algorithms

MDPI

Article

FreqCast: Frequency-Decoupled Statistical and Deep Learning for Multihorizon Return Forecasting and Price Reconstruction

Yu Lu and Haibin Zhang *

School of Finance and Economics, Jiangsu University, Zhenjiang 212013, China; 3230812041@stmail.ujs.edu.cn
* Correspondence: jghb@ujs.edu.cn

Abstract

This study forecasts cumulative log returns at horizons of one to twenty trading days and reconstructs future adjusted prices by exponentiating those return forecasts; it does not optimize a price-level loss. This task is difficult because financial returns are nonstationary, heavy-tailed, horizon-dependent, and subject to rapidly changing volatility. We propose FreqCast, which combines a market-conditioned spectral decomposition, a structured state-space branch for the component designated low-frequency, causal multiscale encoders for the components designated intermediate- and high-frequency, and a horizon-conditioned reliability gate. The gate uses expert representations, predictive scale, and cross-expert disagreement to fuse four cumulative-return estimates. The joint objective covers point loss, an auxiliary directional score, Laplace likelihood, ordered quantile loss, decomposition regularization, and horizon coherence. Experiments use eight large U.S. stocks and a single 2021–2024 test interval. Within that restricted benchmark, the reported point estimates favor FreqCast over the included baselines and show horizon- and volatility-dependent expert allocation. Our main contribution is the coordinated frequency-dependent assignment and reliability fusion of heterogeneous forecasting mechanisms, while broad market robustness and statistical superiority remain to be established.

Keywords: cumulative log-return forecasting; price reconstruction; multihorizon forecasting; adaptive spectral decomposition; state-space modeling; deep learning; prediction intervals

1. Introduction

This study concerns multihorizon cumulative log-return forecasting, followed by deterministic price reconstruction. Let p_t be the adjusted close and $z_t = \log p_t$. At origin t, the optimized target is $r_{t,h} = z_{t+h} - z_t$ for $h = 1, \ldots, H$; the corresponding price reconstruction is $\widehat{p}_{t+h} = p_t \exp(\widehat{r}_{t,h})$. This distinction matters because the loss and reported MAE/RMSE are defined on returns, not on price levels. Forecasts may support market monitoring and risk analysis, but their investment value is not evaluated here.

The classical forecasting framework specifies conditional location and scale from past information. Autoregressive integrated moving-average (ARIMA) models describe linear conditional means, whereas autoregressive conditional heteroscedasticity (ARCH) and generalized ARCH (GARCH) models describe volatility clustering [1–3]. State-space models express these dynamics through a latent transition and observation equation, giving compact, interpretable multistep propagation. Their fixed functional forms, however, can be restrictive under nonlinear interactions and regime changes. Hybrid financial models therefore combine filtering, statistical dynamics, and nonlinear learners [4–6]; this literature frames the statistical–deep balance considered here.

Check for updates

Academic Editor: Takeshi Yamada

Received: 25 July 2026
Revised: 24 August 2026
Accepted: 1 September 2026
Published: 4 September 2026

Copyright: © 2026 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license.

Deep models relax predetermined dynamics. Long short-term memory (LSTM) networks use gated recurrence [7]; temporal convolutional networks (TCNs) use causal convolutions [8]; and Transformer variants use attention, decomposition, or frequency-domain operations [9–12]. Patch-based, periodic, variable-centric, and multiscale architectures include PatchTST, TimesNet, iTransformer, and TimeMixer [13–16]. Three linked concerns remain for the present financial setting.

First, a financial window contains energy at several temporal frequencies, but the economic labels "trend," "cycle," and "shock" do not follow automatically from a frequency cutoff. Their meaning must be treated as an empirical interpretation. Fixed spectral bands can also be poorly aligned with a changing spectrum. Second, one shared encoder need not be equally suitable for smooth and rapidly varying components: structured state propagation and nonlinear representation learning impose different inductive biases. Third, the relevance and uncertainty of a component can change with both market state and forecast horizon, so fixed averaging can be inappropriate.

FreqCast addresses these concerns through adaptive spectral decomposition and heterogeneous forecast routing. FreqCast is the model name rather than an acronym. Two ordered, sample-conditioned boundaries generate differentiable low-, intermediate-, and high-frequency masks whose preliminary components sum exactly to the normalized input. A structured state-space branch processes the component designated low-frequency; separate causal multiscale encoders process the other two components; and a joint nonlinear expert represents their interaction. The labels describe relative spectral position only. Claims that a component represents a trend, cycle, or shock are made only when supported by the diagnostics in Section 5.3.

Each expert predicts the same supervised quantity, namely the cumulative log return from the common origin t to horizon h. The statistical expert obtains it from the propagated low-frequency state; the intermediate and high experts obtain it from their respective pooled encodings; and the joint deep expert obtains it from their interaction representation. These are learned end-to-end against $r_{t,h}$ and are not returns downloaded from an external source. For each origin and horizon, the reliability gate combines their estimates as

$$\widehat{r}_{t,h} = \sum_{e \in \mathcal{E}} \alpha_{t,h}^{(e)} \widehat{r}_{t,h}^{(e)}, \qquad \sum_{e \in \mathcal{E}} \alpha_{t,h}^{(e)} = 1, \tag{1}$$

where $\mathcal{E} = \{\text{stat}, \text{mid}, \text{high}, \text{deep}\}$ and $\alpha_{t,h}^{(e)}$ depends on the expert representation, horizon, market state, predictive scale, and disagreement. The principal methodological contribution is this coordinated chain—sample-conditioned spectral routing, heterogeneous statistical/deep experts, and reliability-based multihorizon fusion—rather than any one standard building block in isolation.

This study asks three questions: (RQ1) do adaptive boundaries provide useful information beyond fixed Fourier bands within this architecture; (RQ2) do the statistical and frequency-specific deep branches contribute differently across horizons; and (RQ3) does the learned gate allocate experts differently across horizons and volatility regimes? In this paper we provide descriptive evidence for these questions. They do not establish universal economic meanings for the bands.

The contributions map directly to these questions and to the preceding gaps: (1) an adaptive, complete spectral partition for RQ1; (2) frequency-dependent assignment of a bounded state-space model and causal nonlinear encoders for RQ2; (3) a horizon-conditioned gate using scale and disagreement for RQ3; and (4) a joint return-forecasting objective with Laplace scale, ordered quantiles, decomposition regularization, and increment-based horizon coherence. Table 1 distinguishes this coordination from closely related methods.

The empirical study uses eight large U.S. stocks and one contiguous 2021–2024 test period. The reported differences are point estimates for this benchmark, not evidence of broad financial-market robustness or statistical superiority. Section 5.6 states the limitations created by the restricted asset universe, single test period, incomplete baseline provenance, and absence of origin-level prediction and computational logs.

The remainder of this paper is organized as follows. Section 2 integrates the background and related forecasting literature. Section 3 defines the return target, price reconstruction, notation, and frequency representation. Section 4 presents adaptive decomposition, the statistical and deep branches, fusion, and optimization. Section 5 describes the protocol and descriptive results.

2. Background and Related Work

2.1. Financial Time-Series and Multihorizon Forecasting

Classical financial forecasting is largely based on explicit assumptions about temporal dependence, stationarity, and conditional variance. Autoregressive integrated moving-average models describe a time series through lagged observations and innovations and remain standard statistical baselines for financial prediction [1]. Because stock returns exhibit volatility clustering and time-varying uncertainty, autoregressive conditional heteroscedasticity was introduced to model conditional variance as a function of past squared innovations [2]. The generalized ARCH formulation further allows current volatility to depend on both previous shocks and previous conditional variances [3]. These statistical models provide interpretable parameters and stable extrapolation, but their fixed linear structures are often insufficient for abrupt regime changes, nonlinear feature interactions, and horizon-dependent market behavior.

Deep learning has subsequently been applied to financial prediction to reduce reliance on manually specified dynamics. Stacked autoencoders and LSTM networks learn nonlinear financial representations [17,18], while multi-frequency feature extraction separates patterns at several temporal scales [19]. Hybrid LSTM–GARCH models combine nonlinear sequence learning with explicit volatility estimation [4]. Recent filtering–combination systems likewise separate nonlinear trends and stochastic residuals before combining autoregressive, neural, and conditional-variance forecasts [5]; local mean decomposition (LMD) has also been coupled with ARIMA and machine learning predictors for financial series [6]. These studies motivate heterogeneous modeling, but their filters or component-to-model assignments are specified before end-to-end multihorizon training.

Probabilistic and multihorizon forecasting provides a broader view than single-step point prediction. DeepAR is a deep autoregressive recurrent model for conditional predictive distributions [20]. Neural Basis Expansion Analysis for Time Series (N-BEATS) uses backward and forward residual blocks to learn trend/seasonal bases without recurrence [21]. The Temporal Fusion Transformer combines recurrent local processing, attention, static covariates, and quantile outputs [22]. These names denote different representation and output choices; they are introduced here because they motivate recurrent distributional forecasts, interpretable bases, and horizon-conditioned quantiles, respectively.

