BEYOND REGRESSION: PROBABILISTIC RANKING FOR FINANCIAL FORECASTING

Anonymous authors
Paper under double-blind review

ABSTRACT

Financial time-series forecasting is commonly formulated as point regression, although minimizing prediction error does not necessarily produce effective ranking or discrimination of rare, decision-relevant movements. We propose a probabilistic market-state forecasting framework that represents both historical patterns and future returns in discrete spaces. Historical OHLC (Open, High, Low, Close) patches are quantized into tokens and encoded by a lightweight causal Transformer, while upward- and downward-oriented target codebooks define future return states predicted through dual classification heads. The probabilities of selected extreme states are aggregated into directional scores for return ranking and tail-event identification. Experiments on ten major equity indices across multiple chronological periods show consistent improvements over regression-based forecasters in rank correlation and extreme-movement identification. Ablation results show that input and target discretization provide complementary benefits. Furthermore, decision-level fusion combines the state probabilities with existing regression forecasts, improving ranking quality. These results demonstrate that discrete return-state probabilities capture decision-relevant information that point-regression objectives alone do not adequately represent.

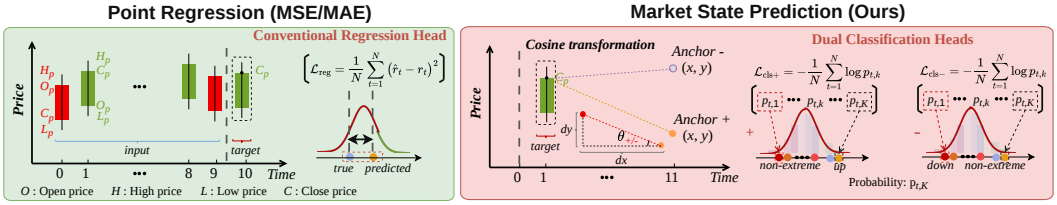

Figure 1: Comparison between point regression and the proposed market-state prediction framework. Point regression minimizes errors between predicted and realized values, whereas our framework models future market-state probabilities from upward and downward directional anchors.

1 INTRODUCTION

Financial time-series forecasting has largely followed the prevailing paradigm of general time-series forecasting, where models are designed to minimize point-wise prediction errors such as mean absolute and squared errors (MAE and MSE) (Xu et al., 2025; Zhu et al., 2025; Bi et al., 2025). However, this objective is not always well aligned with the practical goal of financial prediction. As illustrated in Fig. 1, point regression focuses on how close a predicted value is to the realized value, whereas investment and risk-management decisions may depend more on identifying substantial upward or downward movements (Cenesizoglu & Timmermann, 2012; Zeng et al., 2024). Thus, a forecast with a small numerical error may still fail to capture the directional information most relevant to such decisions (Blaskowitz & Herwartz, 2011; Xia et al., 2024).

Motivated by this mismatch, we propose a market-state probabilistic forecasting framework based on a vanilla Transformer architecture. Instead of relying on a regression head to estimate a future value, the model predicts the probability distribution over market states. Such probabilistic state prediction

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.

is more directly aligned with financial decision making, as it explicitly measures the likelihood of economically meaningful outcomes. Our main contributions are summarized as follows:

- We demonstrate that low point-forecasting error does not necessarily translate into effective return ranking or tail-event identification, motivating financial forecasting as probabilistic market-state prediction rather than conventional regression.
- We propose a lightweight causal Transformer that combines discrete historical patch tokens with upward- and downward-oriented target-state codebooks. The resulting dual probability distributions provide effective signals for global return ranking and extreme-movement identification.
- We show that the learned directional probabilities complement existing regression forecasts through decision-level rank fusion, consistently improving return ordering and bidirectional tail-event identification without retraining or numerically calibrating either model.

The remainder of this paper is organized as follows. Section 2 reviews related work. Section 3 presents the motivating empirical analysis and formulates the forecasting problem. Section 4 introduces the proposed framework. Section 5 reports the main results, ablation studies, fusion experiments, and design analyses. Finally, Section 6 concludes the paper and discusses future research directions. Appendices A and B provide additional experimental details and theoretical analyses.

2 Related Work

Financial time-series forecasting. Early deep-learning studies adopted CNNs (Chen et al., 2020; Huang et al., 2022) and recurrent architectures such as LSTM and GRU, with recurrent models becoming particularly common in financial forecasting (Nasiri & Ebadzadeh, 2023; Lu et al., 2021; Lee & Kim, 2020). More recently, attention-based and Transformer architectures have been increasingly introduced into financial prediction (Wang et al., 2022; Zeng et al., 2023; Zhang et al., 2024). Despite substantial architectural evolution, the underlying forecasting formulation has changed comparatively little: future prices or returns are commonly treated as continuous targets and optimized using point-wise regression losses. As a result, progress in financial forecasting has often been driven by improvements in feature extraction and model architecture (Thakkar & Chaudhari, 2021), while the design of the prediction target and output space has received less attention.

Classification and market-state prediction. Classification has also been explored in financial forecasting, but most studies use coarse directional labels. A common setting predicts whether the future price moves upward or downward (Alamdari et al., 2024), while some studies add a stable class to distinguish upward, downward, and negligible movements (Sezer & Ozbayoglu, 2018; Ding et al., 2021). However, Han et al. (2023) argue that naive binary schemes overlook important market nuances. Neural models may achieve greater practical accuracy by recognizing specific price patterns rather than directional movements alone (Liu & Si, 2022; Zheng et al., 2021). Classification losses have also been combined with regression objectives to provide auxiliary directional or categorical supervision (Park et al., 2022; Sun et al., 2025). Nevertheless, classification in these methods remains auxiliary to continuous forecasting rather than serving as the primary probabilistic output.

Research gap. Existing studies therefore leave a gap between continuous regression and coarse movement classification. Regression preserves return magnitude but may underemphasize decision-relevant market states, whereas binary or ternary classification discards much of the return distribution's structure (Leung et al., 2000). Although richer market states can be identified and forecast from financial data (Procacci & Aste, 2019), they have rarely served as the primary probabilistic output of modern forecasting models. It remains unclear whether a data-driven multi-state output distribution can replace or complement continuous regression at the prediction layer. We address this gap by constructing upward- and downward-oriented target-state codebooks and predicting their probability distributions through dual classification heads, as illustrated in the right panel of Fig. 1.

3 Motivation and Problem Formulation

Limitations of Point Forecasting. Conventional financial forecasting estimates a continuous future target from a historical market sequence. Let $\mathbf{X}_t = [\mathbf{x}_{t-L+1}, \ldots, \mathbf{x}_t] \in \mathbb{R}^{L\times F}$ denote a financial sequence of length L, where $\mathbf{x}_t$ contains the observed OHLC features at time t. A regression model

predicts the next-period return $r_{t+1} = C_{t+1}/C_t - 1$ as $\hat{r}_{t+1} = f_\theta(\mathbf{X}_t)$ and is typically optimized using a point-wise loss such as MSE or MAE. However, low point error does not necessarily imply effective return ranking or reliable identification of market movements.

To examine this issue, we compare five time-series forecasting models across ten equity indices using chronological training, validation, and test periods of 1999–2008, 2009–2014, and 2015–2019, respectively. Each sample uses daily OHLC observations from the previous ten trading days to predict the next-day return. Fig. 2 reports MSE and MAE together with Pearson IC and upward/downward AUC under the 2% return threshold. The results reveal an inconsistency: lower point errors do not consistently correspond to higher IC or stronger extreme-event discrimination. Models with competitive MSE and MAE can exhibit weak ranking or directional AUC, whereas models with larger errors may identify extreme movements more effectively. Thus, minimizing average regression error does not directly optimize financial ranking or tail-event identification.

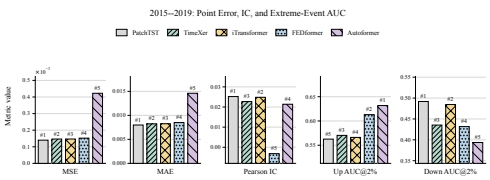

Figure 2: Point-error, ranking, and extreme-event performance during 2015–2019.

Weak Separability of Financial Representations. Identifying extreme upward and downward movements is difficult partly because their representations are weakly separable (Sokolov et al., 2020). Fig. 3 compares the feature distributions of MNIST (LeCun et al., 2010), raw OHLC sequences, and EMA-transformed OHLC sequences. For consistency, all samples are projected into a 64-dimensional space using Gaussian random projection before Fisher-ratio analysis (Sugiyama, 2006) and t-SNE visualization. Financial samples are selected when next-day returns exceed $+2\%$ or fall below -2%. MNIST forms distinct class-dependent clusters, whereas extreme upward and downward samples remain substantially overlapped under both financial representations. Their lower Fisher ratios, as shown in Fig. 3, confirm weak inter-class separation relative to intra-class variation, explaining why conventional financial representations struggle to distinguish extreme directional events.

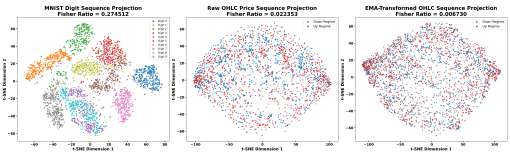

Figure 3: t-SNE visualization of MNIST, raw OHLC, and EMA-transformed OHLC sequences.

Probabilistic Market-State Forecasting. Instead of regressing the future return, we formulate forecasting as probabilistic market-state prediction. For each target, we construct two complementary discrete labels $(s^+_{t+1}, s^-_{t+1}) \in \{1, \ldots, K\}^2$ for the positive- and negative-oriented target spaces. Given the sequence $\mathbf{X}_t$, classification heads produce probability distributions $\mathbf{p}^b_{t+1} = f^b_\theta(\mathbf{X}_t) \in [0,1]^K$, with $\sum_{k=1}^K p^b_{t+1,k} = 1$ and $b \in \{+,-\}$. At inference, the two distributions are converted into directional scores q^+_{t+1} and q^-_{t+1} for ranking upward opportunities and downward risks, respectively, as detailed in Section 4. The heads are jointly optimized by cross-entropy:

$$\mathcal{L} = -\frac{1}{B}\sum_{i=1}^{B}\left(\log p^+_{i,s^+_i} + \log p^-_{i,s^-_i}\right) = \mathcal{L}_{\text{cls}+} + \mathcal{L}_{\text{cls}-}, \qquad B = \text{batch size}. \tag{1}$$

4 Methodology

The proposed framework comprises four components, as illustrated in Fig. 4: (A) quantization of historical patches into discrete input tokens; (B) training-only selection of upward and downward anchors using full-patch geometric representations; (C) target-close transformation and probabilistic prediction through a shared Transformer with two classification heads; and (D) construction and return-based ordering of two directional target codebooks. Three distinct K-Means codebooks are used: one for historical inputs and two for upward- and downward-oriented target states.

A. Causal Normalization and Input-Patch Tokenization. Given an 11-day financial sequence, the first 10 days provide historical observations and day 11 defines the next-day prediction target. To remove absolute price-scale differences, each OHLC variable is normalized by its value on the last historical day. Specifically, for $v \in \{O, H, L, C\}$, we define $\tilde{x}^{(v)}_{i,t} = x^{(v)}_{i,t}/x^{(v)}_{i,10}$ for $t = 1, \ldots, 11$.

