Hierarchical Reinforced Trader: Text-Aware Financial Time-Series Portfolio Control

Zijie Zhao
zijiezha@mit.edu
Massachusetts Institute of Technology
Cambridge, MA, USA

Roy Welsch
rwelsch@mit.edu
Massachusetts Institute of Technology
Cambridge, MA, USA

Abstract

Financial portfolio control often relies on heterogeneous temporal inputs, including market histories, rolling risk measures, and timestamp-aligned text-derived sentiment and risk scores. These inputs must be converted into executable decisions under transaction costs, turnover, and downside-risk constraints. We propose Hierarchical Reinforced Trader (HRT), a bi-level reinforcement learning framework for text-aware financial time-series portfolio control. A factorized high-level controller selects sparse asset-level increase, reduce, or hold directions, while a low-level controller converts these directions into feasible portfolio weight adjustments under turnover, drawdown, and text-risk considerations. We evaluate HRT on an open stock-news benchmark with a fixed 89-stock Nasdaq universe, using 2013–2018 for training, 2019 for validation, and 2020–2023 for out-of-sample testing under a timestamp-clean protocol. Relative to HRT-Base, HRT improves Sharpe from 1.06 to 1.24, reduces daily turnover from 0.112 to 0.090, and is more robust under transaction-cost stress. The results suggest that separating directional selection from risk-aware execution helps convert market histories and time-aligned textual features into portfolio decisions that are more robust to trading costs.

CCS Concepts

• **Computing methodologies** → **Reinforcement learning**; • **Mathematics of computing** → *Time series analysis*; *Financial mathematics*.

Keywords

Financial Time Series, Hierarchical Reinforcement Learning, Portfolio Management, Text-Aware Trading Signals

1 Introduction

Financial markets generate heterogeneous temporal data: price-volume trajectories, rolling risk measures, cross-sectional information, and irregularly arriving news. A portfolio system must do more than forecast the next return from these inputs. It must convert time-aligned market and text-derived information into sequential decisions that remain feasible after turnover, drawdown, and transaction-cost constraints. We study this setting as daily multi-asset portfolio control with market and text-derived temporal inputs.

Recent time-series models have improved long-horizon and multivariate forecasting through architectural designs such as patching and channel independence in PatchTST [11], variate-centric attention in iTransformer [10], and larger model families for time-series and spatio-temporal data [5]. In portfolio management, however, a useful forecast is only an intermediate output. The decision layer must decide which assets to adjust, how large the adjustments should be, and whether the prediction remains useful after trading frictions. This distinction is especially important when market time series are combined with timestamp-aligned textual information.

Financial reinforcement learning provides a standard framework for sequential decision-making in markets. Systems such as FinRL and FinRL-Meta emphasize reproducible trading environments, fixed temporal splits, and realistic concerns such as leakage, overfitting, and market frictions [7–9]. Yet a single flat policy over a large stock universe can conflate several decisions that have different roles in portfolio management. It must infer directional views, size positions, control trading activity, and manage downside risk in one action. This is difficult when signals are noisy, heterogeneous, and only weakly predictive.

Text-derived inputs add another alignment challenge. Large financial news datasets such as FNSPID provide time-aligned news and stock-price records [2], and financial language models can extract sentiment or risk-related features from text [17]. We do not treat a language model as an autonomous trading agent. Instead, text enters the policy as structured stock-day sentiment and risk features aligned with the decision time. Portfolio decisions remain inside the trading policy, while textual risk information enters through these structured inputs.

We introduce Hierarchical Reinforced Trader (HRT), a bi-level framework that separates directional stock selection from portfolio execution. The High-Level Controller (HLC) uses a factorized categorical policy to choose sparse increase, reduce, or hold directions for each asset without enumerating the full 3^N joint direction space. The Low-Level Controller (LLC) receives these directions and the current portfolio state, then determines risk-aware weight adjustments under cost, turnover, drawdown, and text-risk considerations. The hierarchy corresponds to a practical portfolio-management distinction: the HLC forms sparse directional views, while the LLC decides how aggressively to act on them.

We evaluate HRT on an open stock-news benchmark constructed from daily market data and timestamp-aligned news-derived signals. The model is trained on 2013–2018, validated on 2019, and tested once on 2020–2023, covering the COVID shock and recovery, a bull market, a Fed-tightening drawdown, and a technology-led recovery. The experiments compare HRT with an external market proxy, same-universe portfolio baselines, alpha-only selection, flat RL, and hierarchical ablations. Relative to HRT-Base, HRT improves Sharpe from 1.06 to 1.24 and reduces daily turnover from 0.112 to 0.090, while not claiming to dominate every individual risk metric.

Our contributions are:

- We formulate text-aware financial time-series portfolio control as a bi-level sequential decision problem that separates sparse directional selection from risk-aware execution.
- We introduce a factorized sparse High-Level Controller and a risk-aware Low-Level Controller that convert compact market and text-derived temporal signals into feasible portfolio adjustments.
- We evaluate HRT under a fixed public temporal protocol with same-universe baselines, ablations, transaction-cost stress tests, and market-regime analysis, showing higher Sharpe and lower turnover relative to the hierarchical base model under the fixed benchmark protocol.

2 Related Work

2.1 Heterogeneous Financial Time-Series Learning

Modern time-series learning has moved beyond classical forecasting models toward scalable architectures for long-horizon and multivariate settings. PatchTST represents each univariate channel with subseries-level patches and shares parameters across channels [11]. iTransformer inverts the usual tokenization scheme by embedding individual series into variate tokens, improving multivariate forecasting and interpretability of attention across variates [10]. Recent surveys further connect large models with time-series and spatio-temporal analysis [5]. These works focus primarily on temporal representation and prediction. Our focus is different: HRT uses compact market and text-derived temporal features as inputs and studies the constrained decision layer that turns them into portfolio actions.