FreqCast follows this hybrid statistical–deep direction but differs in its coordination mechanism. Instead of feeding an unresolved sequence to complete parallel predictors or fixing the decomposition before training, it learns sample-dependent soft spectral boundaries, routes the resulting components to different predictor classes, and fuses cumulative-return estimates using horizon, scale, and disagreement. This is an architectural distinction; whether it improves forecasting is an empirical question confined here to the stated benchmark.

2.2. Deep, Multiscale, and Frequency-Aware Time-Series Models

Recurrent neural networks form an early foundation for nonlinear sequence forecasting. LSTM introduces gated memory cells that mitigate vanishing gradients and preserve long-range sequential information [7]. Temporal convolutional networks provide an alternative based on dilated causal convolutions and residual connections, allowing long receptive fields with parallel computation [8]. Transformer models replace recurrent transitions with self-attention and have become a major architecture for sequence representation [9]. Informer reduces the computational cost of long-sequence forecasting through sparse attention and a generative decoder [10], while Autoformer incorporates decomposition and autocorrelation mechanisms for long-term prediction [11]. FEDformer performs decomposition and representation learning in the frequency domain to improve efficiency and capture global temporal patterns [12].

Recent work has shown that complex attention architectures are not always necessary for strong forecasting performance. DLinear applies decomposition followed by simple linear projections and demonstrates the competitiveness of direct linear forecasting [23]. PatchTST divides each univariate series into temporal patches, increasing the receptive field while preserving channel-independent representations [13]. TimesNet maps one-dimensional sequences into two-dimensional temporal representations according to detected periods and applies convolutional processing to model intraperiod and interperiod variation [14]. These methods reveal the value of decomposition, patching, and periodic representation, but they generally apply one principal representation mechanism to all temporal components.

The 2024 forecasting literature further emphasizes variable-centric, multiscale, and adaptive representations. iTransformer treats complete variate histories as tokens and performs attention across variables rather than across individual time points [15]. TimeMixer decomposes multiscale sequences and separately mixes seasonal and trend information before combining forecasts from multiple temporal scales [16]. Pathformer learns adaptive pathways through multiscale Transformer components to select appropriate temporal resolutions for each input [24]. CATS constructs auxiliary time series to represent intervariable relationships and injects them as exogenous information into a lightweight predictor [25]. TimeXer extends Transformer forecasting to exogenous variables through a global endogenous token and cross-attention between endogenous and exogenous representations [26]. These models improve scale selection or cross-variable interaction, whereas FreqCast additionally assigns distinct statistical and deep inductive biases to different learned frequency ranges.

Frequency-domain modeling has received renewed attention because periodic patterns and spectral energy are not represented uniformly in the time domain. FEDformer uses Fourier or wavelet blocks within a decomposed Transformer [12]. TimeMixer combines decomposed information across downsampled temporal scales [16]. Amplifier identifies the tendency of forecasting models to neglect low-energy frequency components and explicitly amplifies them during representation learning [27]. WaveletMixer uses multiresolution wavelet representations and mixing operations to capture localized temporal patterns at different scales [28]. Collectively, these works demonstrate that frequency energy and resolution materially affect forecasting quality.

FreqCast is related to these frequency-aware approaches but addresses a different modeling question. Most existing methods use frequency transforms to improve a shared neural representation or to mix information across resolutions. FreqCast instead learns two ordered boundaries that define low-, intermediate-, and high-frequency components for each historical window. The masks satisfy a reconstruction constraint, and the components are assigned to forecasting mechanisms with different extrapolation properties. The

low-frequency component is modeled through a market-conditioned statistical transition process, while the remaining components use nonlinear causal encoders. The model therefore performs frequency-dependent representation selection rather than frequency-domain feature extraction alone.

Another distinction concerns multihorizon fusion. TimeMixer combines predictors associated with downsampled inputs [16], and TimeMixer++ extends mixing across time-domain scales and frequency resolutions [29]. FreqCast instead treats a structured state-space predictor and three nonlinear component predictors as heterogeneous experts, then evaluates them separately for every horizon. Predictive scale and disagreement enter the gate rather than serving only as output diagnostics. Table 1 summarizes the closest methodological contrasts and identifies the coordinated routing-and-fusion mechanism as the primary contribution.

Table 1. Structured comparison with closely related decomposition, frequency-aware, multiscale, and hybrid approaches. "Adaptive" refers to input-window-dependent boundaries learned jointly with the forecasting loss.

Approach	Decomposition or Scale Mechanism	Component Predictor Assignment	Distinction Relative to FreqCast
FEDformer [12]	Trend–seasonal decomposition with Fourier/wavelet blocks	Frequency-enhanced Transformer representation	No learned three-band routing to heterogeneous statistical/deep experts
TimeMixer [16]	Downsampled multiscale seasonal–trend mixing	Multiscale mixing and multiple predictors	No sample-conditioned Fourier boundaries or uncertainty/disagreement gate
WaveletMixer [28]	Wavelet multiresolution analysis	MLP mixing across wavelet features	Localized fixed-basis resolution rather than learned statistical/deep routing
Filtering hybrid [5]	Regression and smoothing filters	Nonlinear/NN autoregression plus ARCH residual modeling	Predefined filtering and forecast combination, without an end-to-end horizon gate
LMD hybrid [6]	Local mean decomposition and signal grouping	ARIMA for one group; machine learning for another	Algorithmic decomposition and staged reconstruction rather than learned soft boundaries
FreqCast	Sample-conditioned, complete soft Fourier partition	State-space low-band expert; mid/high causal encoders; joint deep expert	Joint component routing and horizon-conditioned reliability fusion using scale and disagreement

2.3. Pretrained and Foundation Models for Time Series

Large pretrained models have recently expanded time-series forecasting beyond dataset-specific supervised learning. Time-LLM reprograms numerical time-series inputs into representations compatible with frozen language models and incorporates textual prompts describing temporal characteristics [30]. MOMENT pretrains a family of open time-series foundation models for forecasting, classification, anomaly detection, and imputation [31]. TimesFM develops a decoder-only foundation model that produces zero-shot forecasts through patch-based autoregressive decoding [32]. Moirai introduces a masked encoder-based universal forecasting Transformer trained on a large archive spanning multiple domains and frequencies [33]. Timer formulates forecasting, imputation, and anomaly detection as generative tasks and pretrains a GPT-style model on heterogeneous time-series sequences [34]. TimeSiam uses Siamese pretraining to learn representations from tempo-

rally related views of unlabeled sequences [35]. Chronos quantizes scaled time-series values into discrete tokens and trains language-model architectures to generate probabilistic future sequences [36].

The foundation-model direction continued in 2025 with larger contexts, sparse expert architectures, and broader evaluation. Timer-XL extends generative time-series modeling to longer contexts and unifies forecasting across different sequence configurations [37]. Time-MoE scales time-series foundation models using a sparse mixture-of-experts architecture [38]. TimeMixer++ develops a general pattern machine that extracts representations across time scales and frequency resolutions [29]. Sundial trains large probabilistic forecasters for flexible zero-shot prediction [39], while GIFT-Eval provides a standardized benchmark for comparing statistical, supervised deep-learning, and pretrained forecasting models across heterogeneous datasets [40]. These developments demonstrate the value of large-scale pretraining, although their principal objective is cross-domain generality rather than specialized modeling of a single nonstationary financial asset.

Large language models have also been adapted through explicit cross-modal alignment. CALF aligns temporal and textual branches through cross-modal matching, feature regularization, and output consistency [41]. TimeCMA extracts time-series and prompt-based language-model representations and aligns the two modalities for multivariate forecasting [42]. Such methods can incorporate broad pretrained knowledge, but the numerical behavior of stock prices is strongly influenced by local regimes, volatility changes, and asset-specific dynamics. A large generic representation does not by itself guarantee that persistent and transient market components will be extrapolated using suitable inductive biases.

FreqCast is complementary to foundation-model research. It uses no external pretraining or textual market description. Its proposed contribution is the coordination of statistical and deep representations after adaptive spectral decomposition. The structured state, component-specific estimates, and reliability weights are observable diagnostics, but they constitute architectural transparency rather than proof that a component has a universal economic interpretation.

The literature establishes complementary tools for conditional-variance modeling, nonlinear encoding, decomposition, multiscale learning, and pretraining. Table 1 shows that several methods share individual elements with FreqCast. The narrower gap addressed here is their joint coordination: window-dependent spectral boundaries, explicit routing to heterogeneous predictor classes, and horizon-specific fusion using scale and forecast disagreement. This statement defines the proposed mechanism; it does not imply that every possible decomposition or hybrid comparator is evaluated.