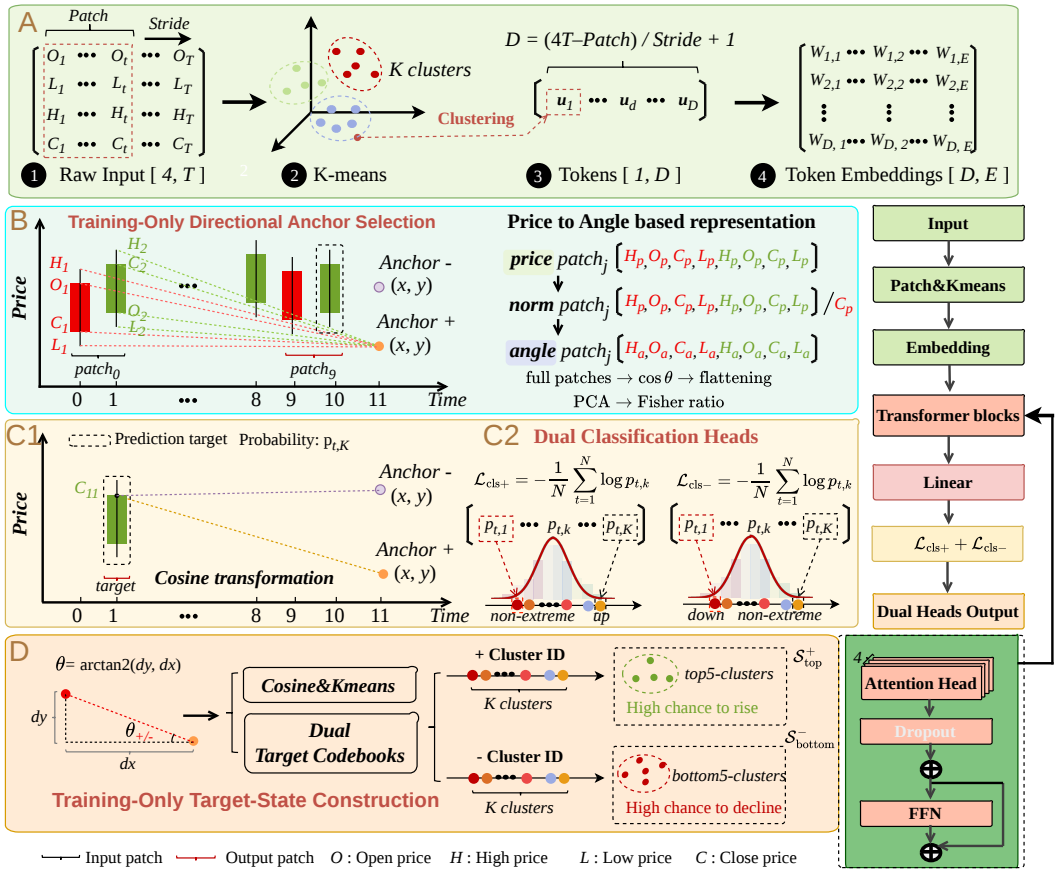

Figure 4: Overview of the framework, **(A)** historical patch tokenization, **(B)** directional anchor selection, **(C)** dual-state prediction, **(D)** directional target-codebook construction and ranking.

The normalized OHLC observations are flattened in temporal order and divided into ten overlapping patches of length $P = 8$ with stride $S = 4$. Each patch is rescaled by its most recent closing price as the local reference. Let $\mathbf{p}_{i,j} \in \mathbb{R}^P$ denote the resulting j-th raw patch. An input codebook $\mathcal{C}^{\text{in}} = \{\mathbf{c}_1^{\text{in}}, \ldots, \mathbf{c}_{K_{\text{in}}}^{\text{in}}\}$ is learned from the training patches, where $K_{\text{in}} = 100$, and each patch is assigned to its nearest centroid as $u_{i,j} = \arg\min_{k \in \{1,\ldots,K_{\text{in}}\}} \|\mathbf{p}_{i,j} - \mathbf{c}_k^{\text{in}}\|_2^2$. The codebook remains fixed during validation and testing. Although patch construction produces ten patch states, only the first nine are used as the Transformer input; the final patch contains next-day observations and is excluded. The input sequence is therefore $(u_{i,1}, \ldots, u_{i,9})$.

B. Directional Anchor Selection. The upward and downward anchors are selected from full-patch geometric representations using only the training data. Let $A^+ = (a_x, a_y^+)$ and $A^- = (a_x, a_y^-)$ denote the two directional anchors. For branch $b \in \{+, -\}$, the patch-level angular representation is transformed using the cosine function, after which the features of all patches are flattened and projected onto a 32-dimensional space:

$$\boldsymbol{\theta}_{i,j}^b(a_y) = \arctan2(a_y - \mathbf{p}_{i,j},\, a_x - x_j), \quad \mathbf{h}_i^b(a_y) = \text{PCA}_{32}\big(\cos\big(\text{vec}\big[\{\boldsymbol{\theta}_{i,j}^b(a_y)\}_{j=1}^{10}\big]\big)\big). \quad (2)$$

Each sample contains ten patches indexed by $x_j = j$ for $j = 1, \ldots, 10$. The horizontal anchor coordinate is fixed at $a_x = 11$, immediately beyond the final patch. Candidate vertical coordinates are sampled uniformly between the minimum and maximum training-patch values. The upward and downward anchors are searched independently using the branch-specific binary labels $z_i^+ = \mathbb{I}(r_i > 0.01)$ and $z_i^- = \mathbb{I}(r_i < -0.01)$, respectively. For each candidate vertical coordinate, directional separability in the resulting representation is evaluated using the Fisher discriminant ratio. The two vertical coordinates are selected independently:

$$J^b(a_y) = S_B^b(a_y)/S_W^b(a_y), \qquad a_y^+ = \arg\max_{a_y} J^+(a_y), \qquad a_y^- = \arg\max_{a_y} J^-(a_y). \quad (3)$$

Here, S_B^b and S_W^b denote the between-class and within-class scatter computed from $\mathbf{h}_i^b(a_y)$ under the binary labels. These directional labels are used only to evaluate anchor separability and are not used as training targets. Once selected, the anchors remain fixed during validation and testing.

C.1 Target-Close Dual K-Means Codebooks. Let $\widetilde{C}_i = C_{i,11}/C_{i,10}$ denote the target closing-price ratio of sample i, which alone is used to construct the output states. Given target coordinate $x = 1$ and anchor $A^b = (a_x, a_y^b)$, we define $g^b(\widetilde{C}_i) = \cos[\arctan2(a_y^b - \widetilde{C}_i, a_x - 1)]$ for $b \in \{+, -\}$. For each direction, a K-Means codebook $\mathcal{C}_{\text{tar}}^b = \{c_1^b, \ldots, c_{K_b}^b\}$ is fitted using training targets only, with $K_+ = K_- = 100$. The target state is assigned as $s_i^b = \arg\min_{k \in \{1,\ldots,K_b\}} |g^b(\widetilde{C}_i) - c_k^b|^2$ for $b \in \{+, -\}$. Thus, s_i^+ and s_i^- provide directional discretizations of the target closing-price ratio. The target codebooks are independent of $\mathcal{C}^{\text{in}}$ and remain fixed during validation and testing.

C.2 Shared Causal Transformer with Dual Classification Heads. Each input state is represented by the sum of a token embedding and a learned positional embedding, $\mathbf{e}_{i,j} = E_{\text{tok}}(u_{i,j}) + E_{\text{pos}}(j)$ for $j = 1, \ldots, 9$. The nine embeddings are processed by a shared causal Transformer encoder with embedding dimension 64, three blocks, four attention heads per block, GELU feed-forward layers, and dropout rate 0.1. A lower-triangular mask prevents attention to later token positions. The final hidden state is passed to two classification heads, each comprising two layers, which predict the target-state distributions $\mathbf{p}_i^+$ and $\mathbf{p}_i^-$. The model is trained with the dual cross-entropy objective in Eq. 1. Optimization uses AdamW with learning rate 10^{-3}, weight decay 10^{-4}, gradient-norm clipping at 1, and a batch size of 32 for training, validation, and testing. Training runs for up to 60 epochs with early stopping after 10 epochs without validation-loss improvement.

D. Probability-Based Upward and Downward Ranking. Because K-Means cluster indices are nominal, we order the clusters by their training-set mean returns, defined as $\mu_k^b = \mathbb{E}_{\text{train}}[r_i \mid s_i^b = k]$ for $b \in \{+, -\}$. Let $\mathcal{S}_{\text{top}}^+$ contain the five upward-branch clusters with the largest μ_k^+ and $\mathcal{S}_{\text{bottom}}^-$ the five downward-branch clusters with the smallest μ_k^-. We define $q_i^+ = \sum_{k \in \mathcal{S}_{\text{top}}^+} p_{i,k}^+$ and $q_i^- = \sum_{k \in \mathcal{S}_{\text{bottom}}^-} p_{i,k}^-$. The former ranks return opportunities and identifies extreme upward movements, whereas the latter identifies extreme downward movements. The cluster ordering and Top-5 and Bottom-5 state sets are determined from training data and remain fixed during validation and testing.

5 Experiments

Experimental Setup and Evaluation Metrics. We evaluate Ours across ten equity indices: S&P 500, Dow Jones, DAX, FTSE 100, CAC 40, Nikkei 225, Hang Seng, SSE 50, CSI 300, and NASDAQ. Data are collected using AKShare (King & Zhang, 2022). We use three chronological train–validation–test splits: 1999–2008 / 2009–2014 / 2015–2019, 2005–2014 / 2015–2020 / 2021–2025, and 2009–2018 / 2019–2024 / 2025–January 2026. The first two compare against five representative forecasting baselines (Wang et al., 2024a), while the last compares with Kronos-small and Kronos-base; model sizes and input–output formulations are summarized in Table 1. Return ranking is evaluated by Spearman IC and Kendall's τ. For upward and downward events, defined by $r_i \geq 0.02$ and $r_i \leq -0.02$, we report the area under the receiver operating characteristic curve (AUROC), average precision (AP), and area under the precision–recall curve (AUPRC). Results are averaged over seeds 1337, 2337, and 3337. Baseline configurations are provided in Appendix A.4.

Table 1: Model size and input–output formulation.

Method	Parameters	Relative size	Input tokenization	Forecast output
Ours	**0.178M**	**1.0×**	Discrete patches	Dual-state probabilities
Autoformer (Wu et al., 2021)	0.291M	1.6×	Point-wise	Continuous return
FEDformer (Zhou et al., 2022)	0.381M	2.1×	Point-wise	Continuous return
PatchTST (Nie et al., 2023)	0.403M	2.3×	Patch-wise	Continuous return
iTransformer (Liu et al., 2024)	6.329M	35.6×	Variate-wise	Continuous return
TimeXer (Wang et al., 2024b)	8.445M	47.5×	Patch-wise	Continuous return
Kronos-small (Shi et al., 2026)	24.7M	138.9×	Discrete K-lines	Continuous prices
Kronos-base (Shi et al., 2026)	102.3M	575.4×	Discrete K-lines	Continuous prices

5.1 Comparison with Point-Forecasting Models

Ours consistently improves return ranking and tail-event identification. Fig. 5 compares Ours with forecasting baselines across three test periods. In both 2015–2019 and 2021–2025, Ours achieves the highest Spearman IC and Kendall's τ, while also substantially outperforming all con-

tinuous baselines in upward and downward tail identification. The gains are especially pronounced for downward events, where several regression baselines remain close to random discrimination.