Financial time-series learning often requires integrating numerical market data with textual information. FNSPID provides a large-scale dataset of time-aligned financial news and stock-price records, enabling open evaluation of models that combine market and news information [2]. Our experiments use a processed stock-news benchmark built on this line of work, where text-derived sentiment and risk scores are aligned with stock-day records. The main question is not whether text improves a standalone forecasting task, but whether such signals can be used by a constrained decision layer without leakage.

2.2 Reinforcement Learning for Financial Sequential Decision-Making

Reinforcement learning is widely used in trading because portfolio management is naturally sequential and path-dependent. FinRL provides a framework for automated trading experiments, while FinRL-Meta and related market-environment work emphasize reproducibility, benchmark design, and the risks of overfitting or leakage in financial RL [7–9]. In our implementation, PPO is used for the discrete high-level direction policy [14], and TD3 is used for the continuous low-level execution controller [3]. The market forecast interface uses LightGBM [6] and Qlib-style factors [16]. Unlike a flat trading policy that maps states directly to portfolio actions, HRT assigns directional selection and portfolio sizing to separate controllers.

2.3 Hierarchical and Text-Aware Portfolio Control

Hierarchical reinforcement learning decomposes long-horizon decision problems into higher-level and lower-level controllers [12]. In financial applications, hierarchy has been used to separate allocation from execution [15], to improve high-frequency trading efficiency [13], and to structure select-and-trade decisions [4]. HRT differs by focusing on daily multi-stock equity portfolio control with timestamp-aligned market and text-derived signals, a factorized sparse direction policy, and a risk-aware execution layer.

Text-aware trading systems incorporate news, sentiment, or language-model outputs into financial decision-making. FinGPT illustrates how instruction-tuned language models can extract financial sentiment signals [17], and FinRL-DeepSeek explores risk-sensitive reinforcement learning with language-model-derived signals [1]. HRT uses text in a constrained role: benchmark-provided text-derived sentiment and risk scores become structured temporal inputs, while the RL hierarchy remains responsible for the trading decision. Our contribution is not a new PPO or TD3 estimator, but the portfolio-control decomposition: a factorized sparse directional policy, a deterministic feasibility layer, and a risk-aware execution reward that convert timestamp-aligned market and text-derived signals into executable weight updates.

3 Methodology

3.1 Overview: Financial Time-Series Portfolio Control

The Hierarchical Reinforced Trader (HRT) is a risk-aware hierarchical reinforcement learning framework for text-aware financial time-series portfolio control. As shown in Figure 1, HRT consists of two components with distinct responsibilities. The HLC forms sparse directional views by selecting whether each asset should be increased, reduced, or held. The LLC then translates these directions into portfolio weight adjustments subject to trading and risk constraints. This decomposition reflects a practical distinction in portfolio management: alpha signals form directional views, whereas execution must also account for costs, turnover, current holdings, drawdown, and risk exposure.

Compared with a flat trading policy that directly maps a high-dimensional state to trading quantities for all assets, HRT makes three choices. First, the HLC uses a factorized sparse policy, which parameterizes stock-level directional decisions without enumerating the full joint action space. Second, the policy consumes compact market and text-derived features rather than raw price-volume sequences or raw text. Third, the LLC performs risk-aware sizing: it follows the HLC direction mask while penalizing excessive turnover, drawdown, and exposure to text-derived risk.

3.2 Heterogeneous Temporal Signal Interface

HRT is agnostic to the specific upstream forecaster or language model. The policy does not operate on raw time series or raw text; it receives compact stock-day states built from market time-series features and timestamp-aligned text-derived signals. For asset i on day t, the HLC receives a compact state

$$s^h_{i,t} = [\bar{w}_{i,t},\ \hat{r}_{i,t},\ \hat{\sigma}_{i,t},\ u_{i,t},\ \rho_{i,t}], \tag{1}$$

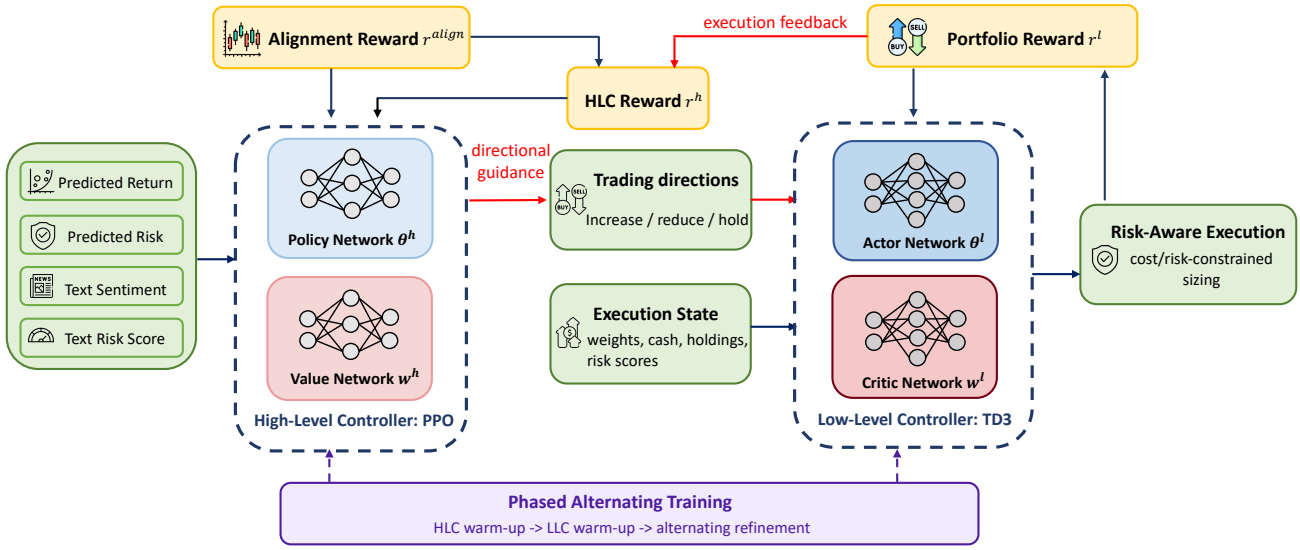

Figure 1: Overview of the Hierarchical Reinforced Trader (HRT). The HLC selects sparse trading directions from heterogeneous temporal signals, while the LLC converts these directions into cost- and risk-aware portfolio adjustments. Red arrows indicate the two key interactions between the controllers: directional guidance from the HLC to the LLC and execution feedback from the LLC to the HLC.

