSON object, which the system checks against the action schema. The schema illustration below shows an order sized by `target_weight`; `BUY | SELL` lists the two allowed values of `side`.

```
{
  "orders": [
    { "stock_id":      "asset_0042",
      "side":          "BUY" | "SELL",
      "target_weight": 0.05,
      "confidence":    0.75,
      "reason":        "..." }
  ],
  "overall_reason": "..."
}
```

An order specifies its size using either `target_weight` or `shares`, never both. We use `confidence` to calculate the calibration metrics (Appendix G) and `reason` to examine trading explanations, including the cases in Appendix A.

E STYLE FACTORS

Table 8 lists the nine factors derived from price data that we use for return attribution. We select them using variance inflation factor (VIF) screening on data from before the evaluation period.

Table 8: Nine style factors used in the cross-sectional attribution model.

Factor	Description
MOM_12_1	12-minus-1 month momentum
RV_60	60-day realized volatility
ILLIQ	Amihud illiquidity
REV_ON	Overnight return
MOM_ID	Intraday momentum
SKEW	Negative return skewness
CORR_PV	Volume–return correlation
HIGH_52W	52-week high anchor
CV_VOL	Amount coefficient of variation

F FACTOR EXPOSURES OF ML AND LLM PORTFOLIOS

For each model, we weight the standardized factor values by portfolio holdings and average the resulting exposures over days with stock holdings. We then average each model's exposures within the Qlib and LLM groups. Table 9 reports these group averages for the comparison in Section 5.3.

Table 9: Mean factor exposures for the eighteen Qlib models and ten LLMs. Positive values indicate exposure above the stock-universe average, and negative values indicate exposure below it. All portfolios are long-only. Bold values mark differences greater than 0.4 standard deviations in absolute value.

Factor	ML	LLM	ML − LLM
MOM_12_1	$+0.34$	$+0.40$	-0.07
RV_60	-0.66	$+0.23$	$\mathbf{-0.89}$
ILLIQ	-0.34	-0.32	-0.02
REV_ON	$+0.12$	$+0.28$	-0.17
MOM_ID	-0.12	$+1.06$	$\mathbf{-1.17}$
SKEW	$+0.42$	-0.08	$\mathbf{+0.50}$
CORR_PV	-0.26	$+0.33$	$\mathbf{-0.59}$
HIGH_52W	$+0.80$	$+0.77$	$+0.03$
CV_VOL	-0.60	$+0.11$	$\mathbf{-0.71}$

G EVALUATION METRICS

The main evaluation reports ten metrics covering returns, trading behavior, and reliability. Tool-validity rate and the Brier score provide supplementary diagnostics.

Returns and risk. *Total return* is the compounded return over the evaluation period, calculated from end-of-day net asset value (NAV). *Sharpe ratio* is annualized using a zero risk-free rate ($r_f = 0$) and 252 trading days per year. *Maximum drawdown* is the largest decline in NAV from a previous peak. *Information ratio* is the annualized mean daily return in excess of CSI300 divided by its annualized standard deviation.

Behavior. *Annualized turnover* sums the absolute changes in portfolio weights at each step and scales the total to a 252-day year. *HHI* measures portfolio concentration: we calculate $\sum_i w_i^2$ over current holding weights at each step, then average across steps. *Cash ratio* is the average fraction of NAV held as cash. *Abstention rate* is the fraction of steps that produce no orders after retries.

Reliability and calibration. *Parse-failure rate* is the fraction of submitted actions that fail JSON schema validation after the allowed retries. *Tool-validity rate* is the fraction of tool calls with arguments that parse and are successfully passed to the requested tool. For *ECE*, we divide per-order `confidence` values into 10 equal-width bins and calculate the weighted average of the absolute gap between each bin's mean confidence and accuracy. *Brier score* evaluates each order's confidence as the probability that its realized return over the next period is positive.