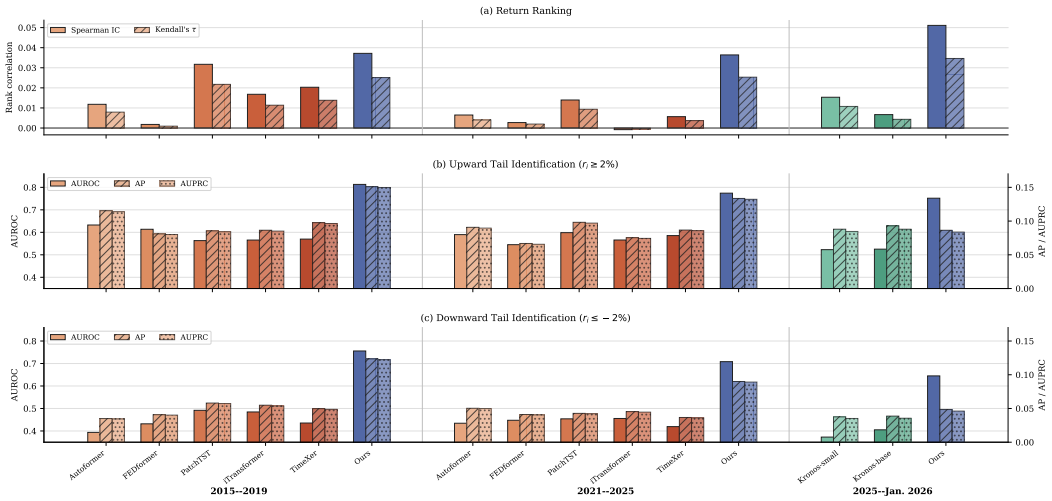

Figure 5: Main results for return ranking and 2% upward and downward tail-event identification.

The advantage persists in the recent out-of-pretraining evaluation. On 2025–January 2026, Ours remains substantially stronger than Kronos-small and Kronos-base in rank correlation, upward and downward AUROC, and downward-tail identification. Although Kronos models achieve slightly higher upward AP and AUPRC, Ours shows the strongest overall performance across ranking and both directional tails. Complete results are reported in Table 9.

5.2 Ablation Study

We examine five variants of the complete model to isolate the effects of input discretization, directional target separation, and probabilistic state prediction. When directional anchoring is removed, they are replaced by a shared $K = 200$ codebook to preserve the target-state capacity. We also consider an upward-only $K = 100$ codebook and a continuous-output model. To separate dual-branch training from the scoring rule, we additionally evaluate the complete model using the bottom-five states of its upward codebook for downward scoring, matching the upward-only variant. Matching colors link the table rows to the corresponding discussions below. Table 2 reports the results.

Table 2: Ablation results for the principal model components.

Input representation		Target representation			Rank correlation		Upward		Downward	
Patch	Discrete	Anchors	q_i^+	q_i^-	IC	τ	AUROC	AP	AUROC	AP
Test period: 2015–2019										
×	✓	✓	✓	✓	**.0372 ± .0032**	**.0251 ± .0023**	**.8131 ± .0046**	**.1510 ± .0060**	.7557 ± .0006	.1236 ± .0041
×	✓	✓	✓	×	**.0372 ± .0032**	**.0251 ± .0023**	**.8131 ± .0046**	**.1510 ± .0060**	**.7585 ± .0033**	**.1269 ± .0015**
✓	×	✓	✓	✓	.0112 ± .0112	.0078 ± .0075	.5692 ± .0426	.0848 ± .0173	.5011 ± .0409	.0644 ± .0102
×	✓	×	×	×	.0354 ± .0025	.0235 ± .0015	.8122 ± .0064	.1505 ± .0031	.7538 ± .0022	.1222 ± .0047
✓	×	×	×	×	.0108 ± .0069	.0074 ± .0046	.5567 ± .0255	.0783 ± .0096	.5365 ± .0354	.0755 ± .0057
×	✓	✓	✓	–	.0362 ± .0060	.0244 ± .0048	.8065 ± .0043	.1468 ± .0035	.7558 ± .0056	.1229 ± .0041
✓	×	×	–	–	.0005 ± .0234	.0003 ± .0160	.4983 ± .0797	.0665 ± .0185	.5006 ± .0649	.0681 ± .0209
Test period: 2021–2025										
×	✓	✓	✓	✓	**.0364 ± .0036**	**.0253 ± .0024**	**.7741 ± .0036**	**.1336 ± .0057**	.7082 ± .0036	.0900 ± .0006
×	✓	✓	✓	×	**.0364 ± .0036**	**.0253 ± .0024**	**.7741 ± .0036**	**.1336 ± .0057**	**.7206 ± .0025**	.0947 ± .0029
✓	×	✓	✓	✓	.0113 ± .0025	.0077 ± .0016	.5440 ± .0073	.0691 ± .0057	.5090 ± .0105	.0533 ± .0011
×	✓	×	×	×	.0314 ± .0024	.0219 ± .0014	.7697 ± .0076	.1307 ± .0037	.7129 ± .0065	.0886 ± .0027
✓	×	×	×	×	.0053 ± .0118	.0037 ± .0080	.5146 ± .0500	.0575 ± .0103	.5018 ± .0322	.0539 ± .0016
×	✓	✓	✓	–	.0303 ± .0029	.0206 ± .0017	.7667 ± .0050	.1273 ± .0011	.7175 ± .0100	**.0959 ± .0033**
✓	×	×	–	–	.0069 ± .0053	.0047 ± .0036	.5263 ± .0214	.0595 ± .0033	.4831 ± .0146	.0566 ± .0092

$q_i^+ = \checkmark$, $q_i^- = -$: upward-only; $q_i^+ = -$, $q_i^- = -$: continuous output; $q_i^+ = \times$, $q_i^- = \times$: shared $K = 200$ codebook.

Input discretization. Under the fixed dual target-close codebooks, replacing discrete input tokens with continuous patches substantially reduces ranking and tail-event performance and increases variation across seeds. This controlled comparison demonstrates the importance of input discretization within the proposed configuration.

Directional target separation. Removing directional anchoring yields a shared $K = 200$ codebook, whose top-five and bottom-five states define the two scores. The full model outperforms this capacity-matched variant on every metric in 2015–2019 and on all but downward AUROC in 2021–2025. This advantage supports the benefit of separating upward and downward target spaces, although the single exception indicates that the improvement is not uniform across all metrics.

Upward-only prediction and downward scoring. The upward-only variant uses a single upward-oriented $K = 100$ codebook for both directional scores. For a matched comparison, we apply the same top-five/bottom-five scoring rule to the full model while retaining its dual-space training. Under this common scoring scheme, the full model performs better on every metric in 2015–2019 and on all but downward AP in 2021–2025. This shows that the dual target spaces provide complementary supervision beyond what can be learned from a single upward-oriented space, leading to stronger overall representations and performance.

Joint removal and continuous output. Removing both input discretization and directional anchoring sharply weakens performance even when the shared $K = 200$ codebook is retained. Replacing the discrete target states with a continuous output further produces unstable rank correlation and near-random tail discrimination. Overall, the results support the joint use of discrete historical representations and probabilistic target states, while indicating that their contributions depend on how the input, target, and scoring spaces are combined.

5.3 Complementarity with Regression Forecasts

We examine whether the directional information learned by Ours complements continuous return forecasts. Test predictions from Ours and each regression baseline are matched by market and prediction date and combined post hoc, without retraining either model.

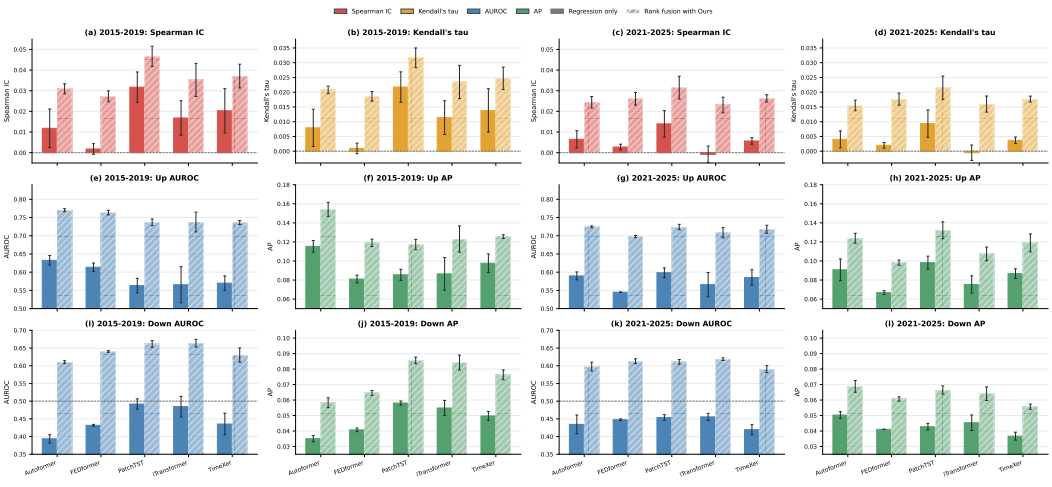

Figure 6: Complementarity between Ours and continuous regression forecasts.

Rank fusion. For ranking and tail-event detection, the regression forecast and two directional scores are converted to percentile ranks. Let $Q_{r,i}$ and $Q_{-r,i}$ denote the ranks of the regression-implied return $\hat{r}_i^{\text{reg}}$ and its negative, and let $Q_{+,i}$ and $Q_{-,i}$ denote those of the upward and downward scores produced by Ours. We define $f_i^+ = (1-\lambda)Q_{r,i} + \lambda Q_{+,i}$ and $f_i^- = (1-\lambda)Q_{-r,i} + \lambda Q_{-,i}$, where f_i^+ ranks upward opportunities, f_i^- ranks downward risk, and $\lambda = 0.50$.

Results. Fig. 6(a)–(d) shows that fusion improves both Spearman IC and Kendall's τ for all five regression baselines in both periods. The improvements therefore reflect more consistent return ordering rather than a change confined to a single rank-correlation measure. Panels (e)–(h) show corresponding gains in upward AUROC and AP, while Panels (i)–(l) demonstrate consistent improvements in downward-tail identification. The benefits hold across forecasting architectures and both chronological evaluations. The consistent gains therefore indicate that the predicted state prob-

abilities provide ordering and tail-event information not captured by continuous regression forecasts alone. Complete numerical results are reported in Table 8.

5.4 Directional-State Design Analysis

We analyze directional-state design choices, including anchor position, geometric encoding, search representation, selection threshold, state aggregation, and downward-score source.

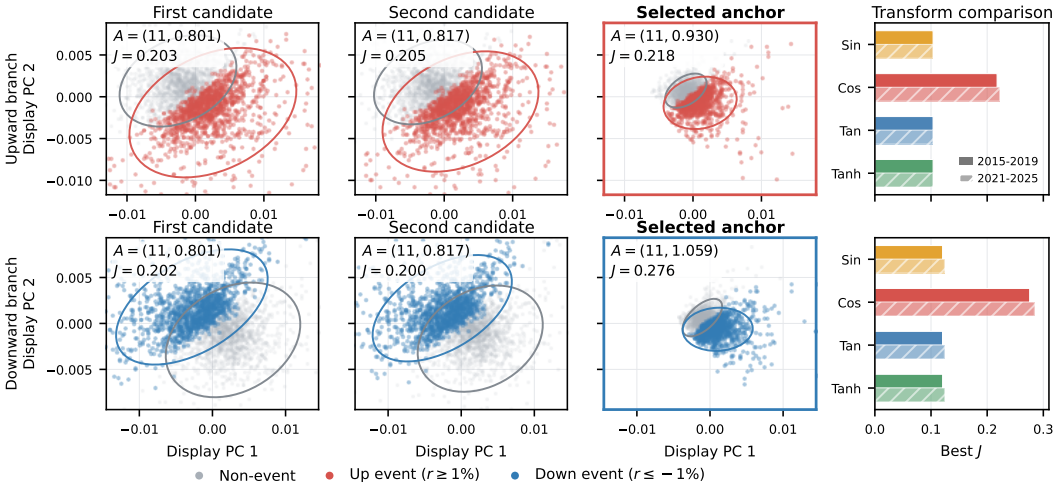

Figure 7: Directional-anchor search on 2015–2019 data and geometric-encoding comparison.

