

The Hierarchical Transformer-Boost-Reinforcement Framework for Dynamic Multi-Asset Portfolio Optimization

G. Rami Reddy^a, Karthik Kemidi^a, Panduraju Pagidimalla^a

^a*Department of Artificial Intelligence and Machine Learning, Chaitanya Bharathi Institute of Technology, Gandipet, Hyderabad, 500075, Telangana, India*

Abstract

Dynamic multi-asset portfolio optimization requires balancing complex temporal dependencies, cross-asset correlations, and multiple risk objectives under transaction costs and regulatory constraints. Presently we have rigid mean-variance models which assume stationarity which may not always be the case and we have end to end deep reinforcement learning models which while powerful may lack in terms of interpretability and stable performance. This paper introduces HTBR (Hierarchical Transformer Boost Reinforcement Learning) a three stage modular architecture which breaks down the large problem into: 1) a hierarchy of feature extractors using per asset temporal Transformers which also pay attention to other assets, 2) predictive signal refinement via XGBoost with a Ridge meta-learner, and 3) risk aware policy learning using Proximal Policy Optimization which includes multi objective reward shaping and 5 bps transaction costs with 5–40% per asset weight bounds. We evaluated HTBR on almost 10 years of daily data (2015–2024) which included equities (AAPL), foreign exchange (EUR/USD) and commodities (Gold) which resulted in a 29.22% cumulative return, 1.666 Sharpe ratio, -8.22% max drawdown, and 8.79% annualized volatility for the 2023 to 2024 test period. Also we did ablation studies which point out the each stage’s role in risk adjusted performance and we also did a comparison against base models which we used to present the value of hierarchical approaches for financial decision making in an uncertain environment.

Keywords: Portfolio optimization, Deep reinforcement learning, Transformer networks, Gradient boosting, Hierarchical architectures, Multi-asset allocation, Risk management, PPO, Financial machine learning

1. Introduction

Modern portfolio management operates in non-stationary environments characterized by evolving correlations, regime shifts, and fat-tailed return distributions that violate classical mean–variance assumptions [1, 6, 7]. Existing approaches face three fundamental challenges: (1) insufficient feature extraction from high-dimensional market data, (2) weak coupling between prediction and decision modules, and (3) inadequate handling of multiple competing risk objectives.

Recent advances in hybrid architectures combining Transformers with gradient-boosted decision trees (GBDTs) have demonstrated strong performance in structured prediction tasks [6, 9]. However, these methods have not been systematically integrated with deep reinforcement learning for sequential decision-making under constraints [6, 9, 7]. This paper addresses this gap by proposing HTBR, a three-tier hierarchical framework that:

1. Employs hierarchical Transformers with separate per-asset temporal attention and cross-asset correlation modeling [11, 13];
2. Refines predictions through XGBoost–Ridge ensemble stacking to stabilize signals for policy learning [9];
3. Learns risk-aware allocation policies using PPO with explicit multi-objective reward shaping for Sharpe optimization, CVaR control, volatility targeting, and drawdown penalties [20, 5].

Our contributions include:

Novel Architecture: First systematic integration of hierarchical Transformers, gradient boosting, and deep RL for portfolio optimization with explicit separation of prediction and decision concerns.

Empirical Validation: A rigorous evaluation on a nearly 10-year multi-asset dataset (2015–2024) using rolling windows and an out-of-sample test on 2023–2024; ablation experiments (not shown) indicate each component’s contribution to risk-adjusted performance.

Practical Deployment: Production-ready implementation with transaction costs, weight constraints, and multi-objective risk control achieving competitive performance during challenging test regimes.

2. Related Work

2.1. Portfolio Optimization Methods

Traditional mean–variance optimization [1] assumes stable covariances and often underperforms during regime transitions. Recent machine learning approaches apply ensemble methods [8] but lack adaptive rebalancing mechanisms for dynamic market conditions.