H QLIB FACTOR MODEL CONFIGURATION

All eighteen Qlib baselines use the same Alpha9 features, with 2008–2022 data for training and 2023 for validation. Model hyperparameters follow the official Qlib model-zoo configurations for CSI300. Each model produces stock scores, which are passed to the same portfolio construction procedure: select the top 20 stocks and rebalance using a cost-aware threshold of 0.06. Table 10 reports all eighteen baselines; Table 5 presents eight representative models.

A-share price-limit thresholds (used by both LLM agents and Qlib baselines). The execution engine applies a small buffer below each board's official daily limit to avoid fills at unreachable prices: 9.5% for main-board issues (Shanghai `6xxxxx` and Shenzhen `00xxxx`, daily limit $\pm 10\%$), 19.5% for ChiNext (`30xxxx`) and STAR Market (`688xxx`, both $\pm 20\%$), and 29.5% for the Beijing Stock Exchange (`8xxxxx`, $\pm 30\%$).

Table 10: Performance of all eighteen Qlib baselines from 2024-01-01 to 2026-04-10, sorted by total return. CSI300 buy-and-hold is shown at its position in the ranking.

Model	Total ret.	Excess	Sharpe	MDD	IR	Turn.
SFM	+86.58%	+49.66%	2.02	−7.41%	+0.57	22.47
DoubleEnsemble	+73.66%	+36.74%	1.41	−18.24%	+0.41	72.84
CatBoost	+72.97%	+36.05%	1.43	−16.74%	+0.40	29.96
XGBoost	+66.41%	+29.49%	1.12	−17.75%	+0.31	101.66
LightGBM	+66.22%	+29.30%	1.29	−15.59%	+0.33	46.18
ADARNN	+66.13%	+29.21%	1.66	−8.52%	+0.35	16.19
KRNN	+62.09%	+25.17%	1.63	−8.88%	+0.31	20.80
Transformer	+60.35%	+23.43%	1.61	−11.29%	+0.27	11.22
Sandwich	+56.57%	+19.65%	1.49	−8.90%	+0.24	29.78
ADD	+54.03%	+17.11%	1.47	−9.92%	+0.20	22.52
Localformer	+53.02%	+16.10%	1.41	−10.59%	+0.19	13.32
TCN	+52.33%	+15.41%	1.31	−10.27%	+0.18	30.17
TabNet	+52.25%	+15.33%	1.38	−10.61%	+0.18	26.13
LSTM	+43.92%	+7.00%	1.12	−11.76%	+0.06	78.03
CSI300 buy-and-hold	+36.92%	0.00%	—	—	—	0.00
MLP	+34.08%	−2.84%	0.69	−21.75%	−0.01	18.48
GRU	+34.05%	−2.87%	0.92	−13.50%	−0.07	57.79
Linear	+29.60%	−7.32%	0.74	−13.88%	−0.12	15.61
ALSTM	+22.67%	−14.25%	0.64	−15.15%	−0.22	95.31

I Evaluation configuration

Table 11 gives the settings used to evaluate the ten LLMs under *blinded* × *open-research* from 2024-01-01 to 2026-04-10.

Table 11: Settings for the ten-model evaluation over the long trading window.

Group	Parameter	Value
Execution	benchmark	CSI300
	initial cash (CNY)	1,000,000
	buy cost (bps)	5
	sell cost (bps)	15
	min cost (CNY)	5
	deal price	next-day open
	T+1 settlement	yes
	long-only	yes
Agent	max candidates / step	100
	max tool calls / step	99
	max positions held	300
	max single weight	1.00
	schema retries	2
	retry with feedback	yes
	min reason length	10
	fallback action	hold
LLM call	temperature	0.0
	max tokens	32,768
	timeout (s)	600
	inter-call gap (s)	3

J Mask sensitivity on additional frontier LLMs

We compare *bright* and *blinded* under *open-research* for three additional LLMs. Each model is evaluated on ten windows covering different market conditions, with three seeds for each masking

setting and window. We first take the median across seeds within each window, then the median across windows (Table 12).