Anchor position and geometric encoding. The left panels of Fig. 7 compare the first two candidates with the selected anchor. The optimized anchors increase directional separability, particularly for downward events, and produce more compact event regions in the two-dimensional projection. This indicates that anchor optimization enlarges the separation between extreme and non-extreme samples in the transformed representation. The right panels compare sine, cosine, tangent, and hyperbolic-tangent encodings under the same search procedure. Cosine consistently achieves the highest Fisher ratio across branches and periods and is therefore used in the main experiments.

Table 3: Sensitivity analysis of directional-state construction.

Design				Ranking		Upward		Downward	
Full-patch	δ	M	q_i^-	Spearman	Kendall	AUROC	AP	AUROC	AP
2015–2019									
✓	1%	5	✓	.0372 ± .0032	.0251 ± .0023	.8131 ± .0046	.1510 ± .0060	.7557 ± .0006	.1236 ± .0041
×	1%	5	✓	.0320 ± .0067	.0215 ± .0046	.7958 ± .0220	.1488 ± .0034	.7360 ± .0254	.1232 ± .0068
✓	2%	5	✓	.0375 ± .0021	.0251 ± .0016	.8049 ± .0109	.1483 ± .0049	.7466 ± .0084	.1202 ± .0021
✓	1%	5	×	.0372 ± .0032	.0251 ± .0023	.8131 ± .0046	.1510 ± .0060	.7585 ± .0033	.1269 ± .0015
✓	1%	40	✓	.0375 ± .0023	.0251 ± .0015	.8128 ± .0044	.1511 ± .0059	.7576 ± .0041	.1235 ± .0035
✓	1%	40	×	.0375 ± .0023	.0251 ± .0015	.8128 ± .0044	.1511 ± .0059	.2484 ± .0050	.0231 ± .0001
2021–2025									
✓	1%	5	✓	.0364 ± .0036	.0253 ± .0024	.7741 ± .0036	.1336 ± .0057	.7082 ± .0036	.0900 ± .0006
×	1%	5	✓	.0298 ± .0019	.0202 ± .0018	.7479 ± .0207	.1288 ± .0059	.7036 ± .0075	.0920 ± .0012
✓	2%	5	✓	.0270 ± .0043	.0186 ± .0029	.7600 ± .0089	.1212 ± .0119	.7102 ± .0093	.0894 ± .0019
✓	1%	5	×	.0364 ± .0036	.0253 ± .0024	.7741 ± .0036	.1336 ± .0057	.7206 ± .0025	.0947 ± .0029
✓	1%	40	✓	.0314 ± .0046	.0215 ± .0030	.7754 ± .0017	.1367 ± .0008	.7194 ± .0014	.0930 ± .0024
✓	1%	40	×	.0314 ± .0046	.0215 ± .0030	.7754 ± .0017	.1367 ± .0008	.2891 ± .0039	.0245 ± .0001

Search representation and selection threshold. Table 3 further evaluates the anchor-search procedure. Full-patch search generally outperforms target-close-only search, especially in ranking and directional discrimination, indicating that the complete patch geometry contains information unavailable from the target close alone. The default threshold $\delta = 1\%$ also provides better overall performance than $\delta = 2\%$, despite isolated improvements under the latter.

State aggregation and score source. Increasing the number of aggregated states from $M = 5$ to $M = 40$ has a limited effect when the two directional scores are used, showing that the dedicated

branches remain robust over a broader aggregation range. Using the upward codebook for both directions can remain competitive at $M = 5$, but its downward performance deteriorates sharply at $M = 40$. In contrast, the dedicated q_i^- score remains stable, demonstrating that the two directional codebooks capture increasingly complementary information as more states are aggregated.

5.5 Output-Formulation Analysis

We compare four output formulations while fixing the discrete input tokens and Transformer backbone. The proposed formulation uses two data-adaptive $K = 100$ K-Means codebooks. We replace them with dual equal-width $K = 100$ codebooks, two binary heads defined by the $\pm 2\%$ thresholds, or a single three-class head representing downward, neutral, and upward movements.

Discrete output formulations are broadly effective. Fig. 8(a)–(c) shows that, within the same discrete-input Transformer framework, replacing the proposed K-Means output space with equal-width binning or direct classification still yields competitive ranking and tail-identification performance. This indicates that the framework is not tied to a particular output discretization and can accommodate simpler classification formulations while retaining strong performance.

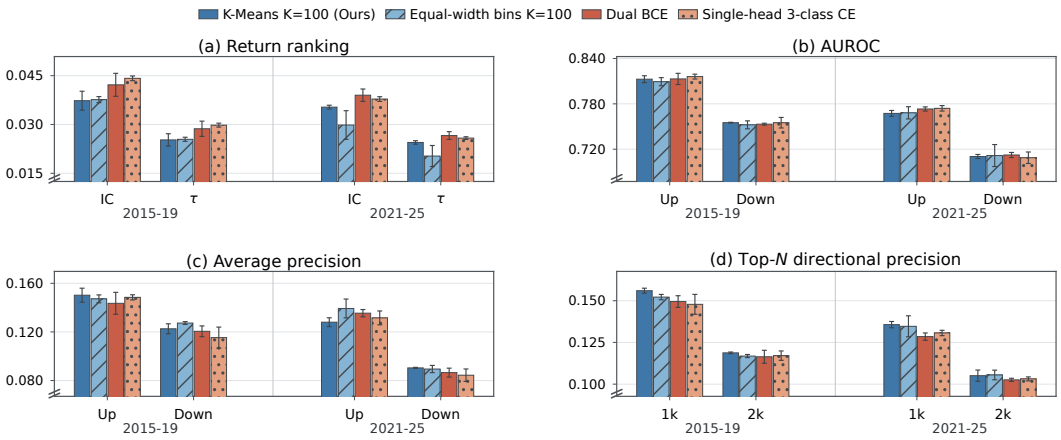

Figure 8: Comparison of output formulations in ranking, tail identification, and Top-N precision.

K-Means however improves high-confidence event concentration. We further examine whether the highest-confidence predictions concentrate realized tail events. Let $\mathcal{T}_N^+$ and $\mathcal{T}_N^-$ contain the N observations with the largest q_i^+ and q_i^-, respectively. We define average directional precision as $\mathrm{DP@N} = \frac{1}{2}\left[N^{-1}\sum_{i\in\mathcal{T}_N^+}\mathbb{I}(r_i \geq 0.02) + N^{-1}\sum_{i\in\mathcal{T}_N^-}\mathbb{I}(r_i \leq -0.02)\right]$. Fig. 8(d) shows that the proposed K-Means formulation concentrates more extreme upward and downward events among the highest-ranked observations, achieving the highest directional precision at $N = 1000$ in both periods and at $N = 2000$ in 2015–2019, while remaining nearly tied with equal-width binning in 2021–2025. This suggests that finer data-adaptive output states provide a more selective ordering of extreme movements at the top of the ranking. Such concentration is not fully reflected by aggregate ranking or discrimination metrics such as IC, AUROC, and AP.

6 Conclusion

This paper reframes financial forecasting around return ranking and extreme-event identification. We discretize historical OHLC patches and future outcomes into input tokens and directional states, whose probabilities are predicted by a lightweight causal Transformer. Across markets and periods, these signals improve ranking and tail-event detection and complement continuous forecasts through rank fusion. Alternative discrete outputs are also effective, while the adaptive dual codebooks more consistently concentrate extreme events among high-confidence predictions. Future work may focus on data augmentation tailored to discrete input–output representations and explore joint objectives in which point-wise regression losses provide auxiliary supervision for probabilistic state classification.

AI USE STATEMENT

Generative AI tools were used to assist with manuscript editing, clarifying mathematical notation, methodological presentation, and reviewing implementation details for consistency with the manuscript. They were also used for literature exploration, language refinement, and improving the organization and readability of the paper.

All research ideas, methodological decisions, experimental settings, implementations, analyses, and conclusions were reviewed and verified by the authors. AI-generated suggestions were not accepted without manual inspection, and reported numerical results were obtained from the authors' experimental implementations rather than generated by AI. The authors take full responsibility for the final content of this work, including all text, claims, experiments, and conclusions.

ETHICS STATEMENT

This work does not involve human subjects, personal data, or privacy-sensitive information. All financial market data used in the experiments were obtained from publicly available sources. The study focuses on methodological research for financial time-series forecasting and does not provide individualized financial advice or recommendations.

REPRODUCIBILITY STATEMENT

To facilitate reproducibility, we provide an anonymized repository containing the dataset and a self-contained implementation of the proposed method at `https://anonymous.4open.science/r/Dual_KMeans-EE7B`. The accompanying notebook documents the principal preprocessing, codebook construction, training, evaluation, and result-generation steps and can be executed directly in Google Colab. Additional experimental settings, evaluation details, and theoretical analyses are provided in the appendix.

REFERENCES

Milad Kamali Alamdari, Akbar Esfahanipour, and Hossein Dastkhan. A portfolio trading system using a novel pixel graph network for stock selection and a mean-cdar optimization for portfolio rebalancing. *Applied Soft Computing*, 152:111213, 2024.

Yu Bi, Zhuohan Wang, Lingxiao Zhao, and Carmine Ventre. Long-term financial forecasting and trading via multi-agent reinforcement learning. In *Proceedings of the 6th ACM International Conference on AI in Finance*, ICAIF '25, pp. 788–796, New York, NY, USA, 2025. Association for Computing Machinery. ISBN 9798400722202. doi: 10.1145/3768292.3770411.

Oliver Blaskowitz and Helmut Herwartz. On economic evaluation of directional forecasts. *International Journal of Forecasting*, 27(4):1058–1065, 2011. ISSN 0169-2070. doi: https://doi.org/10.1016/j.ijforecast.2010.07.002.

Tolga Cenesizoglu and Allan Timmermann. Do return prediction models add economic value? *Journal of Banking & Finance*, 36(11):2974–2987, 2012. ISSN 0378-4266. doi: https://doi.org/10.1016/j.jbankfin.2012.06.008. International Corporate Finance Governance Conference.

Chunchun Chen, Pu Zhang, Yuan Liu, and Jun Liu. Financial quantitative investment using convolutional neural network and deep learning technology. *Neurocomputing*, 390:384–390, 2020.

Qianggang Ding, Sifan Wu, Hao Sun, Jiadong Guo, and Jian Guo. Hierarchical multi-scale gaussian transformer for stock movement prediction. In *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence*, IJCAI'20, 2021. ISBN 9780999241165.

Yechan Han, Jaeyun Kim, and David Enke. A machine learning trading system for the stock market based on n-period min-max labeling using xgboost. *Expert systems with applications*, 211(C), January 2023. ISSN 0957-4174.