3. Preliminaries

3.1. Cumulative-Return Forecasting and Price Reconstruction

Consider a stock observed at equally spaced trading times. At time t, the market observation is $\mathbf{x}_t \in \mathbb{R}^D$. A look-back window of length L gives $\mathbf{X}_t = [\mathbf{x}_{t-L+1}, \ldots, \mathbf{x}_t]^\top \in \mathbb{R}^{L \times D}$. Let $p_t > 0$ denote the adjusted close and $z_t = \log p_t$. The supervised target is the cumulative log return $r_{t,h} = z_{t+h} - z_t$ for $h \in \mathcal{H} = \{1, \ldots, H\}$. The multihorizon task learns a parameterized mapping

$$[\widehat{r}_{t,1}, \ldots, \widehat{r}_{t,H}] = \mathcal{F}_\Theta(\mathbf{X}_t), \tag{2}$$

where Θ contains all trainable parameters. Price is reconstructed only after inference, as follows:

$$\widehat{p}_{t+h} = p_t \exp(\widehat{r}_{t,h}), \qquad \frac{\widehat{p}_{t+h}}{p_{t+h}} = \exp(\widehat{r}_{t,h} - r_{t,h}). \tag{3}$$

Thus, an additive error in the cumulative log return becomes a multiplicative price-reconstruction error. The objective below is not a price-level objective, and the empirical tables report return error in the basis points. Joint forecasting shares information across horizons and permits a coherence penalty, while horizon embeddings allow the relevant component mixture to change with h. For N valid forecasting origins, the basic point objective is

$$\mathcal{L}_{\text{pred}} = \frac{1}{NH} \sum_{t=1}^{N} \sum_{h=1}^{H} \ell(r_{t,h}, \widehat{r}_{t,h}), \tag{4}$$

where $\ell(\cdot, \cdot)$ is the Huber loss. The direction is $d_{t,h} = \mathbb{I}(r_{t,h} > 0)$ and supplies only an auxiliary target; it does not replace the continuous return objective. Table 2 consolidates the principal notation used throughout this paper.

Table 2. Principal notation. Superscript (e) indexes forecasting experts and superscript (b) indexes spectral components.

Symbol	Definition
p_t, z_t	Adjusted closing price and its logarithm at origin t
$r_{t,h}, \widehat{r}_{t,h}$	Observed and predicted cumulative log return from t to $t+h$
$\widehat{p}_{t+h}$	Price reconstructed from $p_t \exp(\widehat{r}_{t,h})$
$\mathbf{X}_t \in \mathbb{R}^{L \times D}$	Input window with look-back L and D channels
$\mathcal{H}, H$	Forecast-horizon set and maximum horizon
$\mathcal{B}$	{low, mid, high}, denoting relative spectral position
$c_{t,1}, c_{t,2}, \tau_t$	Ordered frequency boundaries and soft-mask temperature
$\mathbf{g}_t, \mathbf{e}_h$	Market-state representation and horizon embedding
$\mathcal{E}$	{stat, mid, high, deep} forecasting experts
$\widehat{r}^{(e)}_{t,h}, s^{(e)}_{t,h}$	Expert cumulative-return location and positive Laplace scale
$\alpha^{(e)}_{t,h}$	Non-negative reliability weight, summing to one over experts

3.2. Frequency-Domain Representation of Financial Sequences

A financial sequence contains energy at different frequencies. Low, intermediate, and high refer here to relative spectral position; they are not, by definition, economic trends, cycles, or shocks. Those interpretations require component diagnostics. For a normalized univariate sequence $\mathbf{s}_t = [s_{t-L+1}, \ldots, s_t]^\top$, the discrete Fourier transform is

$$S_t(k) = \sum_{n=0}^{L-1} s_{t-L+1+n} \exp\left(-\mathrm{i}\frac{2\pi k n}{L}\right), \qquad k = 0, \ldots, L-1, \tag{5}$$

where $\mathrm{i}^2 = -1$. Its inverse transform reconstructs the time sequence as

$$s_{t-L+1+n} = \frac{1}{L} \sum_{k=0}^{L-1} S_t(k) \exp\left(\mathrm{i}\frac{2\pi k n}{L}\right). \tag{6}$$

A generic decomposition applies frequency masks $M_b(k)$ to form $S_t^{(b)}(k) = M_b(k) S_t(k)$. To preserve the full signal, the masks should satisfy $\sum_b M_b(k) = 1$ for every k. For the preliminary inverse transforms used by FreqCast, completeness gives the exact identity

$$\mathbf{s}_t = \bar{\mathbf{s}}_t^{\text{low}} + \bar{\mathbf{s}}_t^{\text{mid}} + \bar{\mathbf{s}}_t^{\text{high}}. \tag{7}$$

The overbar denotes the unrefined inverse transform. The later residual refiners can disturb this identity, so their outputs are tied back to the input by the reconstruction loss in Equation (24). Applying the operation to all D channels yields component tensors.

FreqCast replaces fixed binary masks with learnable soft assignments because the spectral energy distribution can differ across windows.

The decomposition is a routing device rather than a complete forecasting model. A discrete Fourier transform (DFT) represents window-global periodic content and has limited time localization; the residual refiners and causal encoders address local departures. This motivates the approximation used after refinement and also limits the economic interpretation of a learned band.

3.3. Statistical and Deep Temporal Representations

Statistical models represent explicit temporal dependence. For a component u_t, an autoregressive model of order P is $u_t = c + \sum_{j=1}^{P} \phi_j u_{t-j} + \varepsilon_t$, where c is an intercept, ϕ_j is a lag coefficient, and ε_t is an innovation. State-space formulations introduce a latent state $\mathbf{a}_t$ with transition $\mathbf{a}_t = \mathbf{T}\mathbf{a}_{t-1} + \boldsymbol{\eta}_t$ and observation $u_t = \mathbf{q}^\top \mathbf{a}_t + \epsilon_t$. These representations are suitable for low-frequency sequences because persistent patterns often admit compact linear or locally linear descriptions. A deep temporal encoder instead learns a nonlinear mapping

$$\mathbf{H}_t^{(b)} = \mathcal{E}_b(\mathbf{X}_t^{(b)}; \Theta_b), \tag{8}$$

where $\mathcal{E}_b$ is the encoder for component b. Causal convolutions, attention, and gated transformations can capture local patterns, long-range dependencies, and nonlinear cross-feature interactions. Let $\mathbf{h}_t^{(b)}$ summarize $\mathbf{H}_t^{(b)}$ and let $\mathbf{e}_h$ be a horizon embedding. A component forecast can be written as $\widehat{r}_{t,h}^{(b)} = \mathcal{G}_b(\mathbf{h}_t^{(b)}, \mathbf{e}_h)$.

The final prediction may then be formed as

$$\widehat{r}_{t,h} = \sum_{b=1}^{B} \alpha_{t,h}^{(b)} \widehat{r}_{t,h}^{(b)}, \qquad \sum_{b=1}^{B} \alpha_{t,h}^{(b)} = 1, \tag{9}$$

where $\alpha_{t,h}^{(b)} \geq 0$ measures the reliability of component b at time t and horizon h. FreqCast develops this formulation by assigning persistent information to a structured statistical branch and nonlinear transient information to deep encoders, followed by dynamic reliability fusion.

4. Methodology

FreqCast combines four stages: adaptive spectral decomposition, structured state propagation, causal nonlinear encoding, and reliability fusion. The motivation is a division of inductive bias rather than a claim that frequency alone determines predictability. Given $\mathbf{X}_t$, the model creates three relative spectral components, routes them to a state-space branch or causal multiscale encoders, and combines four cumulative-return experts using the current state and horizon. Algorithms 1–5 provide the computation order; the following subsections define each mechanism.

4.1. Adaptive Spectral Decomposition

This module is motivated by the variation in window-level spectral energy. Fixed cutoffs route the same normalized frequency to the same branch regardless of the observed state. FreqCast instead learns sample-dependent boundaries while retaining only an ordinal low/intermediate/high interpretation; the module does not impose economic labels on the bands.

Each input channel is normalized within the current historical window. For channel d, the window statistics are

$$\mu_{t,d} = \frac{1}{L}\sum_{\ell=1}^{L} x_{t,\ell,d}, \qquad \sigma_{t,d} = \sqrt{\frac{1}{L}\sum_{\ell=1}^{L}(x_{t,\ell,d} - \mu_{t,d})^2}, \tag{10}$$

and the normalized input is

$$\widetilde{x}_{t,\ell,d} = \frac{x_{t,\ell,d} - \mu_{t,d}}{\sigma_{t,d} + \epsilon_{\mathrm{n}}}, \tag{11}$$

where $\epsilon_{\mathrm{n}} > 0$ ensures numerical stability. The scale vectors $\boldsymbol{\mu}_t$ and $\boldsymbol{\sigma}_t$ are retained as market-state features.