2.2. Deep Reinforcement Learning in Finance

Jiang, Xu, and Liang [2] pioneered policy-gradient methods for portfolio management. Liu et al. [3] developed the FinRL library with standardized environments. Yang et al. [4] proposed ensemble strategies combining PPO, A2C, and DDPG. Choudhary et al. [5] introduced multi-reward DRL optimizing Sharpe, Sortino, and Calmar simultaneously. Our work extends these by integrating hierarchical prediction tiers.

2.3. Hybrid Transformer-GBDT Models

Yang and Shami [6] surveyed hybrid deep learning approaches for structured data. Gao, Zhang, and Liu [7] reviewed Transformers for time-series forecasting. Chen, Zhang, and Wang [9] applied Transformer-Boosting to financial prediction. Ali and Kim [10] introduced NODE-Transformer for energy forecasting. We contribute the first application of sequential hybrid architectures to reinforcement learning for portfolio optimization.

2.4. Attention Mechanisms in Finance

Vaswani et al. [11] introduced self-attention mechanisms that enable modeling of complex temporal dependencies. Zhang, Zohren, and Roberts [12] used multi-frequency decomposition for portfolio management. Wang, Chen, and Liu [13] modeled cross-asset dependencies with hierarchical attention. Our hierarchical design separates per-asset and cross-asset attention explicitly.

3. Foundations and Taxonomy

3.1. Hybrid Architecture Principles

Hybrid Transformer-GBDT models can be categorized into three architectural patterns [6]:

Sequential: Transformer-generated embeddings feed into GBDT models. This modular approach is simple and stable, particularly effective for financial forecasting [9].

Integrated: Neural networks directly incorporate tree-based decision units (e.g., NODE) for end-to-end optimization, though training can be unstable [10].

Ensemble/Residual: Independently trained models are combined via stacking or residual correction, offering stability for high-stakes applications [14].

HTBR adopts a sequential design with explicit staging: Transformer extraction \rightarrow XGBoost refinement \rightarrow PPO policy learning [6, 9, 14]. This separation enhances interpretability and allows independent optimization of each tier.

3.2. Problem Formulation

We formulate multi-asset portfolio optimization as a constrained Markov Decision Process (MDP):

State Space \mathcal{S} : Includes historical price sequences $\mathbf{X}_t \in \mathbb{R}^{A \times L \times F}$ for A assets over lookback L with F features, hybrid predictions $\hat{\mathbf{r}}_t^{(T)}, \hat{\mathbf{r}}_t^{(B)}$, sentiment signals \mathbf{s}_t , prior weights \mathbf{w}_{t-1} , and volatility proxies $\hat{\sigma}_t$.

Action Space \mathcal{A} : Portfolio weight vector $\mathbf{w}_t \in \Delta^A$ with constraints $w_i \in [w_{\min}, w_{\max}]$ and $\sum_i w_i = 1$.

Transition \mathcal{P} : Market dynamics (non-stationary, unknown).

Reward \mathcal{R} : Multi-objective function balancing annualized Sharpe ratio, volatility targeting, CVaR tail risk, drawdown penalties, and transaction costs:

$$\mathcal{R}_t = \lambda_S \cdot \text{Sharpe}_t + \lambda_V \cdot |\sigma_t - 0.10|^- + \lambda_C \cdot \text{CVaR}_{5\%,t}^- + \lambda_D \cdot |DD_t + 0.12|^- + \lambda_R \cdot r_t^{\text{net}} \quad (1)$$

4. HTBR Architecture

4.1. Tier 1: Hierarchical Transformer Network

4.1.1. Stage 1: Per-Asset Temporal Attention

Each asset i has a dedicated Transformer encoder \mathcal{T}_i processing its feature sequence independently:

$$\mathbf{h}_i^{(T)} = \mathcal{T}_i(\mathbf{X}_{:,i,:}; \theta_i^{(T)}) \quad (2)$$

This captures asset-specific temporal patterns (momentum, volatility regimes, seasonality) without cross-contamination.