Wenyang Huang, Huiwen Wang, Haotong Qin, Yigang Wei, and Julien Chevallier. Convolutional neural network forecasting of european union allowances futures using a novel unconstrained

transformation method. *Energy Economics*, 110:106049, 2022. ISSN 0140-9883. doi: https://doi.org/10.1016/j.eneco.2022.106049.

Albert King and Yaojie Zhang. Akshare. `https://github.com/akfamily/akshare`, 2022.

Yann LeCun, Corinna Cortes, and CJ Burges. Mnist handwritten digit database. *ATT Labs [Online]. Available: http://yann.lecun.com/exdb/mnist*, 2, 2010.

Si Woon Lee and Ha Young Kim. Stock market forecasting with super-high dimensional time-series data using convlstm, trend sampling, and specialized data augmentation. *Expert Systems with Applications*, 161:113704, 2020. ISSN 0957-4174. doi: https://doi.org/10.1016/j.eswa.2020.113704.

Mark T. Leung, Hazem Daouk, and An-Sing Chen. Forecasting stock indices: a comparison of classification and level estimation models. *International Journal of Forecasting*, 16(2):173–190, 2000. ISSN 0169-2070. doi: https://doi.org/10.1016/S0169-2070(99)00048-5.

Liying Liu and Yain-Whar Si. 1d convolutional neural networks for chart pattern classification in financial time series. *The Journal of Supercomputing*, 78(12):14191–14214, 2022.

Yong Liu, Tengge Hu, Haoran Zhang, Haixu Wu, Shiyu Wang, Lintao Ma, and Mingsheng Long. itransformer: Inverted transformers are effective for time series forecasting. In *The Twelfth International Conference on Learning Representations*, 2024.

Wenjie Lu, Jiazheng Li, Jingyang Wang, and Lele Qin. A cnn-bilstm-am method for stock price prediction. *Neural Comput. Appl.*, 33(10):4741–4753, May 2021. ISSN 0941-0643. doi: 10.1007/s00521-020-05532-z.

Hamid Nasiri and Mohammad Mehdi Ebadzadeh. Multi-step-ahead stock price prediction using recurrent fuzzy neural network and variational mode decomposition. *Applied Soft Computing*, 148:110867, 2023. ISSN 1568-4946. doi: https://doi.org/10.1016/j.asoc.2023.110867.

Yuqi Nie, Nam H Nguyen, Phanwadee Sinthong, and Jayant Kalagnanam. A time series is worth 64 words: Long-term forecasting with transformers. In *The 11th International Conference on Learning Representations*, 2023.

Hyun Jun Park, Youngjun Kim, and Ha Young Kim. Stock market forecasting using a multi-task approach integrating long short-term memory and the random forest framework. *Applied Soft Computing*, 114:108106, 2022. ISSN 1568-4946. doi: https://doi.org/10.1016/j.asoc.2021.108106.

Pier Francesco Procacci and Tomaso Aste. Forecasting market states. *Quantitative Finance*, 19(9):1491–1498, 2019. doi: 10.1080/14697688.2019.1622313.

Omer Berat Sezer and Ahmet Murat Ozbayoglu. Algorithmic financial trading with deep convolutional neural networks: Time series to image conversion approach. *Applied Soft Computing*, 70:525–538, 2018. ISSN 1568-4946. doi: https://doi.org/10.1016/j.asoc.2018.04.024.

Yu Shi, Zongliang Fu, Shuo Chen, Bohan Zhao, Wei Xu, Changshui Zhang, and Jian Li. Kronos: a foundation model for the language of financial markets. In *Proceedings of the Fortieth AAAI Conference on Artificial Intelligence and Thirty-Eighth Conference on Innovative Applications of Artificial Intelligence and Sixteenth Symposium on Educational Advances in Artificial Intelligence*, AAAI'26/IAAI'26/EAAI'26. AAAI Press, 2026. ISBN 978-1-57735-906-7. doi: 10.1609/aaai.v40i30.39730.

Alik Sokolov, Jonathan Mostovoy, Brydon Parker, and Luis Seco. Neural embeddings of financial time-series data. *The Journal of Financial Data Science*, 2:33–43, 10 2020. doi: 10.3905/jfds.2020.1.041.

Masashi Sugiyama. Local fisher discriminant analysis for supervised dimensionality reduction. In *Proceedings of the 23rd International Conference on Machine Learning*, ICML '06, pp. 905–912, New York, NY, USA, 2006. Association for Computing Machinery. ISBN 1595933832. doi: 10.1145/1143844.1143958.

Yanru Sun, Zongxia Xie, Dongyue Chen, Emadeldeen Eldele, and Qinghua Hu. Hierarchical classification auxiliary network for time series forecasting. In *Proceedings of the Thirty-Ninth AAAI Conference on Artificial Intelligence and Thirty-Seventh Conference on Innovative Applications of Artificial Intelligence and Fifteenth Symposium on Educational Advances in Artificial Intelligence*, AAAI'25/IAAI'25/EAAI'25. AAAI Press, 2025. ISBN 978-1-57735-897-8. doi: 10.1609/aaai.v39i19.34286.

Ankit Thakkar and Kinjal Chaudhari. A comprehensive survey on deep neural networks for stock market: The need, challenges, and future directions. *Expert Systems with Applications*, 177: 114800, 2021.

Chaojie Wang, Yuanyuan Chen, Shuqi Zhang, and Qiuhui Zhang. Stock market index prediction using deep transformer model. *Expert Systems with Applications*, 208:118128, 2022.

Yuxuan Wang, Haixu Wu, Jiaxiang Dong, Yong Liu, Mingsheng Long, and Jianmin Wang. Deep time series models: A comprehensive survey and benchmark. *IEEE transactions on pattern analysis and machine intelligence*, PP, 2024a.

Yuxuan Wang, Haixu Wu, Jiaxiang Dong, Yong Liu, Yunzhong Qiu, Haoran Zhang, Jianmin Wang, and Mingsheng Long. Timexer: Empowering transformers for time series forecasting with exogenous variables. *Advances in Neural Information Processing Systems*, 2024b.

Haixu Wu, Jiehui Xu, Jianmin Wang, and Mingsheng Long. Autoformer: Decomposition transformers with auto-correlation for long-term series forecasting. *Advances in Neural Information Processing Systems*, 34:22419–22430, 2021.

Hongjie Xia, Huijie Ao, Long Li, Yu Liu, Sen Liu, Guangnan Ye, and Hongfeng Chai. Ci-sthpan: pre-trained attention network for stock selection with channel-independent spatio-temporal hypergraph. In *Proceedings of the Thirty-Eighth AAAI Conference on Artificial Intelligence and Thirty-Sixth Conference on Innovative Applications of Artificial Intelligence and Fourteenth Symposium on Educational Advances in Artificial Intelligence*, AAAI'24/IAAI'24/EAAI'24. AAAI Press, 2024. ISBN 978-1-57735-887-9. doi: 10.1609/aaai.v38i8.28770.

Yuanjian Xu, Jianing Hao, Anxian Liu, Zhenzhuo Li, Shichang Meng, Shuai Yuan, and Guang Zhang. Lens: Large pre-trained transformer for exploring financial time series regularities. In *Proceedings of the 6th ACM International Conference on AI in Finance*, ICAIF '25, pp. 771–778, New York, NY, USA, 2025. Association for Computing Machinery. ISBN 9798400722202. doi: 10.1145/3768292.3770349.

Liang Zeng, Lei Wang, Hui Niu, Ruchen Zhang, Ling Wang, and Jian Li. Trade when opportunity comes: price movement forecasting via locality-aware attention and iterative refinement labeling. In *Proceedings of the 33rd International Joint Conference on Artificial Intelligence*, pp. 6134 – 6142, 2024.

Zhen Zeng, Rachneet Kaur, Suchetha Siddagangappa, Tucker Balch, and Manuela Veloso. From pixels to predictions: Spectrogram and vision transformer for better time series forecasting. In *Proceedings of the Fourth ACM International Conference on AI in Finance*, pp. 82–90, 2023.

Qiuyue Zhang, Yunfeng Zhang, Fangxun Bao, Yifang Liu, Caiming Zhang, and Peide Liu. Incorporating stock prices and text for stock movement prediction based on information fusion. *Engineering Applications of Artificial Intelligence*, 127:107377, 2024.

Yuechu Zheng, Yain-Whar Si, and Raymond Wong. Feature extraction for chart pattern classification in financial time series. *Knowledge and Information Systems*, 63(7):1807–1848, 2021.

Tian Zhou, Ziqing Ma, Qingsong Wen, Xue Wang, Liang Sun, and Rong Jin. Fedformer: Frequency enhanced decomposed transformer for long-term series forecasting. In *International Conference on Machine Learning*, pp. 27268–27286. PMLR, 2022.

Zhuohang Zhu, Haodong Chen, Qiang Qu, and Vera Chung. Fincast: A foundation model for financial time-series forecasting. In *Proceedings of the 34th ACM International Conference on Information and Knowledge Management*, CIKM '25, pp. 4539–4549, New York, NY, USA, 2025. Association for Computing Machinery. ISBN 9798400720406. doi: 10.1145/3746252.3761261.

A Appendix

A.1 Dataset and Event Statistics

Table 4 reports the sample counts and prevalence of upward and downward 2% events for every split. Both events are rare in the test sets: upward prevalence ranges from 3.05% to 3.62%, while downward prevalence ranges from 2.94% to 3.85% across the three settings. Their prevalence also varies substantially across indices and periods. This imbalance motivates evaluating both directions using AUROC and AP rather than relying on point forecasting errors alone.

Table 4: Dataset composition across chronological splits. Each split reports the number of 11-day samples (N) and the percentages of upward and downward 2% return events.