where $\tilde{w}_{i,t}$ is the current pre-trade portfolio weight, $\hat{r}_{i,t}$ is a predicted forward return, $\hat{\sigma}_{i,t}$ is a risk proxy, $u_{i,t}$ is a structured textual sentiment score, and $\rho_{i,t}$ is a structured textual risk score. This state is a compact temporal representation at the stock-day level rather than a static tabular feature vector. Including the current holding makes the reduce action meaningful in a long-only portfolio: reducing a zero-weight asset is mapped to no trade by the execution layer. We use asset-level notation when defining the HLC input; vector quantities such as $\hat{r}_t$ and ρ_t later denote the corresponding signals across the tradable universe.

In our experiments, the market return forecast is produced by a LightGBM model [6] trained on OHLCV-derived technical indicators and Qlib-style factors [16] to predict the next tradable-period return. The risk proxy $\hat{\sigma}_{i,t}$ is computed as the 20-day rolling realized volatility using only past returns. For textual inputs, we use the processed benchmark's precomputed text-derived sentiment and risk signals, which are derived from timestamp-aligned financial news rather than generated online during RL training. These signals are already aligned with stock-day records in the benchmark; before RL training, we normalize them using training-period statistics and cache the resulting policy inputs. The text-derived variables are used as fixed benchmark covariates rather than generated by HRT. HRT does not use validation/test statistics, future news, LLM fine-tuning, or online LLM queries; stock-days without eligible pre-cutoff text are not filled from future news. Thus, HRT's text-aware contribution is in how the trading policy uses these signals, not in new text encoding or sentiment modeling.

For each trading decision, the policy uses benchmark-provided market and text-derived signals available before the close-of-day decision cutoff on day t. The portfolio is rebalanced at the next tradable open, and realized returns are used only as ex-post rewards or evaluation targets. When necessary, after-cutoff or non-trading-day records are rolled forward to the next tradable decision date. Thus, text-derived signals are used as decision-time inputs rather than ex-post labels.

3.3 Factorized Sparse High-Level Controller

The HLC selects stock-level trading directions. For asset i, the high-level action is $a^h_{i,t} \in \{-1, 0, 1\}$, where 1 denotes increasing the position, -1 denotes reducing the position, and 0 denotes holding it unchanged. Rather than treating the HLC decision as an unstructured joint action over 3^N possible direction vectors, we use a factorized categorical policy

$$\pi^h_\theta(a^h_t \mid s^h_t) = \prod_{i=1}^{N} \pi^h_\theta(a^h_{i,t} \mid s^h_{i,t}). \tag{2}$$

The policy is implemented with shared network parameters and asset-level directional logits. Thus, the HLC evaluates $O(N)$ categorical distributions and produces $3N$ logits per day, rather than enumerating a joint 3^N action space. Sparse activation further reduces the LLC sizing problem from all N assets to the active set of assets with nonzero executed adjustments.

The HLC is trained with PPO [14]. Sparsity is induced through the hold action, a validation-selected confidence filter, and an activation penalty. Its reward combines directional alignment with downstream execution feedback:

$$r_t^h = \alpha_t \tilde{r}_t^{align} + (1-\alpha_t)\tilde{r}_t^l - \lambda_{act}\frac{|\mathcal{A}_t^{exec}|}{N}. \tag{3}$$

Here $\mathcal{A}_t^{exec}$ is the set of assets with nonzero executed weight changes, and the tildes denote rewards normalized using training-period running statistics. The alignment component is computed only over executed changes; if the HLC produces an all-hold decision or a proposed reduce action is infeasible because the current holding is zero, the corresponding alignment contribution is zero. Normalizing both the alignment term and LLC feedback before mixing gives $\alpha_t = \alpha_0 \exp(-\lambda t)$ a stable interpretation as training shifts from directional correctness toward portfolio-level outcomes. Next-day realized returns are used only to compute ex-post training rewards and evaluation metrics. They are never included in the HLC or LLC state observed at decision time.

3.4 Risk-Aware Low-Level Controller

Given the HLC directions, the LLC determines the portfolio adjustment. We formulate the LLC action in weight space rather than raw share counts, because portfolio weights make risk and turnover constraints explicit. Let $\bar{w}_t$ denote the current pre-trade portfolio weights, V_t the portfolio value, and $d_t = (V_t^{peak} - V_t)/V_t^{peak}$ the current drawdown state. The LLC observes

$$s_t^l = [\bar{w}_t,\ b_t/V_t,\ d_t,\ a_t^h,\ \hat{r}_t,\ \hat{\sigma}_t,\ \rho_t], \tag{4}$$

where b_t is cash and ρ_t is the vector of normalized text-derived risk scores.

The TD3 actor first proposes a raw weight adjustment. Before execution, HRT applies a deterministic feasibility layer:

$$\begin{aligned} \widetilde{\Delta w}_t &= \mu_\phi(s_t^l), \\ \Delta w_t^{exec} &= \mathcal{F}(\widetilde{\Delta w}_t, a_t^h, \bar{w}_t), \\ w_t &= \bar{w}_t + \Delta w_t^{exec}. \end{aligned} \tag{5}$$

The feasibility layer is intentionally simple. It first clips each adjustment to match the HLC direction: increase permits only nonnegative changes, reduce permits only reductions down to zero, and hold sets the adjustment to zero. It then clips weights to the single-name limit and scales only the active positive adjustments when buy demand exceeds available cash or the daily turnover budget, so that hold assets remain unchanged and the HLC direction constraints are preserved. The executed feasible adjustment, rather than the raw actor output, is stored in the replay buffer and used by the TD3 critic. During actor updates, candidate actions are passed through the same feasibility layer before critic evaluation, i.e., the critic evaluates $\mathcal{F}(\mu_\phi(s_t^l), a_t^h, \bar{w}_t)$ rather than the unconstrained raw adjustment. This keeps the learned controller aligned with the action actually executable in the trading environment while avoiding an additional optimization layer.