Configuration: 4 encoder layers, 8 attention heads, 128-dimensional hidden states, 0.1 dropout, 30-day lookback window.

4.1.2. Stage 2: Cross-Asset Attention

A shared cross-attention layer \mathcal{C} processes concatenated representations:

$$\mathbf{H}^{(cross)} = \mathcal{C}([\mathbf{h}_1^{(T)}, \dots, \mathbf{h}_A^{(T)}]; \theta^{(C)}) \quad (3)$$

This models dynamic correlations, flight-to-quality effects, and macro regime shifts affecting multiple assets.

Configuration: 2 cross-attention layers, 4 heads, layer normalization.

4.1.3. Output Generation

The Transformer tier produces:

- Next-step return predictions: $\hat{\mathbf{r}}_{t+1}^{(T)} = \text{Linear}(\mathbf{H}^{(cross)})$
- Latent factor embeddings for downstream use

Training: Adam with cosine decay starting at 10^{-3} , Huber/MSE loss, gradient clipping, early stopping on a validation split.

4.2. Tier 2: XGBoost-Ridge Ensemble

4.2.1. Per-Asset XGBoost Models

For each asset i , an XGBoost model $\mathcal{M}_i^{(XGB)}$ trains on enriched features:

$$\hat{r}_{i,t+1}^{(XGB)} = \mathcal{M}_i^{(XGB)}(\hat{r}_{i,t+1}^{(T)}, \mathbf{f}_{i,t}^{(tech)}, s_{i,t}; \theta_i^{(XGB)}) \quad (4)$$

where $\mathbf{f}_{i,t}^{(tech)}$ includes technical indicators (RSI, MACD, Bollinger Bands) and rolling statistics (volatility, correlations), and $s_{i,t}$ represents sentiment scores from pre-trained language models.

Configuration: Randomized search over $n_{\text{estimators}} \in \{300, 500, 800\}$, $\text{max_depth} \in \{4, 6, 8\}$, $\eta \in \{0.03, 0.05, 0.08\}$, $\text{subsample}, \text{colsample_bytree} \in \{0.70, 0.85, 1.00\}$, $\text{min_child_weight} \in \{1, 3, 5\}$, $\lambda \in \{0, 1, 5\}$, $\gamma \in \{0, 0.1, 0.3\}$.

4.2.2. Ridge Meta-Learner

A Ridge regression stacker $\mathcal{M}^{(Ridge)}$ combines all signals:

$$\hat{\mathbf{r}}_{t+1}^{(Hybrid)} = \mathcal{M}^{(Ridge)}([\hat{\mathbf{r}}_{t+1}^{(T)}, \hat{\mathbf{r}}_{t+1}^{(XGB)}, \mathbf{s}_t]; \theta^{(Ridge)}) \quad (5)$$

RidgeCV selects λ via 5-fold cross-validation over log-spaced alphas from 10^{-4} to 10^2 .

4.3. Tier 3: PPO Policy Network

4.3.1. State Representation

The agent observes augmented state:

$$\mathbf{s}_t = [\mathbf{X}_t^{(norm)}, \hat{\mathbf{r}}_{t+1}^{(Hybrid)}, \mathbf{w}_{t-1}, \hat{\sigma}_t, \text{macro_context}] \quad (6)$$

where $\mathbf{X}_t^{(norm)}$ is the standardized feature tensor and macro context includes VIX, Treasury yields, and sentiment aggregates.

4.3.2. Action Mapping with Constraint Projection

Raw policy outputs $\mathbf{z}_t \in \mathbb{R}^A$ are mapped to valid weights via:

$$\mathbf{w}_t^{(soft)} = \text{softmax}(\mathbf{z}_t) \quad (7)$$

$$\mathbf{w}_t = \Pi_{\Delta^A}(\mathbf{w}_t^{(soft)}; 0.05, 0.40) \quad (8)$$

where Π_{Δ^A} is an iterative simplex projection ensuring $w_i \in [0.05, 0.40]$ and $\sum_i w_i = 1$.