Table 12: Trading performance and behavior under *bright* and *blinded* for three additional LLMs. Ret. denotes total return, Abst. the fraction of steps with no orders, and Turn. annualized turnover. Values are aggregated across seeds and windows as described above.

Model	Mask	Ret. (%)	Abst.	Turn.
`gpt-5.5`	bright	−0.15	0.18	26.7
	blinded	−1.43	0.12	29.9
`doubao-seed-2.0`	bright	−2.97	0.03	31.3
	blinded	−4.07	0.00	33.7
`claude-opus-4-7`	bright	−0.40	0.20	21.4
	blinded	−1.44	0.02	73.5

K IDENTITY RECOVERY TEST DETAILS

Protocol. For each sampled stock and date, we collect outputs from `get_market_context`, `screen_candidates`, and `get_stock_snapshot` under *blinded*. These outputs contain the masked identifiers and numerical observations available to the trading agent. Each model is asked to infer the original ticker, date, sector, and trading board. We score its predictions against the corresponding true labels and calculate Wilson 95% confidence intervals for the recovery rates.

Results by model. Table 13 reports results for ten models given the same 200 randomly sampled stock/date cases. Each input contains the complete masked information available to the trading agent under *blinded*. Recovery rates are calculated over responses that return a parseable answer; failed or unparseable API responses are excluded from the denominator.

Top-1 ticker recovery counts a success when the model's first guess is correct. Its highest rate is 3.0%, compared with 0.3% for random guessing. Joint recovery requires both a correct ticker among the model's five guesses and a date within ± 7 trading days; its highest rate is 1.5%. The trading board remains visible because agents use it to apply price-limit rules. Board accuracy therefore checks whether models can report information already present in the input.

Table 13: Identity recovery rates from the masked inputs available under *blinded*. All models receive the same 200 stock/date cases; percentages are calculated over parseable responses. The columns tk@1 and tk@5 count correct tickers in the first guess and among five guesses, respectively. Date columns give recovery within the stated number of trading days. Joint recovery requires tk@5 and a date within ± 7 trading days. The trading board is visible in the input.

Model	tk@1	tk@5	board	date $\pm$7d	date $\pm$30d	date $\pm$90d	Joint
`step-3.5-flash`	0.0%	2.8%	100.0%	4.4%	13.3%	26.7%	0.0%
`claude-opus-4-7`	3.0%	10.2%	100.0%	1.0%	10.2%	22.8%	0.0%
`doubao-seed-2.0`	1.5%	9.0%	100.0%	7.0%	14.0%	23.5%	0.0%
`gpt-5.5`	2.0%	8.0%	99.5%	7.5%	13.0%	20.5%	0.5%
`qwen3.6-plus`	1.0%	4.6%	100.0%	6.2%	12.3%	28.7%	0.0%
`minimax-m2.7`	1.1%	4.9%	99.5%	3.3%	12.0%	21.7%	0.0%
`deepseek-v4-pro`	0.5%	5.5%	100.0%	10.5%	18.0%	34.0%	0.5%
`glm-5.1`	0.0%	6.4%	100.0%	16.0%	24.5%	45.7%	1.1%
`kimi-k2.6`	2.2%	9.8%	100.0%	1.6%	11.5%	29.0%	0.5%
`gemini-3.1-pro`	3.0%	8.6%	100.0%	14.7%	21.8%	36.5%	1.5%
random baseline	0.3%	1.4%	25.0%	1.9%	8.2%	24.7%	0.0%

L EVALUATION WINDOWS AND EXPANDED LEADERBOARD

Table 14 lists the ten short evaluation windows and the CSI300 return in each. Table 15 adds trading behavior and calibration metrics to the long-window leaderboard.