The LLC reward optimizes net, risk-aware execution:

$$r_t^l = R_t^{net} - \lambda_{turn}\text{Turnover}_t - \lambda_{dd} d_{t+1} - \lambda_{risk}\sum_i w_{i,t}\rho_{i,t}. \tag{6}$$

Here $R_t^{net} = (V_{t+1} - V_t)/V_t$ is the portfolio return after transaction costs have been deducted by the environment, d_{t+1} is the post-trade drawdown, and the final term penalizes exposure to assets with high text-derived risk. This reward does not assume that lower text-risk exposure always improves raw return; instead, it encourages the execution layer to trade off return, cost, turnover, and downside control.

We implement the LLC with a TD3-style actor–critic controller [3]. TD3 is suitable for this continuous-control execution problem because clipped double-Q learning and delayed policy updates reduce overestimation in deterministic actor–critic training. Baseline variants are described in Section 4.

3.5 Phased Alternating Training

The two controllers are trained with a phased alternating procedure. The schedule stabilizes the hierarchy: the HLC first learns usable directional decisions, and the LLC then learns how to size these decisions under portfolio constraints.

Algorithm 1 Training Procedure for HRT

1: Prepare market forecasts $(\hat{r}, \hat{\sigma})$ from benchmark market data and load benchmark-provided text sentiment and risk signals (u, ρ) on the training period.
2: Normalize text-risk scores using training-period statistics.
3: Initialize the HLC policy π_θ^h and the LLC actor–critic networks.
4: **Phase 1: HLC warm-up.** Train π_θ^h using PPO with normalized directional-alignment rewards; LLC feedback is not used in this phase.
5: **Phase 2: LLC warm-up.** Freeze π_θ^h and train the LLC to execute the HLC directions using the risk-aware reward in Eq. (6).
6: **Phase 3: Alternating refinement.** Alternately update HLC and LLC. In the HLC reward, decay α_t so that LLC feedback gradually receives more weight. After HLC updates, new trajectories are collected under the updated hierarchy.
7: Select sparse thresholds, risk penalties, and checkpoints using validation-period risk-adjusted performance.

During alternating refinement, LLC transitions store the HLC action that produced the execution decision. The replay buffer stores the feasible portfolio adjustment actually executed by the environment, rather than the unconstrained actor output, because rewards are generated after enforcing trading constraints. All model selection decisions, including sparse confidence thresholds, risk penalty coefficients, and checkpoints, are made using the 2019 validation period only. The 2020–2023 test period is used once for final out-of-sample evaluation.

3.6 Computational Considerations

Because it acts on stock-day states, the factorized HLC requires $O(N)$ directional evaluations per day rather than enumerating a

Table 1: Benchmark protocol. QQQ is used only as an external Nasdaq-100 market proxy; all portfolio baselines and HRT variants are evaluated on the same tradable stock universe.

Item	Setting
Universe	Fixed 89-stock Nasdaq universe
Train / Val / Test	2013–2018 / 2019 / 2020–2023
Market signals	OHLCV factors, technical indicators, $\hat{r}$, realized-volatility proxy
Text signals	Timestamp-aligned stock-day sentiment and risk scores
Trading protocol	Features/news through close t; after-cutoff news rolled forward
Execution	Daily weight-based rebalance at next tradable open ($t+1$)
Base cost	10 bps per traded notional
Market reference	QQQ, external Nasdaq-100 proxy only
Seeds	5 seeds for stochastic learning-based methods
Model selection	Validation-only selection on 2019

joint 3^N action space. Sparse activation further reduces the number of assets passed to the LLC from N to $|\mathcal{A}_t^{exec}|$. Market forecasts are computed offline before trading decisions, and benchmark-provided text features are loaded as structured stock-day inputs. The online policy consumes only compact forecasts, sentiment scores, and risk scores. This keeps the training cost comparable to standard financial RL baselines while making the method suitable for larger open equity universes.

4 Experiments

The experiments evaluate whether HRT turns market and text-derived temporal inputs into better executable portfolio decisions. The evaluation is organized around four questions: whether the hierarchy improves out-of-sample performance, which components account for the gains, whether the learned policies are robust to transaction-cost stress, and whether the behavior is consistent across market regimes. Unless otherwise stated, deterministic baselines are computed once, and stochastic learning-based methods are averaged over five random seeds.

4.1 Benchmark and Temporal Evaluation Protocol

We evaluate HRT on the open FinRL-DeepSeek stock-trading benchmark, a FinRL-compatible benchmark constructed from FNSPID financial news and daily market data with precomputed text-derived sentiment and risk signals [1, 2, 9]. The tradable universe is the filtered 89-stock Nasdaq universe released with the benchmark. We follow this fixed universe for reproducibility; it should not be interpreted as a point-in-time dynamic tradable universe, and survivorship effects may not be fully removed. The model is trained on 2013–2018, validated on 2019, and tested on 2020–2023. We restrict the main evaluation to 2020–2023 because extending the same text-aware protocol beyond the public benchmark horizon would require constructing an additional timestamp-clean news-signal cache.

For each rebalancing date, we use the benchmark-provided time-aligned market, news, and text-derived signals available before the close-of-day decision cutoff on day t. The portfolio is rebalanced at the next tradable open, and P&L is measured until the next scheduled rebalance. News or signal updates released after the cutoff, including after-hours and non-trading-day records, are assigned to the next tradable decision date. This protocol is applied consistently to training, validation, and testing. The text-derived sentiment score $u_{i,t}$ and risk score $\rho_{i,t}$ are benchmark-provided stock-day signals that are normalized using training-period statistics and cached before RL training.