Algorithm 1 Adaptive Spectral Decomposition

Input: Window $\mathbf{X}_t \in \mathbb{R}^{L\times D}$

Output: Components $\{\mathbf{X}_t^{(b)}\}_{b\in\mathcal{B}}$, state $\mathbf{g}_t$, and loss $\mathcal{L}_{\mathrm{dec},t}$

Normalize each channel of $\mathbf{X}_t$ using its window mean $\boldsymbol{\mu}_t$ and scale $\boldsymbol{\sigma}_t$
Compute $\mathbf{Z}_t \leftarrow \mathrm{DFT}_L(\widetilde{\mathbf{X}}_t)$ and normalized spectral energy $e_t(k)$
Summarize the spectrum into $\mathbf{v}_t$
Construct $\mathbf{u}_t = [\mathbf{v}_t; \log(\sigma_t + \epsilon_{\mathrm{n}}); \rho_t; \nu_t]$
Compute market state $\mathbf{g}_t \leftarrow \mathrm{MLP}_{\mathrm{state}}(\mathbf{u}_t)$
Infer ordered cutoffs $c_{t,1}$ and $c_{t,2}$ and temperature τ_t from $\mathbf{g}_t$
for $k = 0$ to $L-1$ **do**
    Compute $M_t^{(\mathrm{low})}(k)$, $M_t^{(\mathrm{mid})}(k)$, and $M_t^{(\mathrm{high})}(k)$
end for
for all $b \in \mathcal{B} = \{\mathrm{low}, \mathrm{mid}, \mathrm{high}\}$ **do**
    $\mathbf{Z}_t^{(b)} \leftarrow \mathbf{M}_t^{(b)} \odot \mathbf{Z}_t$
    $\overline{\mathbf{X}}_t^{(b)} \leftarrow \mathrm{Re}[\mathrm{IDFT}_L(\mathbf{Z}_t^{(b)})]$
    $\mathbf{X}_t^{(b)} \leftarrow \overline{\mathbf{X}}_t^{(b)} + \mathcal{R}_b(\overline{\mathbf{X}}_t^{(b)})$
end for
Compute $\mathcal{L}_{\mathrm{dec},t} = \lambda_{\mathrm{rec}}\mathcal{L}_{\mathrm{rec},t} + \lambda_{\mathrm{ovl}}\mathcal{L}_{\mathrm{ovl},t} + \lambda_{\mathrm{bd}}\mathcal{L}_{\mathrm{bd},t}$
return $\{\mathbf{X}_t^{(b)}\}_{b\in\mathcal{B}}, \mathbf{g}_t, \mathcal{L}_{\mathrm{dec},t}$

Algorithm 2 Horizon-Conditioned Statistical Representation Learning

Input: Low-frequency component $\mathbf{X}_t^{(\mathrm{low})}$, state $\mathbf{g}_t$, horizon H

Output: Statistical outputs $\{\widehat{r}_{t,h}^{(\mathrm{stat})}, s_{t,h}^{(\mathrm{stat})}, \mathbf{h}_{t,h}^{(\mathrm{stat})}\}_{h=1}^{H}$

Project the low-frequency component into $y_{t,\ell}^{(\mathrm{low})}$
Infer initial state $\mathbf{a}_{t,0}$ from $\mathbf{X}_t^{(\mathrm{low})}$ and $\mathbf{g}_t$
Construct the state-dependent transition matrix $\mathbf{T}_t$ and innovation gain $\mathbf{K}_t$
for $\ell = 1$ to L **do**
    Compute innovation $\varepsilon_{t,\ell} = y_{t,\ell}^{(\mathrm{low})} - \mathbf{q}^\top \mathbf{a}_{t,\ell-1}$
    Robustify the innovation to obtain $\bar{\varepsilon}_{t,\ell}$
    $\mathbf{a}_{t,\ell} \leftarrow \mathbf{T}_t \mathbf{a}_{t,\ell-1} + \mathbf{K}_t \bar{\varepsilon}_{t,\ell}$
end for
for $h = 1$ to H **do**
    $\widehat{\mathbf{a}}_{t,h} \leftarrow \mathbf{T}_t^h \mathbf{a}_{t,L}$
    $\mathbf{h}_{t,h}^{(\mathrm{stat})} \leftarrow \mathrm{MLP}_{\mathrm{stat}}([\widehat{\mathbf{a}}_{t,h}; \mathbf{a}_{t,L}; \mathbf{e}_h; \mathbf{g}_t])$
    Predict $\widehat{r}_{t,h}^{(\mathrm{stat})}$ and positive scale $s_{t,h}^{(\mathrm{stat})}$
end for
Compute $\mathcal{L}_{\text{stat-reg},t}$ from state reconstruction, likelihood, and uncertainty monotonicity
return statistical outputs and $\mathcal{L}_{\text{stat-reg},t}$

Algorithm 3 Causal Multiscale Deep Representation Learning

Input: $\mathbf{X}_t^{(\mathrm{mid})}$, $\mathbf{X}_t^{(\mathrm{high})}$, state $\mathbf{g}_t$, horizon H
Output: Outputs of the mid-, high-, and deep experts

for all $b \in \{\mathrm{mid}, \mathrm{high}\}$ **do**
  Project $\mathbf{X}_t^{(b)}$ and add temporal embeddings
  for $m = 1$ to M **do**
    Apply gated causal convolutions at scales $s = 1, \ldots, S$
    Compute state-conditioned scale weights $\beta_{t,m,s}^{(b)}$
    Aggregate scales and apply residual channel mixing
  end for
end for
Apply bidirectional causal cross-frequency attention between the mid- and high-frequency representations
Incorporate transferred features using state-conditioned gates
Apply recency-aware attention pooling to obtain $\mathbf{c}_t^{(\mathrm{mid})}$ and $\mathbf{c}_t^{(\mathrm{high})}$
Extract shock descriptor $\mathbf{o}_t$
for $h = 1$ to H **do**
  Construct horizon-conditioned representations $\mathbf{h}_{t,h}^{(\mathrm{mid})}$ and $\mathbf{h}_{t,h}^{(\mathrm{high})}$
  Predict $(\widehat{r}_{t,h}^{(\mathrm{mid})}, s_{t,h}^{(\mathrm{mid})})$ and $(\widehat{r}_{t,h}^{(\mathrm{high})}, s_{t,h}^{(\mathrm{high})})$
  $\mathbf{h}_{t,h}^{(\mathrm{deep})} \leftarrow \mathrm{MLP}_{\mathrm{deep}}([\mathbf{c}_t^{(\mathrm{mid})}; \mathbf{c}_t^{(\mathrm{high})}; \mathbf{c}_t^{(\mathrm{mid})} \odot \mathbf{c}_t^{(\mathrm{high})}; |\mathbf{c}_t^{(\mathrm{mid})} - \mathbf{c}_t^{(\mathrm{high})}|; \mathbf{o}_t; \mathbf{g}_t; \mathbf{e}_h])$
  Predict $(\widehat{r}_{t,h}^{(\mathrm{deep})}, s_{t,h}^{(\mathrm{deep})})$
end for
Compute $\mathcal{L}_{\text{deep-reg},t}$ from likelihood, diversity, and component consistency
return expert outputs and $\mathcal{L}_{\text{deep-reg},t}$

The Fourier transform is applied along the temporal dimension, as follows:

$$\mathbf{Z}_t = \mathrm{DFT}_L(\widetilde{\mathbf{X}}_t) \in \mathbb{C}^{L \times D}. \tag{12}$$

The normalized non-negative frequency associated with index k is $\omega_k = 2\min(k, L-k)/L$. The normalized spectral energy is

$$e_t(k) = \frac{\sum_{d=1}^{D} |Z_{t,k,d}|^2}{\sum_{j=0}^{L-1} \sum_{d=1}^{D} |Z_{t,j,d}|^2 + \epsilon_{\mathrm{n}}}. \tag{13}$$

The spectrum is summarized into Q intervals, giving $\mathbf{v}_t = [v_{t,1}, \ldots, v_{t,Q}]^\top$. Together with log scale, lag-one correlation ρ_t, and realized volatility ν_t, it forms

$$\mathbf{u}_t = [\mathbf{v}_t; \log(\sigma_t + \epsilon_{\mathrm{n}}); \rho_t; \nu_t], \qquad \mathbf{g}_t = \mathrm{MLP}_{\mathrm{state}}(\mathbf{u}_t), \tag{14}$$

where $\mathbf{g}_t \in \mathbb{R}^{D_g}$ represents the current market state. The following two ordered boundaries divide the normalized spectrum:

$$c_{t,1} = c_{\min} + (c_{\max} - c_{\min}) \,\mathrm{sigmoid}(a_1 + \mathbf{w}_1^\top \mathbf{g}_t), \tag{15}$$

$$c_{t,2} = c_{t,1} + (1 - c_{t,1}) \,\mathrm{sigmoid}(a_2 + \mathbf{w}_2^\top \mathbf{g}_t). \tag{16}$$

This parameterization guarantees $0 < c_{t,1} < c_{t,2} < 1$. A positive state-dependent temperature is

$$\tau_t = \tau_{\min} + \mathrm{softplus}(a_\tau + \mathbf{w}_\tau^\top \mathbf{g}_t). \tag{17}$$

The three masks are

$$M_t^{\text{low}}(k) = 1 - \text{sigmoid}\left(\frac{\omega_k - c_{t,1}}{\tau_t}\right), \tag{18}$$

$$M_t^{\text{high}}(k) = \text{sigmoid}\left(\frac{\omega_k - c_{t,2}}{\tau_t}\right), \tag{19}$$

$$M_t^{\text{mid}}(k) = 1 - M_t^{\text{low}}(k) - M_t^{\text{high}}(k). \tag{20}$$

They satisfy

$$M_t^{\text{low}}(k) + M_t^{\text{mid}}(k) + M_t^{\text{high}}(k) = 1 \tag{21}$$

for every k, ensuring that the complete spectrum is distributed rather than discarded.