Table 14: Ten short evaluation windows covering different market conditions. †W10 falls after Step-3.5-Flash's knowledge cutoff.

Code	Dates	CSI300	Regime
W1	2024-04-01–06-05	-0.02%	near-flat
W2	2024-05-27–07-25	-6.50%	monotone bear
W3	2024-07-25–09-25	$+0.07\%$	near-flat
W4	2024-09-09–11-15	$+24.30\%$	explosive bull
W5	2024-10-21–12-13	-0.05%	near-flat
W6	2024-12-06–2025-02-14	-0.86%	mild decline
W7	2025-03-03–05-09	-1.09%	V-shape
W8	2025-07-14–09-19	$+12.05\%$	smooth uptrend
W9	2025-10-13–12-19	-0.56%	near-flat
W10†	2026-02-02–04-10	$+0.66\%$	mild rise

Table 15: Account performance, trading behavior, and calibration over the long window. Models are sorted by total return within each group, with CSI300 shown in grey at its position in the ranking. α-rk ranks LLMs by stock selection alpha under the full nine-factor model; bold marks the model with nonnegative alpha.

Model	Total ret.	Excess	Sharpe	MDD	IR	Turn.	HHI	ECE	α-rk
LLM agents, blinded × open-research, long window 2024-01-01 to 2026-04-10									
qwen3.6-plus	+85.29%	+48.37%	1.13	−24.49%	+0.48	31.88	0.005	0.367	2
gpt-5.5	+61.26%	+24.34%	0.98	−24.63%	+0.29	14.38	0.037	0.272	6
doubao-seed-2.0	+61.01%	+24.09%	1.15	−15.62%	+0.29	14.18	0.004	0.425	3
claude-opus-4-7	+58.80%	+21.88%	0.98	−26.69%	+0.26	31.75	0.012	0.246	**1**
minimax-m2.7	+56.31%	+19.39%	0.89	−29.43%	+0.25	43.68	0.029	0.303	4
CSI300 buy-and-hold	+36.92%	0.00%	—	—	—	0.00	—	—	—
step-3.5-flash	+21.86%	−15.06%	0.49	−28.99%	−0.13	24.66	0.012	0.357	5
gemini-3.1-pro	+8.25%	−28.67%	0.29	−49.67%	−0.11	101.23	0.066	0.387	7
deepseek-v4-pro	+1.39%	−35.53%	0.19	−42.35%	−0.29	73.49	0.026	0.352	8
glm-5.1	−3.55%	−40.47%	0.10	−44.88%	−0.40	100.25	0.032	0.322	9
kimi-k2.6	−24.23%	−61.15%	−0.34	−45.49%	−0.77	85.26	0.090	0.306	10
Trained Qlib factor models (Alpha9, 2008–2022 train, 2023 val); 8 representatives, full 18 in Appendix H									
SFM	+86.58%	+49.66%	2.02	−7.41%	+0.57	22.47	—	—	—
DoubleEnsemble	+73.66%	+36.74%	1.41	−18.24%	+0.41	72.84	—	—	—
CatBoost	+72.97%	+36.05%	1.43	−16.74%	+0.40	29.96	—	—	—
LightGBM	+66.22%	+29.30%	1.29	−15.59%	+0.33	46.18	—	—	—
Transformer	+60.35%	+23.43%	1.61	−11.29%	+0.27	11.22	—	—	—
LSTM	+43.92%	+7.00%	1.12	−11.76%	+0.06	78.03	—	—	—
CSI300 buy-and-hold	+36.92%	0.00%	—	—	—	0.00	—	—	—
MLP	+34.08%	−2.84%	0.69	−21.75%	−0.01	18.48	—	—	—
Linear	+29.60%	−7.32%	0.74	−13.88%	−0.12	15.61	—	—	—

M ADDITIONAL RESULTS ON W10

W10 covers 2026-02-02–2026-04-10 (43 trading days) and evaluates Step-3.5-Flash with five seeds. Table 16 reports its trading behavior, and Table 17 counts company, sector, and business references in its trading explanations. The account returns and selection alpha for the same runs are reported in Table 4, using medians across seeds for account returns and means for attribution.