		Training			Validation			Test		
Setting	Index	N	$\geq +2\%$	$\leq -2\%$	N	$\geq +2\%$	$\leq -2\%$	N	$\geq +2\%$	$\leq -2\%$
2015–2019	S&P 500	2,520	4.60	5.04	1,512	3.97	4.50	1,254	1.20	2.47
	Dow Jones	2,522	4.24	4.52	1,510	3.44	3.44	1,253	1.28	2.39
	DAX	2,556	7.79	8.37	1,530	6.73	6.34	1,258	3.42	3.82
	FTSE 100	2,527	4.51	4.91	1,515	3.23	3.96	1,260	1.98	1.98
	CAC 40	2,552	6.35	7.37	1,536	6.58	6.77	1,274	3.14	3.14
	Nikkei 225	2,459	7.08	8.26	1,473	7.67	7.13	1,217	3.53	4.85
	Hang Seng	2,464	8.40	8.28	1,481	5.81	6.41	1,225	3.59	4.24
	SSE 50	2,410	9.29	8.84	1,456	6.32	5.29	1,215	6.50	5.84
	CSI 300	961	14.57	13.22	1,456	8.65	7.35	1,215	7.16	6.01
	NASDAQ	2,515	10.58	12.33	1,508	4.64	5.31	1,252	2.48	3.75
	All	23,486	7.28	7.77	14,977	5.69	5.64	12,423	3.40	3.83
2021–2025	S&P 500	2,521	3.93	4.76	1,512	2.25	3.70	1,254	2.79	3.19
	Dow Jones	2,524	3.57	4.00	1,511	2.71	3.71	1,254	1.59	1.99
	DAX	2,548	5.53	5.85	1,517	4.75	4.94	1,273	2.28	2.91
	FTSE 100	2,526	3.37	4.47	1,519	3.09	3.23	1,260	0.87	1.59
	CAC 40	2,559	5.63	6.45	1,536	4.49	4.23	1,280	2.50	2.73
	Nikkei 225	2,456	7.08	7.45	1,463	4.51	5.33	1,222	5.07	4.91
	Hang Seng	2,466	6.41	7.62	1,478	4.06	5.07	1,227	8.80	7.42
	SSE 50	2,427	9.11	8.45	1,462	6.43	5.47	1,211	2.23	2.56
	CSI 300	2,417	11.01	9.68	1,462	7.52	5.81	1,211	3.39	3.47
	NASDAQ	2,515	4.69	5.81	1,510	3.91	4.90	1,254	6.78	7.81
	All	24,959	5.99	6.43	14,970	4.36	4.63	12,446	3.62	3.85
2025–2026	S&P 500	2,519	2.90	3.73	1,510	3.31	4.17	269	2.23	2.97
	Dow Jones	2,516	2.62	3.06	1,510	2.91	3.38	269	1.12	1.86
	DAX	2,542	5.63	5.51	1,526	3.47	4.06	274	2.92	2.92
	FTSE 100	2,527	2.85	3.21	1,515	2.11	2.97	274	1.09	1.09
	CAC 40	2,560	5.43	5.43	1,538	3.64	3.90	276	2.54	1.81
	Nikkei 225	2,453	6.11	6.44	1,463	5.47	4.92	262	4.96	4.96
	Hang Seng	2,465	4.91	5.60	1,476	7.72	7.79	267	7.87	3.37
	SSE 50	2,431	6.58	5.92	1,456	3.57	2.82	263	0.38	1.14
	CSI 300	2,431	8.14	7.12	1,456	5.01	3.85	263	2.28	1.90
	NASDAQ	2,513	3.82	4.81	1,510	6.89	7.42	269	5.20	7.43
	All	24,957	4.88	5.07	14,960	4.40	4.53	2,686	3.05	2.94

A.2 Directional Anchor Selection

We select the upward and downward anchors independently using only the training split of each evaluation period. At a directional threshold δ, the upward branch separates samples with $r_i \geq \delta$ from the remaining observations, whereas the downward branch separates samples with $r_i \leq -\delta$. For each candidate anchor, we apply the cosine geometric transformation followed by PCA and evaluate directional separability using the Fisher ratio $J = S_B/S_W$.

We first perform a training-only search over the horizontal coordinate $x \in \{11, \ldots, 29\}$ together with the corresponding vertical candidate grid. Since $x = 11$ is consistently selected across evaluation periods, directions, and thresholds, we fix the horizontal coordinate at 11 in the reported experiments and search only the vertical coordinate. Table 5 reports the resulting anchors for

$\delta \in \{1\%, 2\%, 3\%, 4\%\}$. We use $\delta = 1\%$ by default because it maintains strong directional separability while providing substantially more event samples than the rarer thresholds. Anchor selection is performed exclusively on the training data, and the binary directional labels used for this search do not enter the forecasting objective.

Table 5: Training-set directional anchor selection at different return thresholds. $\mathbf{a}^+$ and $\mathbf{a}^-$ denote the independently selected upward and downward anchors, respectively, and J^+ and J^- are their binary Fisher scores. The default setting is shown in bold.

Evaluation period	Threshold δ	$\mathbf{a}^+ = (x^+, y^+)$	J^+	$\mathbf{a}^- = (x^-, y^-)$	J^-
2015–2019	**1%**	**(11, 0.929737)**	**0.217603**	**(11, 1.058732)**	**0.275633**
	2%	(11, 0.961986)	0.213592	(11, 1.042608)	0.273303
	3%	(11, 0.978110)	0.214430	(11, 1.026483)	0.265244
	4%	(11, 0.978110)	0.204611	(11, 1.026483)	0.236123
2021–2025	**1%**	**(11, 0.945862)**	**0.222178**	**(11, 1.058732)**	**0.283962**
	2%	(11, 0.961986)	0.206285	(11, 1.026483)	0.287805
	3%	(11, 0.978110)	0.205689	(11, 1.026483)	0.277943
	4%	(11, 0.978110)	0.180773	(11, 1.026483)	0.243554
2025–2026.01	**1%**	**(11, 0.964126)**	**0.282965**	**(11, 1.039325)**	**0.311334**
	2%	(11, 0.985612)	0.243914	(11, 1.017840)	0.313952
	3%	(11, 0.985612)	0.185552	(11, 1.017840)	0.275563
	4%	(11, 0.985612)	0.125897	(11, 1.007097)	0.244806

A.3 Additional Sensitivity Analysis

Target-coordinate sensitivity. We further examine whether target-state construction is sensitive to the horizontal coordinate assigned to the target close. The anchor coordinate remains fixed at $a_x = 11$, while $x_{\text{target}} \in \{-5, 1, 2, 5\}$ is varied in $\cos\left[\arctan2(a_y - \widetilde{C}_{t+1}, a_x - x_{\text{target}})\right]$. All other model components are fixed, with $x_{\text{target}} = 1$ corresponding to the complete model. Table 6 shows that the coordinate affects the resulting ranking and tail-event performance. Nevertheless, most configurations retain relatively strong AUROC, AP, and positive rank correlation across both periods, indicating that the effectiveness of the target representation is not restricted to a single coordinate choice. Among the tested settings, $x_{\text{target}} = 1$ provides the strongest overall performance, although $x_{\text{target}} = -5$ achieves a slightly higher downward AUROC in 2021–2025.

Table 6: Sensitivity to the target-close horizontal coordinate. The anchor coordinate is fixed at $a_x = 11$. Results are mean $\pm$ sample standard deviation over three seeds.

	Ranking		Upward		Downward	
x_{target}	Spearman	Kendall	AUROC	AP	AUROC	AP
Test period: 2015–2019						
1 (main)	**.0372 ± .0032**	**.0251 ± .0023**	**.8131 ± .0046**	**.1510 ± .0060**	**.7557 ± .0006**	**.1236 ± .0041**
2	.0322 ± .0081	.0215 ± .0056	.7950 ± .0161	.1447 ± .0065	.7472 ± .0058	.1212 ± .0034
5	.0360 ± .0058	.0242 ± .0043	.8014 ± .0145	.1477 ± .0052	.7453 ± .0091	.1235 ± .0100
−5	.0329 ± .0036	.0220 ± .0023	.7993 ± .0134	.1494 ± .0021	.7407 ± .0139	.1228 ± .0041
Test period: 2021–2025						
1 (main)	**.0364 ± .0036**	**.0253 ± .0024**	**.7741 ± .0036**	**.1336 ± .0057**	.7082 ± .0036	**.0900 ± .0006**
2	.0315 ± .0027	.0213 ± .0020	.7603 ± .0130	.1289 ± .0061	.7086 ± .0045	.0889 ± .0021
5	.0355 ± .0057	.0241 ± .0042	.7582 ± .0027	.1286 ± .0062	.7029 ± .0088	.0886 ± .0046
−5	.0292 ± .0079	.0196 ± .0057	.7603 ± .0234	.1303 ± .0071	**.7118 ± .0017**	.0896 ± .0029

A.4 Baseline Configuration

Learned forecasting baselines. For each index and chronological split, the baseline model receives the same ten-day normalized OHLC history, and forecasts the normalized next-day closing-price ratio $\widetilde{C}_{t+1} = C_{t+1}/C_t$.The rolling-window step is four observations, corresponding to one trading day. All learned baselines optimize MAE with batch size 32, without additional input or target scaling. Training is limited to 8,000 optimization steps, with validation loss evaluated every 100 steps. Model-specific configurations are summarized in Table 7. Each experiment is repeated with seeds 1337, 2337, and 3337 using identical data partitions.

Table 7: Hyperparameter settings of the learned baseline models.

Model	Architecture	Learning rate	Key settings
Autoformer (0.29M)	$d = 128$, E2/D1, H4	1×10^{-4}	MovingAvg $= 25$, dropout $= 0.05$
FEDformer (0.38M)	$d = 128$, E2/D1, H8	1×10^{-4}	Fourier modes $= 64$, dropout $= 0.05$
PatchTST (0.40M)	$d = 128$, E3, H16	1×10^{-4}	PatchLen $= 16$, stride $= 8$, RevIN
iTransformer (6.33M)	$d = 512$, E2/D1, H8	1×10^{-3}	FFN $= 2048$, dropout $= 0.1$
TimeXer (8.44M)	$d = 512$, E2, H8	1×10^{-3}	PatchLen $= 16$, FFN $= 2048$

The number in parentheses denotes the number of trainable parameters. d denotes the hidden dimension, E/D denote the numbers of encoder and decoder layers, respectively, and H denotes the number of attention heads.

Kronos foundation models. We additionally evaluate Kronos-small and Kronos-base using their pre-trained models and K-line tokenizer.[1] The two variants contain approximately 24.7M and 102.3M parameters, respectively. Because Kronos was pre-trained on data through 2024, we evaluate it over 2025–January 2026 to ensure that all test samples are out of sample. Daily normalized OHLC observations are processed through its forecasting pipeline, and the predicted next-day close.

A.5 Evaluation Metrics

Let r_i denote the realized next-day return, $\hat{r}_i$ a return forecast, and s_i the score used to rank observation i. For regression models, $s_i = \hat{r}_i$; for our model, $s_i = q_i^+$ when evaluating return ranking.

Point-forecasting errors. The mean squared error (MSE) and mean absolute error (MAE) are

$$\mathrm{MSE} = \frac{1}{N}\sum_{i=1}^{N}(r_i - \hat{r}_i)^2, \qquad \mathrm{MAE} = \frac{1}{N}\sum_{i=1}^{N}|r_i - \hat{r}_i|. \tag{4}$$

Return-ranking metrics. Pearson information coefficient (Pearson IC) measures linear association between realized returns and prediction scores:

$$\mathrm{IC_P} = \frac{\sum_{i=1}^{N}(r_i - \bar{r})(s_i - \bar{s})}{\sqrt{\sum_{i=1}^{N}(r_i - \bar{r})^2}\sqrt{\sum_{i=1}^{N}(s_i - \bar{s})^2}}. \tag{5}$$

Spearman IC evaluates their monotonic association and is obtained by applying Pearson correlation to the corresponding ranks:

$$\mathrm{IC_S} = \rho(\mathrm{rank}(r), \mathrm{rank}(s))\,. \tag{6}$$

Kendall's τ_b provides an additional rank-dependence measure that accounts for ties:

$$\tau_b = \frac{C - D}{\sqrt{(C + D + T_r)(C + D + T_s)}}, \tag{7}$$

where C and D are the numbers of concordant and discordant pairs, while T_r and T_s count pairs tied only in the realized returns and prediction scores, respectively. Higher Pearson IC, Spearman IC, and Kendall's τ_b indicate stronger return-ranking performance.

Directional event metrics. For a return threshold $\gamma \in \{0.02, 0.03, 0.04\}$, with $\gamma = 0.02$ used in the main experiments, the upward and downward event labels are

$$y^+_{i,\gamma} = \mathbb{I}(r_i \geq \gamma), \qquad y^-_{i,\gamma} = \mathbb{I}(r_i \leq -\gamma). \tag{8}$$

For Ours, upward metrics are computed from $(y^+_{i,\gamma}, q_i^+)$ and downward metrics from $(y^-_{i,\gamma}, q_i^-)$. For regression baselines, the corresponding scores are $\hat{r}_i$ and $-\hat{r}_i$, respectively. Thus, larger scores rank observations as more likely to exhibit the corresponding directional event.