Algorithm 4 Cross-Frequency Reliability Fusion

Input: Expert outputs $\{\widehat{r}_{t,h}^{(e)}, s_{t,h}^{(e)}, \mathbf{h}_{t,h}^{(e)}\}_{e\in\mathcal{E}}$
Output: $\widehat{r}_{t,h}$, $s_{t,h}^{(\text{fuse})}$, and $\{\alpha_{t,h}^{(e)}\}_{e\in\mathcal{E}}$

$\bar{r}_{t,h} \leftarrow \frac{1}{|\mathcal{E}|}\sum_{e\in\mathcal{E}} \widehat{r}_{t,h}^{(e)}$
for all $e \in \mathcal{E}$ **do**
  $d_{t,h}^{(e)} \leftarrow |\widehat{r}_{t,h}^{(e)} - \bar{r}_{t,h}|$
  Compute reliability score $a_{t,h}^{(e)}$ from $\mathbf{h}_{t,h}^{(e)}$, $\mathbf{g}_t$, $\mathbf{e}_h$, $s_{t,h}^{(e)}$, and $d_{t,h}^{(e)}$
end for
$\boldsymbol{\alpha}_{t,h} \leftarrow \text{softmax}(\mathbf{a}_{t,h}/\tau_g)$
$\widehat{r}_{t,h} \leftarrow \sum_{e\in\mathcal{E}} \alpha_{t,h}^{(e)} \widehat{r}_{t,h}^{(e)}$
$(s_{t,h}^{(\text{fuse})})^2 \leftarrow \sum_{e\in\mathcal{E}} \alpha_{t,h}^{(e)} \left[(s_{t,h}^{(e)})^2 + (\widehat{r}_{t,h}^{(e)} - \widehat{r}_{t,h})^2\right]$
Compute auxiliary directional score $\widehat{a}_{t,h}^{(\text{up})}$
Generate ordered quantiles around $\widehat{r}_{t,h}$ using positive increments
return fused outputs and reliability weights

For $b \in \mathcal{B} = \{\text{low}, \text{mid}, \text{high}\}$, the masked spectrum and preliminary temporal component are

$$Z_{t,k,d}^{(b)} = M_t^{(b)}(k) Z_{t,k,d}, \qquad \overline{\mathbf{X}}_t^{(b)} = \text{Re}[\text{IDFT}_L(\mathbf{Z}_t^{(b)})]. \tag{22}$$

A lightweight residual refiner handles local nonstationarity as follows:

$$\mathbf{X}_t^{(b)} = \overline{\mathbf{X}}_t^{(b)} + \mathcal{R}_b(\overline{\mathbf{X}}_t^{(b)}), \tag{23}$$

where $\mathcal{R}_b$ consists of gated depthwise causal convolutions specialized for each component.

The decomposition is regularized by reconstruction, overlap, and boundary stability. The reconstruction term is

$$\mathcal{L}_{\text{rec}} = \frac{1}{NLD}\sum_{t=1}^{N} \left\| \widetilde{\mathbf{X}}_t - \sum_{b\in\mathcal{B}} \mathbf{X}_t^{(b)} \right\|_F^2. \tag{24}$$

The energy-weighted overlap penalty is

$$\mathcal{L}_{\text{ovl}} = \frac{1}{N}\sum_{t=1}^{N}\sum_{k=0}^{L-1} e_t(k) \sum_{b<b'} M_t^{(b)}(k) M_t^{(b')}(k). \tag{25}$$

For adjacent forecasting origins, boundary stability is encouraged through

$$\mathcal{L}_{\mathrm{bd}} = \frac{1}{N-1}\sum_{t=2}^{N} \kappa_t[(c_{t,1}-c_{t-1,1})^2 + (c_{t,2}-c_{t-1,2})^2], \tag{26}$$

where $\kappa_t = \exp(-\|\mathbf{g}_t - \mathbf{g}_{t-1}\|_2^2)$ relaxes the penalty when the market state changes abruptly. The complete decomposition objective is

$$\mathcal{L}_{\mathrm{dec}} = \lambda_{\mathrm{rec}}\mathcal{L}_{\mathrm{rec}} + \lambda_{\mathrm{ovl}}\mathcal{L}_{\mathrm{ovl}} + \lambda_{\mathrm{bd}}\mathcal{L}_{\mathrm{bd}}. \tag{27}$$

Algorithm 5 Joint Training of FreqCast

Input: Training set $\{(\mathbf{X}_t, \mathbf{r}_t)\}_{t=1}^{N}$, horizon H
Output: Trained parameters Θ

Initialize Θ, optimization step $u \leftarrow 0$, and a high gate temperature τ_{g}
for each mini-batch $\mathcal{D}$ **do**
  $\mathcal{L}_{\mathrm{batch}} \leftarrow 0$
  for all $t \in \mathcal{D}$ **do**
    Apply Algorithm 1
    Apply Algorithm 2
    Apply Algorithm 3
    for $h = 1$ to H **do**
      Apply Algorithm 4
    end for
    Compute prediction, likelihood, direction, quantile, horizon, and gate-smoothness losses
    $\lambda_{\mathrm{ent}}(u) \leftarrow \lambda_{\mathrm{ent}}^{0} \max\left(0, 1 - \frac{u}{U_{\mathrm{a}}}\right)$
    Compute gate-entropy loss $\mathcal{L}_{\mathrm{ent},t}$
    $\mathcal{L}_t \leftarrow \mathcal{L}_{\mathrm{pred},t} + \lambda_{\mathrm{nll}}\mathcal{L}_{\mathrm{nll},t} + \lambda_{\mathrm{dir}}\mathcal{L}_{\mathrm{dir},t} + \lambda_{\mathrm{quant}}\mathcal{L}_{\mathrm{quant},t}$
      $+ \lambda_{\mathrm{hor}}\mathcal{L}_{\mathrm{hor},t} + \lambda_{\mathrm{gate}}\mathcal{L}_{\mathrm{gate},t} + \lambda_{\mathrm{ent}}(u)\mathcal{L}_{\mathrm{ent},t}$
      $+ \mathcal{L}_{\mathrm{dec},t} + \mathcal{L}_{\mathrm{stat\text{-}reg},t} + \mathcal{L}_{\mathrm{deep\text{-}reg},t}$
    $\mathcal{L}_{\mathrm{batch}} \leftarrow \mathcal{L}_{\mathrm{batch}} + \mathcal{L}_t$
  end for
  Update Θ using the average mini-batch loss
  Anneal τ_{g} after the warm-up stage
  $u \leftarrow u + 1$
end for
At inference, recover prices as $\widehat{p}_{t+h} = p_t \exp(\widehat{r}_{t,h})$
return Θ

4.2. Horizon-Conditioned Statistical Representation Learning

The statistical branch is used because a low-dimensional transition gives bounded, inspectable multistep propagation. The component routed to it can still exhibit continuation, reversal, and changing dispersion, so its transition parameters are amortized from the current window rather than fixed globally. This design motivation does not presume that every low-frequency coefficient is predictable. A learned projection extracts a scalar persistent signal $y_{t,\ell}^{\mathrm{low}} = \mathbf{w}_y^{\top}\mathbf{x}_{t,\ell}^{\mathrm{low}} + b_y$. The statistical state is

$$\mathbf{a}_{t,\ell} = [m_{t,\ell}, s_{t,\ell}, q_{t,\ell}^{(1)}, \widetilde{q}_{t,\ell}^{(1)}, \ldots, q_{t,\ell}^{(J)}, \widetilde{q}_{t,\ell}^{(J)}]^{\top}, \tag{28}$$

where $m_{t,\ell}$ is the local level, $s_{t,\ell}$ is the local slope, and each pair $(q^{(j)}, \widetilde{q}^{(j)})$ represents a damped oscillatory mode. The state update and innovation are

$$\mathbf{a}_{t,\ell} = \mathbf{T}_t\mathbf{a}_{t,\ell-1} + \mathbf{K}_t\bar{\varepsilon}_{t,\ell}, \qquad \varepsilon_{t,\ell} = y_{t,\ell}^{\mathrm{low}} - \mathbf{q}^{\top}\mathbf{a}_{t,\ell-1}. \tag{29}$$

The robust innovation is $\bar{\varepsilon}_{t,\ell} = \delta_t \tanh(\varepsilon_{t,\ell}/(\delta_t + \epsilon_n))$, with $\delta_t = \delta_{\min} + \mathrm{softplus}(b_\delta + \mathbf{w}_\delta^\top \mathbf{g}_t)$.