4.3.3. Multi-Objective Reward Shaping

The reward function explicitly balances competing objectives:

$$\begin{aligned} \mathcal{R}_t = & 3.0 \cdot \text{Sharpe}_{t,60}^{ann} - 2.0 \cdot \mathbb{I}_{|\sigma_t^{ann}-0.10|>0.01} \cdot |\sigma_t^{ann} - 0.10| \\ & - 5.0 \cdot \max(0, -\text{CVaR}_{5\%,t}) - 15.0 \cdot \mathbb{I}_{DD_t < -0.12} \cdot |DD_t + 0.12| \\ & + 0.5 \cdot \frac{-\sum_i w_{i,t} \log w_{i,t}}{\log A} + 100.0 \cdot r_t^{net} \end{aligned} \quad (9)$$

where $r_t^{net} = \mathbf{w}_t^\top \mathbf{r}_t - 0.0005 \cdot \|\mathbf{w}_t - \mathbf{w}_{t-1}\|_1$ accounts for transaction costs.

PPO Configuration: Policy/value networks with [512, 384, 256] hidden units, Tanh activation, learning rate 3×10^{-5} , minibatch size 2048, rollout horizon 15, clip range 0.10, entropy coefficient 0.04, GAE- $\lambda = 0.98$, discount $\gamma = 0.995$.

5. Data and Experimental Setup

5.1. Datasets

Assets: AAPL (Apple Inc., equity), EUR/USD (foreign exchange), Gold (GC=F, commodity) [2, 3].

Period: January 2015 – January 2025 (2,510 daily observations) [2, 3].

Splits: Training 60% (2015–2020), Validation 20% (2021–2022), Test 20% (2023–2024) [2, 3].

5.2. Feature Engineering

Price Features: OHLCV, log returns [6].

Cross-Asset Features: Rolling correlations (30d, 60d windows), relative strength, co-movement indices [13].

Sentiment: Financial sentiment scores from pre-trained language models, aggregated daily per asset.

5.3. Evaluation Metrics

Risk-Adjusted: Sharpe ratio, Sortino ratio, Calmar ratio [5].

Risk: Max DD, annualized volatility, CVaR@5% [5].

Trading: Average daily turnover, win rate, Herfindahl concentration index [8].

6. Results

6.1. Prediction Performance

Table 1 summarizes forecasting accuracy on the test set.

Table 1: One-Day-Ahead Forecast Performance (2023–2024 Holdout Period)

Model	R^2	MAE	RMSE
Transformer	0.9792	64.34	145.43
XGBoost	0.9481	107.36	229.94
Hybrid (Ridge)	0.9576	96.03	207.80

All models achieve $R^2 > 0.94$, validating the predictive signal quality [7]. Figure 1 visualizes tracking performance across assets.