Baselines. We compare HRT with three groups of baselines. First, we report QQQ as an external Nasdaq-100 market proxy to contextualize the market environment. Second, we compute same-universe portfolio baselines. Equal Weight rebalances the benchmark universe uniformly. Minimum Variance uses a rolling covariance estimate with long-only and single-name weight constraints. Momentum Top-K ranks assets by past 20-day return and holds the top-K names with equal weights. Alpha Top-K ranks assets by the same predicted forward return $\hat{r}_{i,t}$ used by HRT and holds the top-K names without reinforcement learning. Third, we compare with Flat TD3 and HRT variants.

All same-universe portfolio strategies use the same 89-stock universe, base transaction cost, long-only budget convention, and weight-based execution protocol. QQQ is not part of the same-universe fairness comparison; it is included only as an external market reference. The values of K, the covariance lookback, and execution hyperparameters are listed in Appendix A.

HRT variants. We use the following variants to isolate the contribution of each component. **HRT-Base** is a hierarchical PPO–TD3 model using market forecasts and sentiment, without the sparse confidence filter or activation penalty, without text-risk scores, and with a return-only LLC reward; its categorical HLC can still output hold decisions. **+ Sparse HLC** adds sparse high-level activation and the activation penalty. **+ Text Risk Signal** adds text-derived risk scores to the controller states. **+ Risk-Aware LLC (HRT)** adds turnover, drawdown, and text-risk exposure penalties to the LLC reward. **HRT w/o Sparse HLC** removes sparse high-level activation from the full HRT model.

Metrics and compute. Cumulative return is computed as $\prod_t(1+r_t)-1$. We report annualized return as the compound annual growth rate over the test period, $(1+\text{Cum. Ret.})^{1/T}-1$. Annualized volatility is $\sqrt{252}\sigma_d$, and Sharpe is the standard daily-return Sharpe, $\sqrt{252}(\bar{r}_d - r_{f,d})/\sigma_d$ with $r_f = 0$. Daily turnover is traded notional divided by portfolio value, $\sum_i |\Delta w_{i,t}^{exec}|$; transaction cost is turnover multiplied by the per-notional cost rate, so no one-half turnover convention is used. We also report maximum drawdown and 5% CVaR. HLC is trained with PPO and LLC with TD3. Both controllers use two-layer MLPs with 256 hidden units and 5×10^5 training steps. We use two NVIDIA A100 GPUs only for parallelizing seeds and ablations; each training run fits on a single GPU and takes approximately 3–6 hours. All text-derived signals are precomputed and cached, and no LLM fine-tuning or online LLM querying is performed during RL training. Full hyperparameters are listed in Appendix A.

4.2 Overall Out-of-Sample Performance

This section addresses whether the proposed hierarchy improves out-of-sample portfolio control. Figure 2 shows cumulative returns over the 2020–2023 test period. HRT participates in the 2020–2021 and 2023 upward regimes, while reducing downside during the

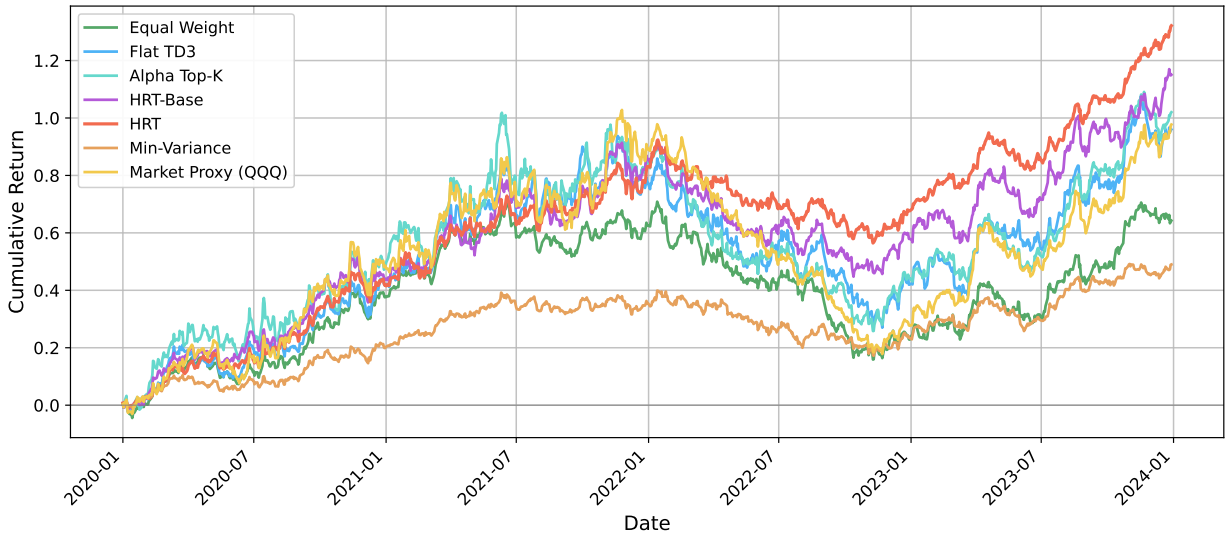

Figure 2: Cumulative returns over the 2020–2023 test period. Market Proxy denotes QQQ, an external Nasdaq-100 proxy. HRT does not dominate every year, but it reduces downside during the 2022 drawdown and finishes above the other reported learning-based strategies on risk-adjusted performance.

2022 drawdown. Table 2 reports the corresponding return, risk, and execution metrics.

Compared with HRT-Base, HRT improves the standard daily-return Sharpe from 1.06 to 1.24, reduces daily turnover from 0.112 to 0.090, and improves maximum drawdown from -0.305 to -0.245. Compared with Alpha Top-K, HRT obtains higher cumulative return with lower drawdown and substantially lower turnover, suggesting that the hierarchy adds value beyond the supervised alpha signal by translating it into more cost- and risk-controlled portfolio actions. HRT does not dominate every individual risk metric: Min-Variance has lower maximum drawdown and CVaR, and HRT-Base has slightly better one-day CVaR. The result therefore supports HRT as a risk- and cost-aware learning-based strategy, rather than as a pure tail-risk minimization method.