Table 16: Trading behavior on W10 under *open-research*. Values are means across five seeds.

Metric	Bright	Blinded
Capital utilization	57.2%	65.6%
Annualized turnover	38.42	45.77
Abstention rate	7.9%	0.9%

Table 17: Trading steps on W10 whose final-action explanations contain explicit company names or company, sector, or business descriptions absent from the numerical inputs. Each masking setting has 215 steps across five seeds under *open-research*.

Mask	Detected steps	Rate
Bright	14/215	6.5%
Date-blind	9/215	4.2%
Stock-blind	0/215	0.0%
Blinded	0/215	0.0%

The same audit under *memory-only* finds 64/215 flagged steps (29.8%) when real stock tickers are visible, compared with 0/215 when tickers are masked. In this mode, agents receive identifiers and account information without price features.

N IDENTITY RECOVERY RESULTS

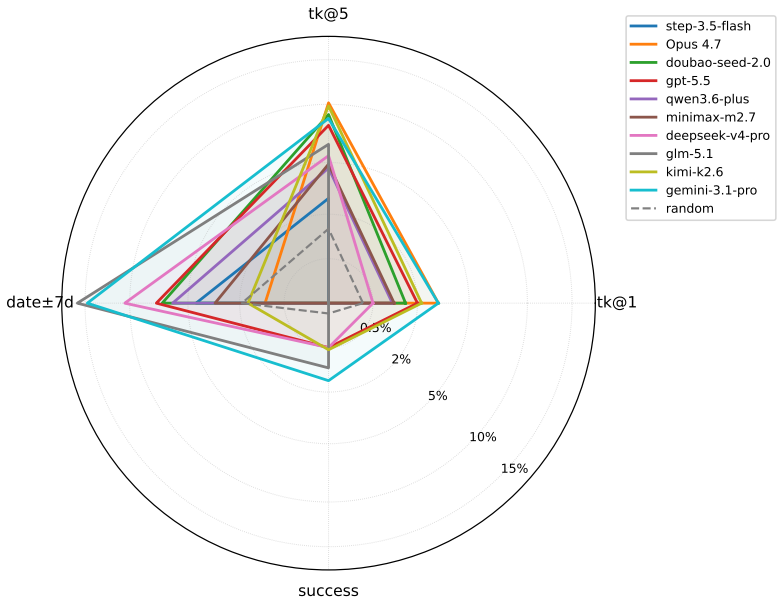

Figure 3: Identity recovery rates for each model under *blinded*, showing tk@1, tk@5, date recovery within ± 7 trading days, and joint recovery. Metrics are defined in Table 13. The radial axis uses a square-root scale. The dashed grey line shows the random baseline.

O FACTOR SENSITIVITY OF RETURN ATTRIBUTION

Table 18 summarizes the analysis in Section 4.7. All specifications use the same trading trajectories and averaging across seeds.

Table 18: Return attribution under different factor specifications on fixed LLM trading trajectories. Negative residual counts models with negative residual return contributions. ρ is the Spearman correlation with the full nine-factor ranking. The ranges summarize nine specifications, each retaining eight factors.

Specification	Negative residual	ρ
Full nine-factor model	9/10	1.000
Leave one factor out (nine specifications)	6/10–10/10	0.964–1.000
Common-only (no style factors)	4/10	0.939

We also fit a common-only model without style factors, termed market-only in the sensitivity analysis. Its residual combines returns associated with style exposure and stock selection. Four models have negative residuals under this specification, and the rank correlation with the full model is 0.939. Removing style factors therefore changes both the residual signs and the ranking, showing how style exposure affects the interpretation of agent returns.