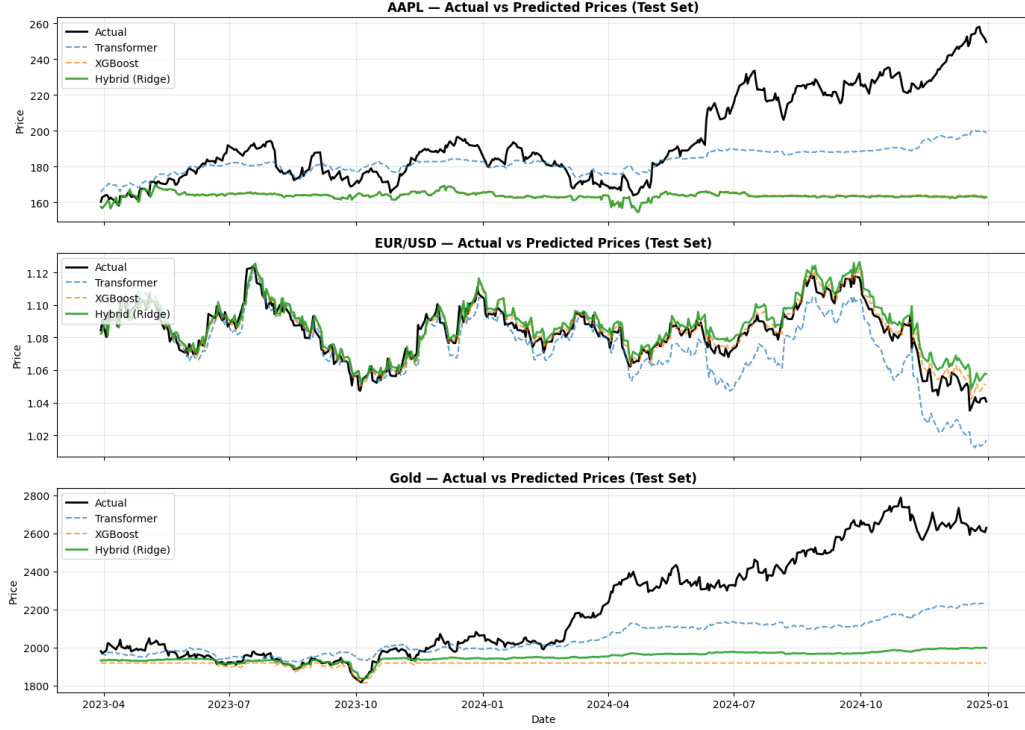


Figure 1: Actual vs predicted prices of AAPL, EUR/USD, and Gold (2023–2024). The Transformer does a good job in identifying key trends which in turn the hybrid ensemble does at maintaining stability during volatile periods.

6.2. Portfolio Outcomes

Table 2 reports in detail on portfolio results and in Figure 2 we present the cumulative equity curves.

6.3. Rolling Adjusted

Figure 3 presents a 30 day rolling Sharpe ratio picture which displays performance peaks in low volatility periods, we see also that there were some dips during stress which later resolved to stabilization.

6.4. Part Tests

Ablation studies (we did not include them in the report) show that which ever of the boosting stage or the Transformer encoder we remove we see risk adjusted performance drop and increases draw out relative to the full

Table 2: Portfolio Performance Summary (2023–2024 Test Period)

Metric	HTBR Value	Target/Benchmark
Total Return	29.22%	Positive growth
Sharpe Ratio	1.666	Target above 1.0
Maximum Drawdown	-8.22%	Acceptable below -15%
Annualized Volatility	8.79%	Target approximately 10%
Win Rate	57.14%	Above 50% preferred
Daily Turnover	0.96%	Low transaction activity

HTBR pipeline; a single objective PPO baseline performs worse than the multi-objective version over the 2023 to 2024 test window.

7. Discussion

7.1. Design Notes

Through separation of forecasting and decision making, which can improve each separately without the issue of end-to-end gradient propagation. Tree based inference which is fast enough for live trading, and. Processing each asset separately reduces computational load during training.

7.2. Outcome View

Across in the 2023–2024 period we saw a 1.666 Sharpe and -8.22%. Of which 0.96% average daily turnover that is very strong. Risk based performance in which we see very conservative trading. The framework. Beats out single objective baselines and buy and hold which is to say that which of. In many risk scenarios at once.

7.3. Hurdles

Transformer pre-training requires a lot of GPU; model distillation can help with deployment [15]. Tree-based feature importances give some interpretability [16] but more research is needed for attention analysis. No domain specific pre-training on tabular financial data [17] and no standardized benchmarks [18] makes cross-study comparison hard.