4.3 Component Ablation

Table 3 isolates the effect of the main design choices and links the empirical gains to the design choices in Section 3. The first four rows add components incrementally, moving from the hierarchical base model to the full HRT system. The final row removes sparsity from the full HRT model as a diagnostic check: it obtains higher raw cumulative return, but at the cost of higher turnover and drawdown.

The ablation suggests that different components improve different parts of the trading pipeline. Sparse high-level activation sacrifices some raw return but tends to improve Sharpe, drawdown, and turnover. Adding text-derived risk information reduces ex-post text-risk exposure and improves path-level downside control. The full risk-aware LLC further improves the return–risk–cost balance by combining higher Sharpe, lower turnover, lower text-risk exposure, and smaller maximum drawdown. The unconstrained variant confirms that higher raw return can be obtained by trading more aggressively, but this comes with substantially worse drawdown and turnover; therefore the full HRT is selected for risk-adjusted and cost-aware performance rather than maximum raw return or pure tail-risk minimization.

4.4 Transaction-Cost Robustness

A central motivation for the LLC is that directional signals should not be converted into excessive turnover. We therefore evaluate out-of-sample cost sensitivity by applying the policies selected under the 10 bps validation setting to alternative transaction-cost assumptions without retraining. Figure 3 reports standard daily-return Sharpe ratios under costs from 1 to 50 bps.

At the base 10 bps setting, HRT achieves a standard daily-return Sharpe ratio of 1.24. As transaction costs rise, all strategies lose risk-adjusted performance, but the decline is substantially sharper for the higher-turnover Alpha Top-K and Flat TD3 baselines. HRT-Base remains more resilient than the higher-turnover baselines, and the full HRT has the most stable Sharpe profile among the reported strategies. At the 50 bps setting, HRT retains a positive Sharpe ratio of 0.80, compared with 0.57 for HRT-Base and near-zero values for Alpha Top-K and Flat TD3. This pattern supports the role of sparse high-level activation and risk-aware execution in improving cost robustness by limiting unnecessary trading activity. Additional trading behavior statistics, including active names, annualized cost drag, concentration, and text-risk exposure, are reported in Appendix B.

4.5 Market-Regime Robustness

Aggregate test-period metrics can obscure whether a policy behaves reasonably under different temporal regimes. Table 4 therefore summarizes yearly returns and maximum drawdowns for the market proxy, HRT-Base, and HRT. HRT is not designed to beat the external market proxy in every bull-market year. Its main advantage appears

Table 2: Main out-of-sample performance over 2020–2023. Learning-based methods report mean ± seed standard deviation over five seeds. HRT is not best on every individual risk metric; the table reports return, risk, and execution outcomes under the fixed temporal protocol.

Model	Cum. Ret.	Ann. Ret.	Ann. Vol.	Sharpe	Max DD	CVaR 5%	Turnover
External market proxy							
Market Proxy (QQQ)	0.977	0.186	0.258	0.80	-0.356	-0.031	–
Same-universe non-RL baselines							
Equal Weight	0.643	0.132	0.215	0.70	-0.292	-0.027	0.015
Min-Variance	0.490	0.105	0.155	0.74	-0.205	-0.020	0.045
Momentum Top-K	0.920	0.177	0.295	0.72	-0.410	-0.037	0.160
Alpha Top-K	1.020	0.192	0.285	0.82	-0.340	-0.033	0.210
Learning-based strategies							
Flat TD3	0.960 ± 0.110	0.183 ± 0.017	0.255 ± 0.018	0.83 ± 0.11	−0.315 ± 0.040	−0.030 ± 0.003	0.195 ± 0.025
HRT-Base	1.151 ± 0.080	0.211 ± 0.011	0.220 ± 0.010	1.06 ± 0.07	−0.305 ± 0.025	−0.027 ± 0.002	0.112 ± 0.010
HRT	1.321 ± 0.060	0.234 ± 0.009	0.208 ± 0.010	1.24 ± 0.06	−0.245 ± 0.015	−0.028 ± 0.003	0.090 ± 0.007

Table 3: Ablation study. The first four rows add components incrementally. The last row removes sparse activation from the full model to show the cost of unconstrained trading activity. Values are mean ± seed standard deviation over five seeds. Text-risk exposure is computed ex post for all variants using the same normalized risk scores.

Variant	Cum. Ret.	Sharpe	Max DD	Turnover	Text-Risk Exp.
HRT-Base	1.151 ± 0.080	1.06 ± 0.07	−0.305 ± 0.025	0.112 ± 0.010	0.43 ± 0.02
+ Sparse HLC	1.120 ± 0.075	1.10 ± 0.06	−0.270 ± 0.020	0.101 ± 0.008	0.42 ± 0.02
+ Text Risk Signal	1.205 ± 0.065	1.17 ± 0.06	−0.255 ± 0.018	0.098 ± 0.008	0.39 ± 0.02
+ Risk-Aware LLC (HRT)	1.321 ± 0.060	1.24 ± 0.06	−0.245 ± 0.015	0.090 ± 0.007	0.35 ± 0.02
HRT w/o Sparse HLC	1.365 ± 0.095	1.12 ± 0.08	−0.335 ± 0.035	0.150 ± 0.018	0.38 ± 0.02

Table 4: Yearly robustness over the 2020–2023 test period. HRT is not intended to dominate the market proxy in every bull-market year; its advantage is most visible in downside control during adverse regimes.

Year	Regime	Market Ret.	HRT-Base Ret.	HRT Ret.	Base Max DD	HRT Max DD
2020	COVID/recovery	0.486	0.440	0.420	-0.285	-0.245
2021	Bull market	0.274	0.290	0.290	-0.110	-0.100
2022	Fed tightening	-0.326	-0.150	-0.085	-0.305	-0.235
2023	Tech-led recovery	0.549	0.362	0.385	-0.130	-0.115

in adverse regimes: during the 2022 Fed-tightening drawdown, HRT has a less negative return and smaller maximum drawdown than HRT-Base, while still participating in the 2023 technology-led recovery. This pattern is consistent with the role of the sparse high-level controller and the risk-aware execution layer: the hierarchy improves the way temporal signals are converted into constrained portfolio decisions, rather than simply increasing exposure in all regimes.