The trend block and cyclic blocks are

$$\mathbf{T}_t^{\mathrm{tr}} = \begin{bmatrix} 1 & \gamma_t \\ 0 & \rho_t^{\mathrm{tr}} \end{bmatrix}, \qquad \mathbf{T}_{t,j}^{\mathrm{cyc}} = \rho_{t,j}^{\mathrm{cyc}} \begin{bmatrix} \cos\varphi_{t,j} & -\sin\varphi_{t,j} \\ \sin\varphi_{t,j} & \cos\varphi_{t,j} \end{bmatrix}. \tag{30}$$

The complete transition is

$$\mathbf{T}_t = \mathrm{BlockDiag}(\mathbf{T}_t^{\mathrm{tr}}, \mathbf{T}_{t,1}^{\mathrm{cyc}}, \ldots, \mathbf{T}_{t,J}^{\mathrm{cyc}}). \tag{31}$$

The following state-conditioned bounded parameterizations are used: $\gamma_t = \mathrm{sigmoid}(b_\gamma + \mathbf{w}_\gamma^\top \mathbf{g}_t)$; $\rho_t^{\mathrm{tr}} = \tanh(b_{\mathrm{tr}} + \mathbf{w}_{\mathrm{tr}}^\top \mathbf{g}_t)$; $\rho_{t,j}^{\mathrm{cyc}} = \mathrm{sigmoid}(b_{\rho,j} + \mathbf{w}_{\rho,j}^\top \mathbf{g}_t)$; and $\varphi_{t,j} = \pi\,\mathrm{sigmoid}(b_{\varphi,j} + \mathbf{w}_{\varphi,j}^\top \mathbf{g}_t)$. The adaptive innovation gain is $\mathbf{K}_t = \mathrm{sigmoid}(\mathbf{W}_K \mathbf{g}_t + \mathbf{b}_K)$.

The initial state is inferred from the mean and final value of the low-frequency window and the market-state vector. After filtering the observed sequence, the future state is propagated as

$$\hat{\mathbf{a}}_{t,h} = \mathbf{T}_t^h \mathbf{a}_{t,L}, \qquad h = 1, \ldots, H. \tag{32}$$

A horizon-conditioned representation compares the propagated and current states as follows:

$$\mathbf{h}_{t,h}^{\mathrm{stat}} = \mathrm{MLP}_{\mathrm{stat}}([\hat{\mathbf{a}}_{t,h}; \mathbf{a}_{t,L}; \mathbf{e}_h; \mathbf{g}_t]), \tag{33}$$

with return and uncertainty heads

$$\hat{r}_{t,h}^{\mathrm{stat}} = \mathbf{w}_r^\top \mathbf{h}_{t,h}^{\mathrm{stat}} + b_r, \qquad s_{t,h}^{\mathrm{stat}} = s_{\min} + \mathrm{softplus}(\mathbf{w}_s^\top \mathbf{h}_{t,h}^{\mathrm{stat}} + b_s). \tag{34}$$

Here, $s_{t,h}^{\mathrm{stat}}$ denotes the predictive standard deviation. When the likelihood is written as a Laplace distribution, its scale parameter is $b_{t,h}^{\mathrm{stat}} = s_{t,h}^{\mathrm{stat}}/\sqrt{2}$ because a Laplace variable with scale b has variance $2b^2$. This convention makes the mixture-variance calculation in Equation (47) dimensionally consistent. Uncertainty monotonicity is softly encouraged by

$$\mathcal{L}_{\mathrm{unc}} = \frac{1}{N(H-1)} \sum_{t=1}^{N} \sum_{h=1}^{H-1} \max(0, s_{t,h}^{\mathrm{stat}} - s_{t,h+1}^{\mathrm{stat}}). \tag{35}$$

The branch additionally minimizes one-step state reconstruction and a heteroscedastic Laplace likelihood. Their weighted sum is denoted by $\mathcal{L}_{\text{stat-reg}}$.

4.3. Causal Multiscale Deep Representation Learning

The two higher-frequency components are assigned separate causal encoders because localized nonlinear patterns can require several receptive fields, while causal processing preserves the information set at origin t. Cross-frequency attention is applied only after separate encoding so that interaction is possible without erasing component identity. For $b \in \{\mathrm{mid}, \mathrm{high}\}$, the projected input is $\widetilde{\mathbf{H}}_{t,0}^{(b)} = \mathbf{X}_t^{(b)} \mathbf{W}_{\mathrm{in}}^{(b)} + \mathbf{P}$, where $\mathbf{P}$ is a learnable temporal embedding. At layer m and scale s, a causal convolution produces

$$\mathbf{U}_{t,m,s}^{(b)} = \mathrm{CConv}_{k_s,d_s}^{(b,m)}(\mathrm{LN}(\mathbf{H}_{t,m-1}^{(b)})). \tag{36}$$

Splitting $\mathbf{U}$ into content and gate tensors gives $\mathbf{V}_{t,m,s}^{(b)} = \mathrm{GELU}(\mathbf{A}_{t,m,s}^{(b)}) \odot \mathrm{sigmoid}(\mathbf{G}_{t,m,s}^{(b)})$. The market-conditioned scale weights are

$$\beta_{t,m,s}^{(b)} = \frac{\exp(a_{m,s}^{(b)} + (\mathbf{w}_{m,s}^{(b)})^\top \mathbf{g}_t)}{\sum_{s'=1}^{S} \exp(a_{m,s'}^{(b)} + (\mathbf{w}_{m,s'}^{(b)})^\top \mathbf{g}_t)}, \tag{37}$$

leading to $\mathbf{V}_{t,m}^{(b)} = \sum_s \beta_{t,m,s}^{(b)} \mathbf{V}_{t,m,s}^{(b)}$. Residual and channel-mixing transformations update the representation as

$$\mathbf{H}_{t,m}^{(b)} = \overline{\mathbf{H}}_{t,m}^{(b)} + \mathrm{MLP}_m^{(b)}(\mathrm{LN}(\overline{\mathbf{H}}_{t,m}^{(b)})). \tag{38}$$

Cross-frequency attention captures events expressed jointly across the intermediate and high components. With a causal mask $\mathbf{C}$,

$$\mathbf{A}_t^{\mathrm{mid}\leftarrow\mathrm{high}} = \mathrm{softmax}\left(\frac{\mathbf{Q}_t^{\mathrm{mid}}(\mathbf{K}_t^{\mathrm{high}})^\top}{\sqrt{D_a}} + \mathbf{C}\right)\mathbf{V}_t^{\mathrm{high}}, \tag{39}$$

with a symmetric expression for $\mathbf{A}_t^{\mathrm{high}\leftarrow\mathrm{mid}}$. State-conditioned gates selectively incorporate the transferred features as follows:

$$\widetilde{\mathbf{H}}_t^{(b)} = \mathbf{H}_t^{(b)} + \mathbf{\Gamma}_t^{(b)} \odot \mathbf{A}_t^{(b\leftarrow b')}, \tag{40}$$

where b' denotes the other nonlinear frequency component.

Recency-aware attention pooling forms component summaries $\mathbf{c}_t^{(b)} = \sum_{\ell=1}^{L} \pi_{t,\ell}^{(b)} \widetilde{\mathbf{h}}_{t,\ell}^{(b)}$, where

$$\pi_{t,\ell}^{(b)} = \frac{\exp(\xi_{t,\ell}^{(b)} - \zeta_b(L-\ell))}{\sum_{j=1}^{L}\exp(\xi_{t,j}^{(b)} - \zeta_b(L-j))}. \tag{41}$$

A shock descriptor $\mathbf{o}_t$ summarizes recent high-frequency magnitude, peak intensity, and temporal changes. The joint nonlinear representation is

$$\mathbf{h}_{t,h}^{\mathrm{deep}} = \mathrm{MLP}_{\mathrm{deep}}([\mathbf{c}_t^{\mathrm{mid}}; \mathbf{c}_t^{\mathrm{high}}; \mathbf{c}_t^{\mathrm{mid}} \odot \mathbf{c}_t^{\mathrm{high}}; |\mathbf{c}_t^{\mathrm{mid}} - \mathbf{c}_t^{\mathrm{high}}|; \mathbf{o}_t; \mathbf{g}_t; \mathbf{e}_h]). \tag{42}$$

The branch produces separate $\widehat{r}_{t,h}^{\mathrm{mid}}$ and $\widehat{r}_{t,h}^{\mathrm{high}}$ forecasts and a joint estimate

$$\widehat{r}_{t,h}^{\mathrm{deep}} = \mathbf{w}_r^{\mathrm{deep}\top}\mathbf{h}_{t,h}^{\mathrm{deep}} + b_r^{\mathrm{deep}}, \qquad s_{t,h}^{\mathrm{deep}} = s_{\min} + \mathrm{softplus}(\mathbf{w}_s^{\mathrm{deep}\top}\mathbf{h}_{t,h}^{\mathrm{deep}} + b_s^{\mathrm{deep}}). \tag{43}$$

Representation diversity discourages collapse between normalized intermediate- and high-frequency summaries, while a component-consistency loss keeps the joint deep forecast connected to the two component-wise forecasts. Together with the deep likelihood, these terms form $\mathcal{L}_{\text{deep-reg}}$.

4.4. Cross-Frequency Reliability Fusion and Joint Optimization

Fusion is required because no component is assumed to be uniformly reliable across origins or horizons. Let $\mathcal{E} = \{\mathrm{stat}, \mathrm{mid}, \mathrm{high}, \mathrm{deep}\}$ denote the four experts. Each produces a cumulative-return location $\widehat{r}_{t,h}^{(e)}$, a representation $\mathbf{h}_{t,h}^{(e)}$, and a positive predictive standard deviation $s_{t,h}^{(e)}$; the corresponding Laplace scale is $b_{t,h}^{(e)} = s_{t,h}^{(e)}/\sqrt{2}$. The intermediate- and high-frequency scales are predicted from their horizon-conditioned representations in the same manner as the statistical and joint-deep scales.