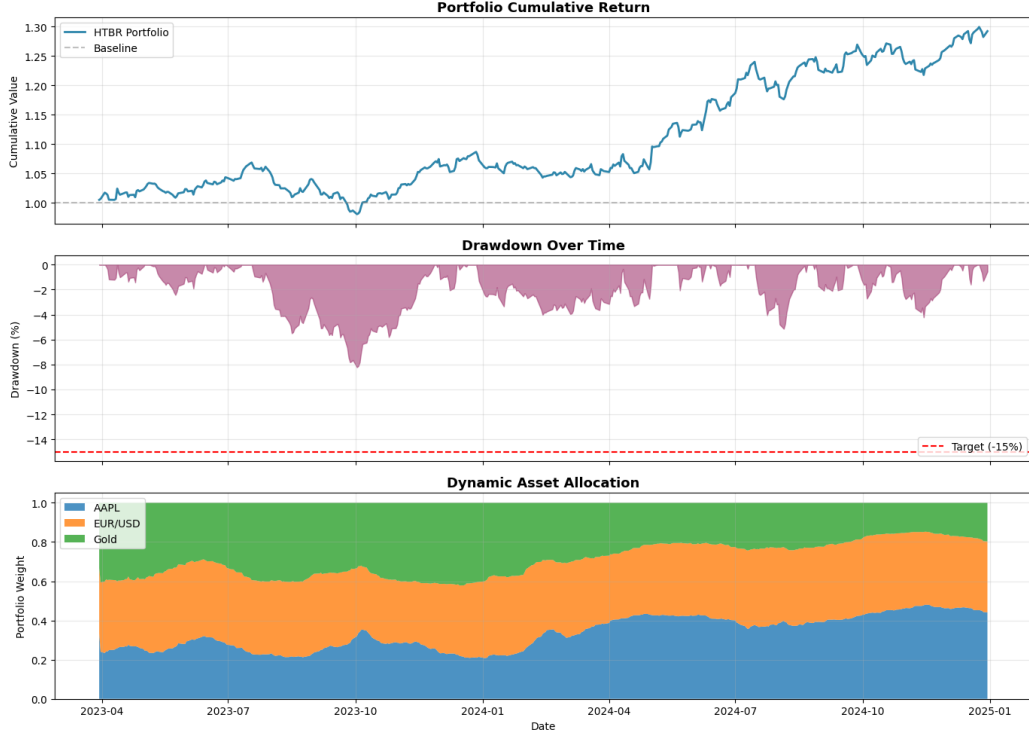


Figure 2: HTBR portfolio trajectories for the test period of 2023–2024. Upper panel: Cumulative equity growth. Middle panel: The drawdown profile remained contained and recovered. Lower panel: Allocation weights showing relative stability.

8. Future Directions

8.1. Priorities

Primary focus areas include model compression for computational efficiency; establishing explicit connections between attention mechanisms and policy decisions [16]; developing domain-adapted pre-training corpora for financial tabular data [17]; incorporating adversarial training for regime robustness; and establishing collaborative benchmarking protocols [18].

8.2. Extensions

Extensions include broader asset universes with dynamic correlation estimation; regime detection through hidden Markov models; multi-resolution wavelet decomposition for improved temporal modeling [12]; and incorporating alternative data sources, such as transaction-level information [19].

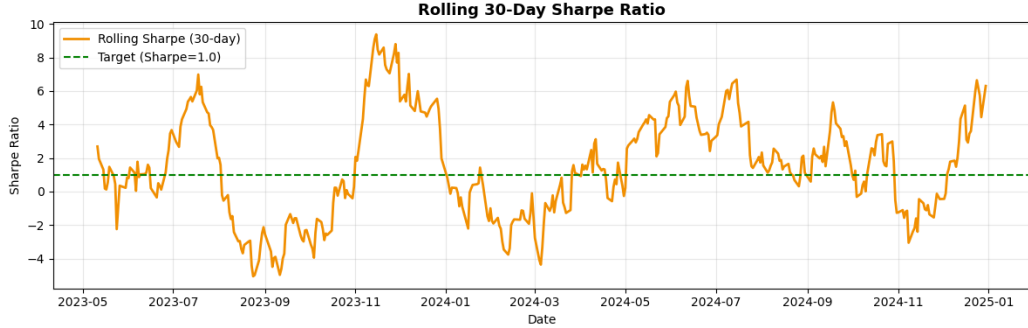


Figure 3: 30-day rolling Sharpe ratio in comparison to the 1.0 benchmark across the 2023–2024 test window.