The area under the receiver operating characteristic curve (AUROC) admits the probabilistic interpretation

$$\mathrm{AUROC} = \mathbb{P}(s^{(1)} > s^{(0)}) + \frac{1}{2}\mathbb{P}(s^{(1)} = s^{(0)}), \tag{9}$$

[1] https://github.com/shiyu-coder/Kronos

where $s^{(1)}$ and $s^{(0)}$ are the scores of randomly selected event and non-event observations. An AUROC of 0.5 corresponds to random ranking, whereas a value of 1 indicates perfect discrimination.

For a score threshold u, precision and recall are

$$\mathrm{Precision}(u) = \frac{\mathrm{TP}(u)}{\mathrm{TP}(u) + \mathrm{FP}(u)}, \qquad \mathrm{Recall}(u) = \frac{\mathrm{TP}(u)}{\mathrm{TP}(u) + \mathrm{FN}(u)}. \tag{10}$$

Average precision (AP) is the stepwise weighted average of precision over successive recall levels:

$$\mathrm{AP} = \sum_m \big(R_m - R_{m-1}\big) P_m, \tag{11}$$

where P_m and R_m denote precision and recall at the m-th score threshold. The area under the precision–recall curve (AUPRC) instead uses trapezoidal interpolation:

$$\mathrm{AUPRC} = \sum_m \frac{P_m + P_{m-1}}{2} \big(R_m - R_{m-1}\big). \tag{12}$$

AP and AUPRC summarize the same precision–recall relationship but may differ numerically because of their interpolation rules.

A.6 Decision-Level Rank Fusion

Table 8: Decision-level fusion of continuous regression forecasts with the directional scores produced by our model. The fusion weight is fixed at $\lambda = 0.50$. Results are reported as mean $\pm$ sample standard deviation over three seeds. AUROC and AP are evaluated at the 2% return threshold.

		Ranking		Upward		Downward	
Baseline	+Ours	Spearman IC	Kendall's τ	AUROC	AP	AUROC	AP
Test period: 2015–2019							
Autoformer	×	.0118 ± .0093	.0079 ± .0063	.6325 ± .0131	.1154 ± .0062	.3937 ± .0117	.0350 ± .0020
Autoformer	✓	**.0310 ± .0024**	**.0208 ± .0013**	**.7703 ± .0038**	**.1542 ± .0073**	**.6104 ± .0039**	**.0583 ± .0032**
FEDformer	×	.0018 ± .0026	.0009 ± .0018	.6135 ± .0116	.0810 ± .0040	.4317 ± .0027	.0408 ± .0012
FEDformer	✓	**.0273 ± .0026**	**.0186 ± .0016**	**.7635 ± .0063**	**.1190 ± .0039**	**.6404 ± .0024**	**.0646 ± .0016**
PatchTST	×	.0317 ± .0074	.0217 ± .0052	.5631 ± .0200	.0855 ± .0058	.4918 ± .0146	.0581 ± .0013
PatchTST	✓	**.0467 ± .0049**	**.0317 ± .0033**	**.7369 ± .0090**	**.1172 ± .0054**	**.6615 ± .0095**	**.0856 ± .0021**
iTransformer	×	.0168 ± .0084	.0114 ± .0057	.5656 ± .0495	.0864 ± .0171	.4845 ± .0290	.0549 ± .0049
iTransformer	✓	**.0352 ± .0080**	**.0235 ± .0056**	**.7376 ± .0272**	**.1230 ± .0138**	**.6637 ± .0109**	**.0842 ± .0048**
TimeXer	×	.0203 ± .0107	.0138 ± .0074	.5699 ± .0200	.0976 ± .0098	.4353 ± .0305	.0497 ± .0029
TimeXer	✓	**.0371 ± .0057**	**.0247 ± .0038**	**.7363 ± .0055**	**.1256 ± .0019**	**.6303 ± .0200**	**.0763 ± .0032**
Test period: 2021–2025							
Autoformer	×	.0065 ± .0042	.0040 ± .0029	.5898 ± .0109	.0908 ± .0114	.4344 ± .0266	.0503 ± .0022
Autoformer	✓	**.0244 ± .0028**	**.0155 ± .0018**	**.7245 ± .0026**	**.1239 ± .0052**	**.5977 ± .0123**	**.0688 ± .0038**
FEDformer	×	.0027 ± .0014	.0020 ± .0010	.5451 ± .0005	.0668 ± .0020	.4475 ± .0024	.0411 ± .0001
FEDformer	✓	**.0261 ± .0031**	**.0177 ± .0020**	**.6979 ± .0029**	**.0983 ± .0027**	**.6128 ± .0068**	**.0607 ± .0014**
PatchTST	×	.0140 ± .0064	.0094 ± .0046	.5985 ± .0133	.0982 ± .0069	.4539 ± .0080	.0428 ± .0021
PatchTST	✓	**.0315 ± .0055**	**.0215 ± .0039**	**.7240 ± .0074**	**.1323 ± .0088**	**.6111 ± .0065**	**.0665 ± .0027**
iTransformer	×	−.0008 ± .0040	−.0005 ± .0026	.5658 ± .0332	.0754 ± .0090	.4555 ± .0100	.0453 ± .0050
iTransformer	✓	**.0231 ± .0038**	**.0160 ± .0027**	**.7082 ± .0142**	**.1075 ± .0072**	**.6186 ± .0037**	**.0641 ± .0044**
TimeXer	×	.0057 ± .0016	.0037 ± .0011	.5855 ± .0212	.0867 ± .0051	.4193 ± .0142	.0367 ± .0025
TimeXer	✓	**.0263 ± .0018**	**.0177 ± .0010**	**.7178 ± .0110**	**.1190 ± .0094**	**.5904 ± .0097**	**.0557 ± .0016**

A.7 Complete Forecasting Results

For the complete numerical results corresponding to Figure 5, we report the mean and sample standard deviation over three random seeds in Table 9. Statistical significance in this table is evaluated using the seed-averaged predictions, rather than by treating the three seeds as the statistical sample. Specifically, we apply a two-sided, market-stratified moving-block bootstrap with 1,000 resamples and a block length of 20 trading days to observations matched by market and prediction date. For each metric, significance against all baselines is assessed using an intersection–union test, and the resulting eight metric-level p-values within each test period are adjusted using the Holm procedure. A superscript * in Table 9 indicates that Ours significantly outperforms every baseline for the corresponding metric after correction ($p < 0.05$).

Table 9: Complete main results corresponding to Fig. 5.

		Return ranking		Upward tail			Downward tail		
Test period	Method	Spearman IC	Kendall's τ	AUROC	AP	AUPRC	AUROC	AP	AUPRC
2015–2019	Autoformer	.0118±.0093	.0079±.0063	.6325±.0131	.1154±.0062	.1139±.0060	.3937±.0117	.0350±.0020	.0345±.0019
	FEDformer	.0018±.0026	.0009±.0018	.6135±.0116	.0810±.0040	.0801±.0039	.4317±.0027	.0408±.0012	.0402±.0011
	PatchTST	.0317±.0074	.0217±.0052	.5631±.0200	.0855±.0058	.0842±.0060	.4918±.0146	.0581±.0013	.0573±.0013
	iTransformer	.0168±.0084	.0114±.0057	.5656±.0495	.0864±.0171	.0852±.0173	.4845±.0290	.0549±.0049	.0537±.0049
	TimeXer	.0203±.0107	.0138±.0074	.5699±.0200	.0976±.0098	.0964±.0099	.4353±.0305	.0497±.0029	.0488±.0028
	Ours	**.0372±.0032**	**.0251±.0023**	**.8131±.0046***	**.1510±.0060**	**.1497±.0061**	**.7557±.0006***	**.1236±.0041***	**.1223±.0043***
2021–2025	Autoformer	.0065±.0042	.0040±.0029	.5898±.0109	.0908±.0114	.0896±.0118	.4344±.0266	.0503±.0022	.0499±.0021
	FEDformer	.0027±.0014	.0020±.0010	.5451±.0005	.0668±.0020	.0656±.0021	.4475±.0024	.0411±.0001	.0405±.0001
	PatchTST	.0140±.0064	.0094±.0046	.5985±.0133	.0982±.0069	.0970±.0068	.4539±.0080	.0428±.0021	.0422±.0022
	iTransformer	−.0008±.0040	−.0005±.0026	.5658±.0332	.0754±.0090	.0744±.0093	.4555±.0100	.0453±.0050	.0446±.0050
	TimeXer	.0057±.0016	.0037±.0011	.5855±.0212	.0867±.0051	.0857±.0050	.4193±.0142	.0367±.0025	.0361±.0026
	Ours	**.0364±.0036**	**.0253±.0024**	**.7741±.0036***	**.1336±.0057***	**.1323±.0055***	**.7082±.0036***	**.0900±.0006***	**.0892±.0007***
2025–Jan. 2026	Kronos-small	.0153±.0236	.0108±.0166	.5229±.0089	.0880±.0025	.0846±.0030	.3728±.0250	.0376±.0080	.0352±.0083
	Kronos-base	.0067±.0147	.0043±.0107	.5252±.0193	**.0930±.0265**	**.0880±.0277**	.4051±.0152	.0387±.0069	.0356±.0071
	Ours	**.0511±.0070**	**.0346±.0049**	**.7517±.0302***	.0864±.0141	.0838±.0133	**.6449±.0085***	**.0484±.0036**	**.0462±.0031**

B Theoretical Analysis

This section provides a theoretical interpretation of three empirical observations: the discrepancy between point accuracy and tail ranking, the effect of directional state prediction, and the role of directional anchors in adaptive target quantization. The analysis does not assume that discretization creates additional information. Instead, it characterizes how the proposed output representation changes the statistical objective and the geometry used for finite-state quantization. Let X denote the historical market sequence and R the future return. For a threshold $\tau > 0$, define the upward and downward events as

$$Y_\tau^+ = \mathbb{I}(R \geq \tau), \qquad Y_\tau^- = \mathbb{I}(R \leq -\tau). \tag{13}$$

The analysis below applies to either branch $b \in \{+, -\}$.

B.1 Point Regression and Tail Ranking

Definition B.1 (Point and tail targets)**.** *For an input x, define the conditional mean, conditional median, and directional event posterior by*

$$\mu(x) = \mathbb{E}[R \mid X = x], \quad m(x) \in \mathrm{Median}(R \mid X = x), \quad \eta_\tau^b(x) = \mathbb{P}(Y_\tau^b = 1 \mid X = x). \tag{14}$$

Proposition B.2 (Different losses recover different targets)**.** *At the population optimum, squared-error regression recovers $\mu(x)$, absolute-error regression recovers $m(x)$, and binary cross-entropy for Y_τ^b recovers $\eta_\tau^b(x)$. Moreover, ranking observations by $\eta_\tau^b(x)$, or by any strictly increasing transformation of it, is Bayes optimal for the population AUROC of event Y_τ^b.*

Proof. The first two statements follow by minimizing the corresponding conditional risks. Binary cross-entropy is a strictly proper scoring rule and is uniquely minimized by the conditional event probability. At any fixed false-positive rate, ranking observations by $\eta_\tau^b(x)$ selects those with the largest conditional event probabilities and therefore maximizes the true-positive rate. Hence, $\eta_\tau^b(x)$ induces a Bayes-optimal ROC ranking. □

Proposition B.2 shows that point prediction and directional ranking generally target different conditional functionals. A model may therefore attain the minimum possible MSE without producing the best ranking of extreme movements.