5 Discussion and Conclusion

This paper presents HRT, a bi-level reinforcement learning framework for text-aware financial time-series portfolio control. HRT separates the trading decision into a sparse high-level direction policy that decides what to trade and a risk-aware low-level execution controller that decides how aggressively to trade. This decomposition makes portfolio control more selective and execution-aware, rather than optimizing raw return without regard to cost or risk.

The empirical results support this design. On a fixed open stock-news benchmark, HRT improves risk-adjusted performance and turnover relative to HRT-Base and Flat TD3. The ablation study shows that sparse high-level activation, text-risk signals, and risk-aware execution affect different parts of the trading pipeline. Cost-sensitivity results show that HRT degrades more slowly than higher-turnover baselines, and yearly results show that its benefit is most visible during the 2022 drawdown while it still participates in the 2023 recovery.

The framework has limitations. The evaluation uses the filtered 89-stock universe and 2020–2023 horizon supported by the public benchmark; this fixed-universe protocol evaluates the proposed execution mechanism rather than claiming production-level live-trading performance. The backtest uses weight-based fractional

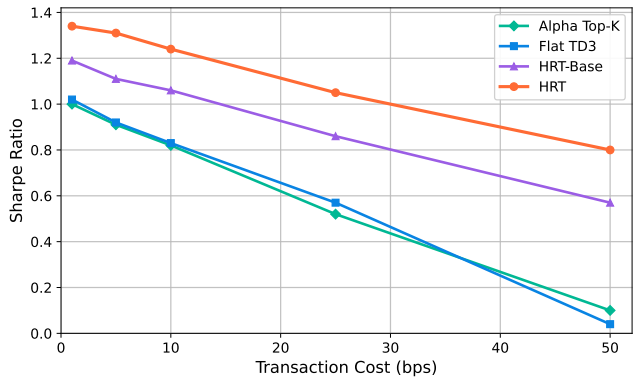

Figure 3: Transaction-cost sensitivity. Policies selected under the 10 bps validation setting are replayed under alternative cost assumptions without retraining. All strategies deteriorate as transaction costs increase. The higher-turnover Alpha Top-K and Flat TD3 baselines degrade more sharply, while HRT and HRT-Base are more resilient due to hierarchical selection and lower trading activity.

execution without integer-share rounding, market impact, or capacity constraints. HRT is signal-agnostic, but its performance still depends on the quality, calibration, and timestamp alignment of upstream market and text-derived signals. The reported seed standard deviations quantify stochastic training variability, but they should not be interpreted as confidence intervals over future market regimes. Stronger statistical evidence would require paired block-bootstrap or related return-series tests, together with evaluation on additional point-in-time universes and alternative text-signal sources. The factorized HLC is also a scalable approximation and does not fully model all cross-asset dependencies.

Taken together, the results suggest that market and text-derived information should not be passed directly to an unconstrained trading policy. Separating sparse directional views from risk-aware execution gives the policy a clearer structure, preserves a transparent input interface, and keeps final portfolio changes tied to turnover, drawdown, and feasibility constraints. By keeping the input interface explicit and the final weight updates constrained, HRT links time-aligned market and text information to executable portfolio decisions without turning the trading policy into an unconstrained black box.

References

[1] Mostapha Benhenda. 2025. FinRL-DeepSeek: LLM-Infused Risk-Sensitive Reinforcement Learning for Trading Agents. arXiv preprint arXiv:2502.07393.

[2] Zihan Dong, Xinyu Fan, and Zhiyuan Peng. 2024. FNSPID: A Comprehensive Financial News Dataset in Time Series. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. Association for Computing Machinery, New York, NY, USA. doi:10.1145/3637528.3671629

[3] Scott Fujimoto, Herke van Hoof, and David Meger. 2018. Addressing Function Approximation Error in Actor-Critic Methods. In *Proceedings of the 35th International Conference on Machine Learning*. PMLR, 1587–1596.

[4] Weiguang Han, Boyi Zhang, Qianqian Xie, Min Peng, Yanzhao Lai, and Jimin Huang. 2023. Select and Trade: Towards Unified Pair Trading with Hierarchical Reinforcement Learning. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. Association for Computing Machinery, New York, NY, USA, 4123–4134.

[5] Ming Jin, Qingsong Wen, Yuxuan Liang, Chaoli Zhang, Siqiao Xue, Xue Wang, James Zhang, Yi Wang, Haifeng Chen, Xiaoli Li, Shirui Pan, Vincent S. Tseng, Yu Zheng, Lei Chen, and Hui Xiong. 2023. Large Models for Time Series and Spatio-Temporal Data: A Survey and Outlook. arXiv preprint arXiv:2310.10196.

[6] Guolin Ke, Qi Meng, Thomas Finley, Taifeng Wang, Wei Chen, Weidong Ma, Qiwei Ye, and Tie-Yan Liu. 2017. LightGBM: A Highly Efficient Gradient Boosting Decision Tree. In *Advances in Neural Information Processing Systems*, Vol. 30. Curran Associates, Inc., 3146–3154.

[7] Xiao-Yang Liu, Ziyi Xia, Jingyang Rui, Jiechao Gao, Hongyang Yang, Ming Zhu, Christina Dan Wang, Zhaoran Wang, and Jian Guo. 2022. FinRL-Meta: Market Environments and Benchmarks for Data-Driven Financial Reinforcement Learning. In *Advances in Neural Information Processing Systems*, Vol. 35. Curran Associates, Inc.

[8] Xiao-Yang Liu, Ziyi Xia, Hongyang Yang, Jiechao Gao, Daochen Zha, Ming Zhu, Christina Dan Wang, Zhaoran Wang, and Jian Guo. 2024. Dynamic Datasets and Market Environments for Financial Reinforcement Learning. *Machine Learning* 113, 5 (2024), 2795–2839. doi:10.1007/s10994-023-06511-w

[9] Xiao-Yang Liu, Hongyang Yang, Jiechao Gao, and Christina Dan Wang. 2021. FinRL: Deep Reinforcement Learning Framework to Automate Trading in Quantitative Finance. In *Proceedings of the Second ACM International Conference on AI in Finance*. Association for Computing Machinery, New York, NY, USA, 1–9.