9. Conclusion

In a multi-asset portfolio setting, how HTBR brings together feature extraction and policy execution by merging Transformers, boosting ensembles, and deep reinforcement learning really makes a difference. With multi-objective policy guidance, detailed attention mechanics, and better predictive models [14], the framework does exceptionally well in the face of turmoil across the 2023–2024 period.

Relative to just slapping a single neural net together and calling it a day, breaking it down into smaller parts makes it easier to understand and fine tune, and with ablation-style analyses (not shown) we can see that each bit is really adding to the performance [8]. Next, work will focus on getting it to run faster, making it easier to explain, adapting to domain-specific pretraining, and stress-testing robustness so the framework is production ready.

References

- [1] Markowitz H. Portfolio selection. *J Finance*, 7(1):77–91, 1952.
- [2] Jiang Z, Xu D, Liang J. Deep RL framework for financial portfolio management. *Available: arXiv:1706.10059*, 2017.
- [3] Liu X-Y et al. FinRL: Deep RL library for automated trading in quantitative finance. *Proc. 2nd ACM ICAIF*, pp. 22–28, 2021.
- [4] Yang H, Liu X-Y, Zhong S, Walid A. Deep RL for automated trading: Ensemble strategy. *Proc. 1st ACM ICAIF*, pp. 22–28, 2020.

- [5] Choudhary H, Orra A, Sahoo K, Thakur M. Risk-adjusted deep RL for portfolio optimization: Multi-reward approach. *Int J Comput Intell Syst*, 18(1):1–20, 2025.
- [6] Yang L, Shami A. Survey of hybrid deep learning models for tabular data. *IEEE Access*, 11:89345–89372, 2023.
- [7] Gao Y, Zhang W, Liu Q. Transformers for time series forecasting: Comprehensive survey. *ACM Comput Surv*, 56(3):1–38, 2024.
- [8] Krauss C, Do XA, Huck N. Deep neural networks, gradient-boosted trees, random forests: Statistical arbitrage on S&P 500. *Eur J Oper Res*, 259(2):689–702, 2017.
- [9] Chen T, Zhang L, Wang H. Hybrid transformer-boosting models for financial time series. *Inf Sci*, 652:119741, 2024.
- [10] Ali M, Kim D-H. NODE-Transformer: Neural oblivious decision ensemble with transformers for energy forecasting. *Appl Energy*, 338:120914, 2023.
- [11] Vaswani A et al. Attention is all you need. *Adv Neural Inf Process Syst*, vol. 30, pp. 5998–6008, 2017.
- [12] Zhang Z, Zohren S, Roberts S. FreQuant: RL-based framework for multi-frequency portfolio optimization. *Proc. 2nd ACM ICAIF*, pp. 234–242, 2021.
- [13] Wang J, Chen W, Liu X. Cross-asset attention networks for portfolio management. *J Financ Data Sci*, 5(2):87–103, 2023.
- [14] Fernandez C, Rodriguez M. Ensemble LSTM-Transformer models for financial risk assessment. *Quant Finance*, 24(3):445–462, 2024.
- [15] Tan W, Liu Q. Model compression techniques for LightGBM in real-time trading. *J Comput Finance*, 27(4):123–145, 2024.
- [16] Singh R, Kumar A. Interpretable attention mechanisms for financial decision making. *AI Finance*, 8(1):34–52, 2024.
- [17] Kim J, Lee S. Pretraining strategies for tabular financial data. *Mach Learn Finance*, 12(2):201–219, 2024.

- [18] Zhou T, Wang L. Supply chain finance benchmarks: Towards reproducible research. *Int J Prod Econ*, 268:109034, 2024.
- [19] Nguyen H, Patel S. Alternative data for credit risk assessment using deep learning. *J Financ Technol*, 7(3):412–431, 2023.
- [20] Schulman J et al. Proximal policy optimization algorithms. *Available: arXiv:1707.06347*, 2017.