The consensus forecast and expert disagreement are

$$\bar{r}_{t,h} = \frac{1}{|\mathcal{E}|}\sum_{e\in\mathcal{E}} \widehat{r}_{t,h}^{(e)}, \qquad d_{t,h}^{(e)} = |\widehat{r}_{t,h}^{(e)} - \bar{r}_{t,h}|. \tag{44}$$

The unnormalized reliability score is

$$\begin{aligned} a_{t,h}^{(e)} = {} & \mathbf{v}_e^\top \tanh(\mathbf{W}_e\mathbf{h}_{t,h}^{(e)} + \mathbf{U}_e\mathbf{g}_t + \mathbf{R}_e\mathbf{e}_h + \mathbf{b}_e) \\ & - \eta_e \log(s_{t,h}^{(e)} + \epsilon_{\mathrm{n}}) - \chi_e d_{t,h}^{(e)}, \end{aligned} \tag{45}$$

where $\eta_e, \chi_e \geq 0$ is learned. The reliability weight is

$$\alpha_{t,h}^{(e)} = \frac{\exp(a_{t,h}^{(e)}/\tau_g)}{\sum_{e' \in \mathcal{E}} \exp(a_{t,h}^{(e')}/\tau_g)}, \tag{46}$$

and the final forecast follows Equation (1).

The fused variance includes both within-expert uncertainty and between-expert disagreement, as follows:

$$(s_{t,h}^{\text{fuse}})^2 = \sum_{e \in \mathcal{E}} \alpha_{t,h}^{(e)} [(s_{t,h}^{(e)})^2 + (\hat{r}_{t,h}^{(e)} - \hat{r}_{t,h})^2]. \tag{47}$$

The point objective uses Huber loss. The heteroscedastic likelihood is parameterized as $R_{t,h} \mid \mathbf{X}_t \sim \text{Laplace}(\hat{r}_{t,h}, b_{t,h}^{\text{fuse}})$ with $b_{t,h}^{\text{fuse}} = s_{t,h}^{\text{fuse}}/\sqrt{2}$. The sigmoid quantity used by the directional loss is

$$\hat{a}_{t,h}^{\text{up}} = \text{sigmoid}\left(\frac{\hat{r}_{t,h}}{s_{t,h}^{\text{fuse}} + \epsilon_n}\right), \tag{48}$$

and binary cross-entropy between this score and $\mathbb{I}(r_{t,h} > 0)$ gives $\mathcal{L}_{\text{dir}}$. This is an auxiliary directional score, not a probability derived from the Laplace predictive distribution and not an input to PICP or quantile-loss calculations. The formally implied Laplace probability of an upward cumulative return is

$$\Pr(R_{t,h} > 0 \mid \mathbf{X}_t) = \begin{cases} 1 - \frac{1}{2}\exp[-\hat{r}_{t,h}/b_{t,h}^{\text{fuse}}], & \hat{r}_{t,h} \geq 0, \\ \frac{1}{2}\exp[\hat{r}_{t,h}/b_{t,h}^{\text{fuse}}], & \hat{r}_{t,h} < 0. \end{cases} \tag{49}$$

For ordered quantile levels $\mathcal{Q} = \{\tau_1, \ldots, \tau_K\}$ with median index k_0, the median is fixed to $\hat{r}_{t,h}$. Quantiles above and below the median are generated through positive increments, as follows:

$$\hat{q}_{t,h}^{(\tau_k)} = \hat{q}_{t,h}^{(\tau_{k-1})} + \text{softplus}(u_{t,h}^{(k)}) s_{t,h}^{\text{fuse}}, \qquad k > k_0, \tag{50}$$

$$\hat{q}_{t,h}^{(\tau_k)} = \hat{q}_{t,h}^{(\tau_{k+1})} - \text{softplus}(u_{t,h}^{(k)}) s_{t,h}^{\text{fuse}}, \qquad k < k_0. \tag{51}$$

The pinball loss over these quantiles is denoted by $\mathcal{L}_{\text{quant}}$.

Because $r_{t,h}$ is cumulative, the implied future increment is $\hat{z}_{t,h} = \hat{r}_{t,h} - \hat{r}_{t,h-1}$ with $\hat{r}_{t,0} = 0$. Horizon coherence is encouraged through

$$\mathcal{L}_{\text{hor}} = \frac{1}{N(H-2)} \sum_{t=1}^{N} \sum_{h=2}^{H-1} \omega_{t,h}^{\text{hor}} \text{Huber}(\hat{z}_{t,h+1} - \hat{z}_{t,h}), \tag{52}$$

where $\omega_{t,h}^{\text{hor}} = \exp(-s_{t,h}^{\text{fuse}} - s_{t,h+1}^{\text{fuse}})$. A separate gate-smoothness loss discourages abrupt expert switching across adjacent horizons. To prevent early expert collapse, an annealed entropy term is used, as follows:

$$\mathcal{L}_{\text{ent}} = \frac{1}{NH} \sum_{t=1}^{N} \sum_{h=1}^{H} \sum_{e \in \mathcal{E}} \alpha_{t,h}^{(e)} \log(\alpha_{t,h}^{(e)} + \epsilon_n), \tag{53}$$

with coefficient

$$\lambda_{\text{ent}}(u) = \lambda_{\text{ent}}^0 \max\left(0, 1 - \frac{u}{U_a}\right). \tag{54}$$

The complete training objective is

$$\begin{aligned} \mathcal{L} = \mathcal{L}_{\text{pred}} &+ \lambda_{\text{nll}}\mathcal{L}_{\text{nll}} + \lambda_{\text{dir}}\mathcal{L}_{\text{dir}} + \lambda_{\text{quant}}\mathcal{L}_{\text{quant}} + \lambda_{\text{hor}}\mathcal{L}_{\text{hor}} \\ &+ \lambda_{\text{gate}}\mathcal{L}_{\text{gate}} + \lambda_{\text{ent}}(u)\mathcal{L}_{\text{ent}} + \mathcal{L}_{\text{dec}} + \mathcal{L}_{\text{stat-reg}} + \mathcal{L}_{\text{deep-reg}}. \end{aligned} \tag{55}$$

Training begins with a warm-up stage that maintains a high gate temperature and encourages broad expert participation. Joint optimization then reduces the temperature and entropy coefficient, allowing the gate to become selective after all experts have learned stable forecasts. At inference time, predicted prices are recovered as $\widehat{p}_{t+h} = p_t \exp(\widehat{r}_{t,h})$.

5. Experiments

The empirical analysis uses chronological partitions and examines point estimates for horizon dependence, component contributions, sensitivity, cross-asset consistency, and volatility-conditioned behavior. Aggregate metrics are first computed per asset and then macro-averaged. Origin-level predictions and full training records are not reported, so the results below are descriptive and do not support statistical-significance or computational-efficiency claims.

5.1. Experimental Setup

Datasets and assets. The benchmark consists of eight highly traded, large-capitalization U.S. stocks selected to cover several industries. Table 3 identifies the assets and sources. The daily open, high, low, close, adjusted-close, and volume fields cover 3 January 2005–31 December 2024. This purposive sample is not a probability sample of the MSCI World or of global equities; exchange, size, survivorship, and U.S.-market effects therefore limit external validity.

Stock splits and other vendor adjustments are handled causally at each date by $a_t = p_t^{\text{adj}} / p_t^{\text{close}}$ and $p_t^{j,\text{adj}} = a_t p_t^j$ for $j \in \{\text{open}, \text{high}, \text{low}\}$. Adjusted close supplies the closing channel. A row is removed when adjusted close is unavailable; an isolated missing non-close channel is carried forward from its latest observed value. No backward fill is used. Retained row counts and channel-specific missing percentages are not reported, which limits reproducibility, as discussed in Section 5.6.

Table 3. Stock datasets used in the experiments. The common requested daily interval is 3 January 2005–31 December 2024; the sample is restricted to eight large U.S. companies and is not globally representative.

Company	Ticker	Exchange	Broad Industry
Apple Inc.	AAPL	NASDAQ	Technology
Microsoft Corporation	MSFT	NASDAQ	Technology
Amazon.com, Inc.	AMZN	NASDAQ	Consumer services
NVIDIA Corporation	NVDA	NASDAQ	Semiconductors
JPMorgan Chase & Co.	JPM	NYSE	Financial services
Exxon Mobil Corporation	XOM	NYSE	Energy
Johnson & Johnson	JNJ	NYSE	Healthcare
Caterpillar Inc.	CAT	NYSE	Industrial manufacturing

Historical pages: https://finance.yahoo.com/quote/AAPL/history/ (accessed on 25 May 2026), https://finance.yahoo.com/quote/MSFT/history/ (accessed on 25 May 2026); https://finance.yahoo.com/quote/AMZN/history/ (accessed on 25 May 2026); https://finance.yahoo.com/quote/NVDA/history/ (accessed on 25 May 2026); https://finance.yahoo.com/quote/JPM/history/ (accessed on 25 May 2026); https://finance.yahoo.com/quote/XOM/history/ (accessed on 25 May 2026); https://finance.yahoo.com/quote/JNJ/history/ (accessed on 25 May 2026); and https://finance.yahoo.com/quote/CAT/history/ (accessed on 25 May 2026).