[10] Yong Liu, Tengge Hu, Haoran Zhang, Haixu Wu, Shiyu Wang, Lintao Ma, and Mingsheng Long. 2024. iTransformer: Inverted Transformers Are Effective for Time Series Forecasting. In *International Conference on Learning Representations*.

[11] Yuqi Nie, Nam H. Nguyen, Phanwadee Sinthong, and Jayant Kalagnanam. 2023. A Time Series is Worth 64 Words: Long-Term Forecasting with Transformers. In *International Conference on Learning Representations*.

[12] Shubham Pateria, Budhitama Subagdja, Ah-Hwee Tan, and Chai Quek. 2021. Hierarchical Reinforcement Learning: A Comprehensive Survey. *Comput. Surveys* 54, 5 (2021), 1–35.

[13] Molei Qin, Shuo Sun, Wentao Zhang, Haochong Xia, Xinrun Wang, and Bo An. 2024. EarnHFT: Efficient Hierarchical Reinforcement Learning for High Frequency Trading. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 38. AAAI Press, 14669–14676.

[14] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. 2017. Proximal Policy Optimization Algorithms. arXiv preprint arXiv:1707.06347.

[15] Rundong Wang, Hongxin Wei, Bo An, Zhouyan Feng, and Jun Yao. 2020. Deep Stock Trading: A Hierarchical Reinforcement Learning Framework for Portfolio Optimization and Order Execution. arXiv preprint arXiv:2012.12620.

[16] Xiao Yang, Weiqing Liu, Dong Zhou, Jiang Bian, and Tie-Yan Liu. 2020. Qlib: An AI-Oriented Quantitative Investment Platform. arXiv preprint arXiv:2009.11189.

[17] Boyu Zhang, Hongyang Yang, and Xiao-Yang Liu. 2023. Instruct-FinGPT: Financial Sentiment Analysis by Instruction Tuning of General-Purpose Large Language Models. arXiv preprint arXiv:2306.12659.

A Implementation and Hyperparameters

The feasibility layer in Eq. (5) is implemented as deterministic clipping and scaling. Hold assets are assigned zero adjustment. Reduce assets can only decrease down to zero weight, and increase assets can only use available cash from existing cash and reductions. Desired increases are clipped to the single-name cap and, when buy demand exceeds available cash or the daily turnover budget, positive active adjustments are scaled proportionally. The procedure never applies a final proportional rescaling to all holdings, so hold assets remain unchanged and active adjustments preserve the HLC direction. The replay buffer stores the resulting executed adjustment. Target actions in TD3 are passed through the same feasibility layer, and the layer is treated as deterministic post-processing during actor updates.

B Additional Results

Table 6 summarizes trading behavior over the 2020–2023 test period. HRT trades about 25% of the universe on an average day and reduces annualized cost drag relative to HRT-Base, while maintaining lower text-risk exposure. It is not the least concentrated strategy by HHI, which is consistent with the view that HRT balances multiple execution and risk criteria rather than optimizing every metric independently.

Table 5: Implementation configuration. Hyperparameters that affect model selection are chosen using the 2019 validation period only.

Component	Setting
HLC algorithm	PPO
LLC algorithm	TD3
Market encoder	LightGBM on OHLCV and technical factors
Forecast target	Return from next open execution to next scheduled rebalance
Risk proxy	20-day rolling realized volatility
Top-K	$K = 30$ for Momentum Top-K and Alpha Top-K
Min-Variance lookback	252 trading days
Single-name cap	$w_{\max} = 5\%$
LLC turnover budget	$\tau_{\max} = 0.20$
Sparse confidence threshold	0.58, selected on validation
Base transaction cost	10 bps per traded notional
Reward penalties	$\lambda_{turn} = 0.10$, $\lambda_{dd} = 0.05$, $\lambda_{risk} = 0.03$, $\lambda_{act} = 0.01$
HLC reward schedule	$\alpha_t = \alpha_0 \exp(-\lambda t)$, $\alpha_0 = 1.0$, $\lambda = 3 \times 10^{-6}$
HLC / LLC warm-up steps	5×10^4 / 1×10^5
Alternating refinement	One HLC update epoch every five LLC update epochs
PPO learning rate	3×10^{-4}
PPO rollout length / epochs	2048 / 10
PPO clip ratio / entropy coefficient	0.20 / 0.01
TD3 actor / critic learning rates	3×10^{-4} / 1×10^{-3}
TD3 policy delay	2
TD3 target smoothing noise	0.20, clipped at 0.50
TD3 soft update	0.005
Replay buffer / batch size	2×10^5 / 256
Discount factor	0.99
Network architecture	Two-layer MLP, 256 hidden units
Training steps	5×10^5
Execution assumption	Weight-based next-open execution; no integer-share rounding
Model selection	Validation-only selection on 2019
LLM fine-tuning / online querying	None / none

Table 6: Trading behavior over the 2020–2023 test period. Active ratio is active stocks divided by 89. Turnover is $\sum_i |\Delta w_{i,t}^{exec}|$ without the one-half convention. Annualized cost drag is daily turnover $\times$ 10 bps $\times$ 252. HHI is $\sum_i w_{i,t}^2$, averaged over the test period. Text-risk exposure is $\sum_i w_{i,t}\rho_{i,t}$ using the normalized and clipped text-risk score; it is computed ex post for all strategies.

Model	Active Stocks	Active Ratio	Turnover	Ann. Cost Drag	HHI	Text-Risk Exp.
Alpha Top-K	30	0.34	0.210	0.053	0.033	0.48
Flat TD3	40	0.45	0.195	0.049	0.050	0.50
HRT-Base	27	0.30	0.112	0.028	0.044	0.43
HRT	22	0.25	0.090	0.023	0.048	0.35
HRT w/o Sparse HLC	38	0.43	0.150	0.038	0.041	0.38