Proposition B.3 (Point optimality does not imply tail optimality)**.** *There exist return distributions for which the Bayes-optimal squared-error predictor reverses the Bayes-optimal tail-event ranking.*

Proof. Let $X \in \{x_1, x_2\}$ with equal probability and set $\tau = 0.8$. Suppose

$$R \mid X = x_1 = \begin{cases} 1, & \text{with probability } 0.1, \\ 0, & \text{with probability } 0.9, \end{cases} \qquad R \mid X = x_2 = 0.2.$$

The conditional mean and upward-event posterior are therefore

$$\mu(x_1) = 0.1 < 0.2 = \mu(x_2), \qquad \eta_\tau^+(x_1) = 0.1 > 0 = \eta_\tau^+(x_2).$$

Hence, the minimum-MSE predictor ranks x_2 above x_1, whereas the event posterior ranks x_1 above x_2. To compute the corresponding AUROCs, represent the distribution by 200 equally weighted observations. Of the 100 observations with $X = x_1$, 10 are positive tail events and 90 are negative; all 100 observations with $X = x_2$ are negative. There are thus 10 positive and 190 negative observations, producing $10 \times 190 = 1900$ positive–negative pairs. Each correctly ordered pair contributes one, each tie contributes one half, and each incorrectly ordered pair contributes zero.

Under the conditional-mean score, every positive observation has score $\mu(x_1) = 0.1$. Its comparisons with the 90 negative observations from x_1 produce $10 \times 90 = 900$ ties, while its comparisons with the 100 negative observations from x_2, whose score is $\mu(x_2) = 0.2$, produce $10 \times 100 = 1000$ incorrectly ordered pairs. Therefore,

$$\mathrm{AUC}(\mu) = \frac{\frac{1}{2}(900)}{1900} = \frac{9}{38}.$$

Under the event-posterior score, observations from x_1 have score 0.1 and those from x_2 have score 0. The same 900 within-x_1 comparisons remain ties, but all 1000 comparisons against negative observations from x_2 are now correctly ordered. Consequently,

$$\mathrm{AUC}(\eta_\tau^+) = \frac{\frac{1}{2}(900) + 1000}{1900} = \frac{29}{38}.$$

Thus, even a Bayes-optimal point predictor can be suboptimal for tail-event ranking, so optimal point accuracy provides no general guarantee of optimal tail discrimination. □

This mismatch explains why lower MSE or MAE need not correspond to higher AUROC, AP, or rank correlation in financial forecasting.

B.2 State Probabilities as Directional Ranking Scores

Let $Z^b \in \{1, \ldots, K\}$ denote the codebook state assigned to branch $b \in \{+, -\}$, and let

$$p_k^b(x) = \mathbb{P}(Z^b = k \mid X = x)$$

be its conditional state probability. Let $\mathcal{S}^b$ contain selected extreme states: the states with the highest mean returns for the upward branch and those with the lowest mean returns for the downward branch.

Definition B.4 (Aggregated state score)**.** *The directional score associated with $\mathcal{S}^b$ is*

$$q^b(x) = \sum_{k \in \mathcal{S}^b} p_k^b(x). \tag{15}$$

Theorem B.5 (Posterior interpretation of state aggregation)**.** *Assume that the classification head reaches the population minimum of its cross-entropy risk. Then*

$$q^b(x) = \mathbb{P}\big(Z^b \in \mathcal{S}^b \mid X = x\big)\,. \tag{16}$$

Consequently, q^b is Bayes optimal for ranking the codebook-induced event $\{Z^b \in \mathcal{S}^b\}$.
For the return-threshold event Y_τ^b, define

$$\rho_{\tau,k}^b(x) = \mathbb{P}\big(Y_\tau^b = 1 \mid Z^b = k, X = x\big)\,, \qquad s_k^b = \mathbb{I}(k \in \mathcal{S}^b). \tag{17}$$

The difference between the aggregated state score and the threshold-event posterior satisfies

$$\big|q^b(x) - \eta_\tau^b(x)\big| \le \sum_{k=1}^{K} p_k^b(x)\,\big|s_k^b - \rho_{\tau,k}^b(x)\big|\,. \tag{18}$$

Proof. Cross-entropy minimization recovers the conditional state probabilities,

$$p_k^b(x) = \mathbb{P}(Z^b = k \mid X = x).$$

Therefore, summing over the selected states gives

$$q^b(x) = \sum_{k \in \mathcal{S}^b} p_k^b(x)$$
$$= \mathbb{P}(Z^b \in \mathcal{S}^b \mid X = x).$$

This identity holds for any selected state set, irrespective of its cardinality or empirical frequency.

To relate this score to the threshold event, the law of total probability gives

$$\eta_\tau^b(x) = \sum_{k=1}^{K} p_k^b(x) \rho_{\tau,k}^b(x).$$

Since $s_k^b = \mathbb{I}(k \in \mathcal{S}^b)$, the directional score can also be written as

$$q^b(x) = \sum_{k=1}^{K} p_k^b(x) s_k^b.$$

Hence,

$$q^b(x) - \eta_\tau^b(x) = \sum_{k=1}^{K} p_k^b(x) \left(s_k^b - \rho_{\tau,k}^b(x) \right).$$

Taking absolute values and applying the triangle inequality yields

$$\left| q^b(x) - \eta_\tau^b(x) \right| \leq \sum_{k=1}^{K} p_k^b(x) \left| s_k^b - \rho_{\tau,k}^b(x) \right|,$$

where $p_k^b(x) \geq 0$. This proves Eq. 18. □

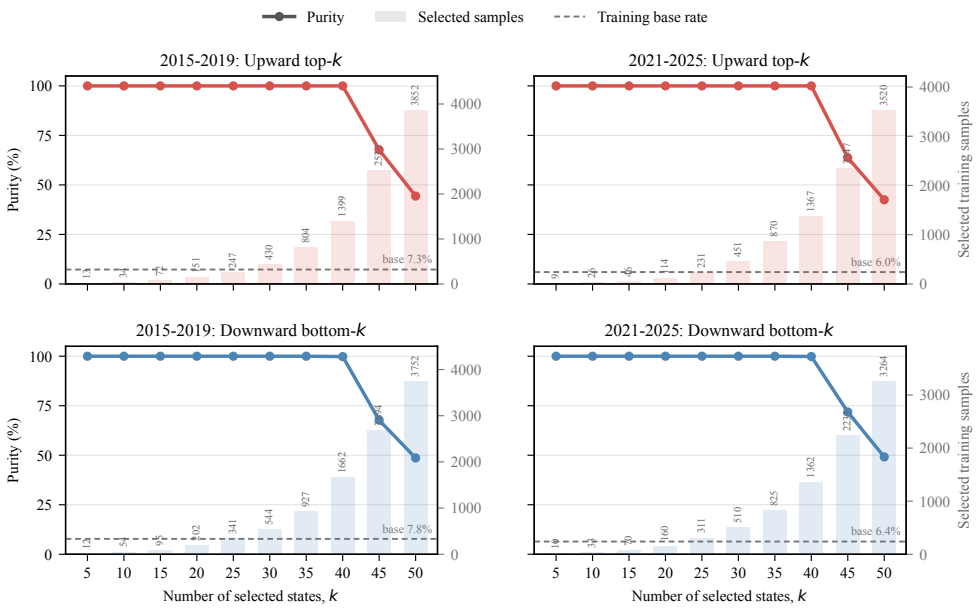

Figure 9: Purity and sample size of the selected directional states in the training data. Lines report the proportion satisfying the corresponding 2% return event, bars report the number of selected samples, and dashed lines indicate the unconditional event rates.

Empirical state concentration. The training-set statistics confirm that the codebook ordering isolates highly concentrated directional tails. For the 1999–2008 training period, upward and downward 2% events account for 7.28% and 7.77% of all observations, respectively. Nevertheless, all

13 observations assigned to the upward top-five states satisfy $R \geq 0.02$, and all 12 observations in the downward bottom-five states satisfy $R \leq -0.02$. For the 2005–2014 training period, the corresponding unconditional rates are 5.99% and 6.43%, while the selected top-five and bottom-five states again attain 100% purity, containing 9 upward and 10 downward observations.

This concentration is not restricted to the five most extreme states. As shown in Fig. 9, purity remains 100% for the upward top-40 states in both periods, covering 1,399 and 1,367 training observations. The downward bottom-40 states contain 1,662 and 1,362 observations and retain purities of 99.82% and 99.85%, respectively. Purity decreases markedly only when the selected set expands beyond 40 states. The ordering induced by the directional codebooks therefore concentrates the corresponding return tails consistently rather than only isolating a few individual observations.

Eq. 18 separates two sources of discrepancy from the threshold-event posterior. For selected states ($s_k^b = 1$), the term $1 - \rho_{\tau,k}^b(x)$ measures contamination by non-tail observations. For unselected states ($s_k^b = 0$), the term $\rho_{\tau,k}^b(x)$ measures threshold events omitted from the selected set. The observed purity directly supports the first component by showing that the selected states are directionally consistent. The top-five score used in the main experiments deliberately represents a more concentrated event than the complete 2% tail. Accordingly, $q^b(x)$ is interpreted as the posterior probability of entering the most extreme directional states, rather than as a calibrated probability of crossing the 2% threshold.

B.3 Directional Anchors as Adaptive Target Quantization

We analyze how the directional anchor changes the geometry of the scalar target space before target-state quantization. Let y denote a normalized target closing-price ratio and let $d = a_x - x_{\mathrm{tar}} > 0$, where x_{tar} is the fixed horizontal coordinate used for the target transformation. The cosine angular mapping is

$$T_a(y) = \cos[\arctan2(a_y - y, d)] = \frac{d}{\sqrt{d^2 + (a_y - y)^2}}. \tag{19}$$

Proposition B.6 (Anchor-induced local metric)**.** *For a sufficiently small perturbation δ,*

$$[T_a(y + \delta) - T_a(y)]^2 = w_a(y)\delta^2 + O(|\delta|^3), \tag{20}$$

where

$$w_a(y) = \left[\frac{d(a_y - y)}{(d^2 + (a_y - y)^2)^{3/2}}\right]^2. \tag{21}$$

Proof. Differentiating Eq. 19 gives

$$T_a'(y) = \frac{d(a_y - y)}{(d^2 + (a_y - y)^2)^{3/2}}. \tag{22}$$

A first-order Taylor expansion yields

$$T_a(y + \delta) - T_a(y) = T_a'(y)\delta + O(\delta^2). \tag{23}$$

Squaring both sides gives Eq. 20, with $w_a(y) = [T_a'(y)]^2$ as defined in Eq. 21. □

Proposition B.6 shows that Euclidean quantization after the cosine transformation is locally equivalent to a non-uniformly weighted quantization of the original scalar target space. The induced weight varies with the target value and the anchor position. Consequently, independently selected upward and downward anchors define different local geometries and can allocate finite codebook resolution differently across the two directional target spaces.

The transformation is locally informative on regions where $y \neq a_y$. Moreover, because $T_a(y)$ is symmetric around a_y, it is not globally injective. In our framework, the upward and downward anchors are used as directional target transformations rather than as a single global coordinate system. The relevant operating regions therefore lie on the corresponding monotonic sides of the selected anchors, under which the transformation preserves local ordering.