For each stock, $\mathbf{x}_t$ contains $D = 12$ variables. The first five are split-adjusted open, high, low, and close prices and logarithmic trading volume. The remaining variables are the one-day close-to-close log return, intraday open-to-close log return, normalized high–low range, one-day log-volume change, five-day cumulative log return, ten-day realized

volatility, and twenty-day realized volatility. The realized volatility over a window of length w is

$$v_t^{(w)} = \sqrt{\frac{1}{w}\sum_{j=0}^{w-1}(\Delta z_{t-j} - \overline{\Delta z}_t^{(w)})^2}, \tag{56}$$

where $\Delta z_t = z_t - z_{t-1}$ and $\overline{\Delta z}_t^{(w)}$ is its backward-looking mean.

Chronological partition. Figure 1 shows the single holdout design. The data through December 2017 are used for fitting, January 2018–December 2020 for validation, and January 2021–December 2024 for testing. The one contiguous test interval avoids random temporal splitting but cannot separate performance across multiple independent market eras; no rolling-origin re-estimation periods are reported.

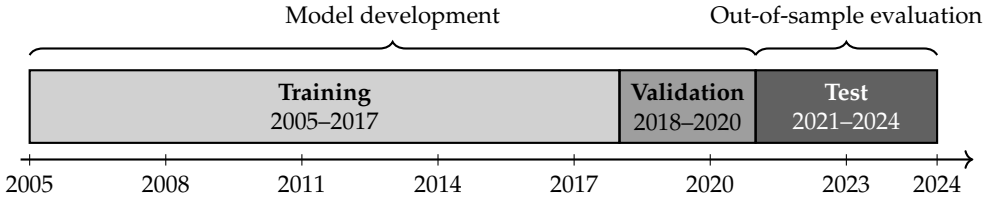


Figure 1. Chronological experimental partition. Light gray denotes training, medium gray validation/model selection, and dark gray the single held-out 2021–2024 test interval.

The chronological design prevents future behavior from influencing earlier predictions. Window normalization is performed independently for every historical input rather than using statistics calculated over the complete dataset. Technical indicators use only observations available at or before the forecasting origin. The validation and test windows are retained only when both the complete input window and complete target sequence remain within their corresponding partition.

Forecasting protocol. The historical look-back is $L = 60$ trading days and the model jointly predicts cumulative log returns for $h = 1, \ldots, 20$. Figure 2 illustrates the rolling sample. Results are reported only for the prespecified horizons $h \in \{1, 3, 5, 10, 20\}$; therefore, Figure 3 is not a complete report of all twenty trained horizons.

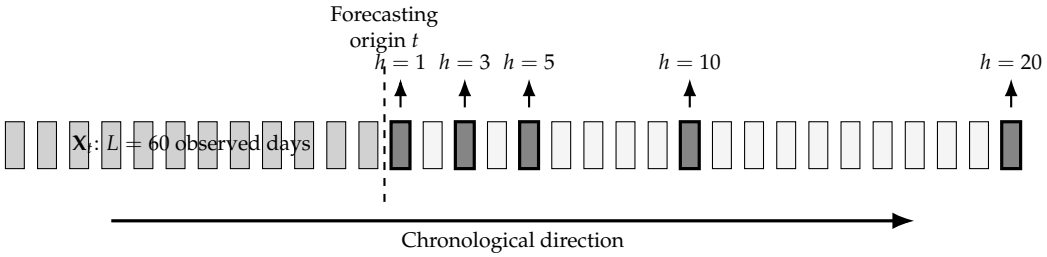


Figure 2. Rolling-window construction. Gray blocks left of the dashed line form the 60-day input; future blocks are twenty cumulative-return targets. Dark blocks identify the five reported horizons.

Adjacent samples overlap because the forecasting origin advances by one trading day. This reflects operational daily forecasting and does not create leakage because every prediction uses only the preceding historical window. Each asset is modeled independently. All methods use identical partitions, features, targets, and evaluation origins.

Comparison methods. The benchmark includes ARIMA; exponential smoothing; Prophet, an additive trend/seasonality model [43]; LSTM recurrence [7]; TCN causal convolution [8]; the original Transformer [9]; Informer sparse attention [10]; Autoformer decomposition/autocorrelation [11]; the Frequency Enhanced Decomposed Transformer (FEDformer) [12]; decomposition-linear modeling (DLinear) [23]; the patch-based time-

series Transformer (PatchTST) [13]; TimesNet periodic two-dimensional modeling [14]; the inverted Transformer (iTransformer) [15]; and TimeMixer multiscale mixing [16]. These choices cover classical conditional-mean models, recurrent and convolutional networks, attention models, and recent decomposition/multiscale architectures. Deep baselines receive the same 60×12 input and twenty return targets.

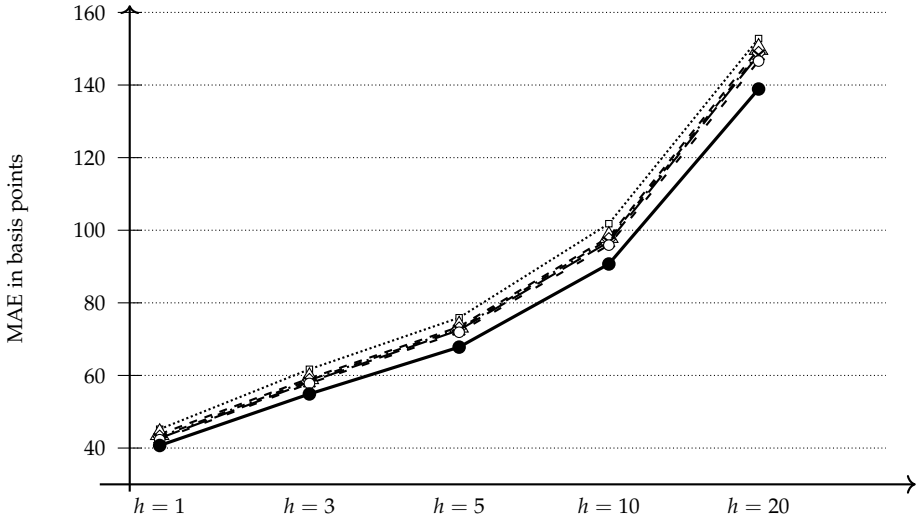


Figure 3. MAE at the five retained horizons. Densely dotted squares denote DLinear, dashed triangles PatchTST, dash–dot diamonds TimesNet, long-dashed crosses iTransformer, loosely dashed open circles TimeMixer, and solid filled circles FreqCast. Lines connect reported points and do not show confidence bands.

Hyperparameter search and feature construction can materially change machine learning comparisons, as emphasized in the AutoML literature [44]. Validation MAE selected hidden dimensions, layer counts, dropout, patch settings, and decomposition kernels, but model-specific search spaces, trial counts, initialization lists, repository commits, and selected configurations are not reported. Consequently, the baseline comparison cannot establish that every architecture is optimally or equivalently tuned. Table 4 records the reported FreqCast configuration.

Implementation. Table 4 gives the reported FreqCast settings: three spectral components, two statistical cyclic modes, three multiscale layers, and three output quantiles. AdamW uses learning-rate reduction and early stopping.

The optimizer uses gradient clipping at 1.0. The first ten epochs form the warm-up; gate temperature then decreases linearly to 0.5 by epoch 30. The loss coefficients are $\lambda_{\text{nll}} = 0.2$; $\lambda_{\text{dir}} = 0.1$; $\lambda_{\text{quant}} = 0.1$; $\lambda_{\text{hor}} = 0.05$; $\lambda_{\text{gate}} = 0.02$; $\lambda_{\text{rec}} = 0.1$; $\lambda_{\text{ovl}} = 0.02$; and $\lambda_{\text{bd}} = 0.01$. Five runs are reported for each neural setting, but exact seed values, software versions, hardware, source commits, parameter counts, and training/inference logs are not. These omissions prevent exact reproduction and empirical efficiency comparison.

Metrics. Point forecasting is evaluated on cumulative log returns using MAE and RMSE in basis points; these are not price-level errors.

$$\text{MAE}_h = \frac{10^4}{N_{\text{test}}} \sum_{t=1}^{N_{\text{test}}} |r_{t,h} - \widehat{r}_{t,h}|, \tag{57}$$

$$\text{RMSE}_h = 10^4 \sqrt{\frac{1}{N_{\text{test}}} \sum_{t=1}^{N_{\text{test}}} (r_{t,h} - \widehat{r}_{t,h})^2}. \tag{58}$$

The directional accuracy is

$$\mathrm{DA}_h = \frac{100}{N_{\text{test}}} \sum_t \mathbb{I}[\mathrm{sign}(\hat{r}_{t,h}) = \mathrm{sign}(r_{t,h})]. \tag{59}$$

Directional accuracy (DA) concerns the sign of the cumulative return. Interval diagnostics comprise mean quantile loss (QL), coverage of the central 80% interval (PICP), and its mean width (MPIW). Metrics are computed per stock before macro-averaging. PICP and MPIW alone do not establish calibration. Proper scores such as the continuous ranked probability score (CRPS) or weighted interval score (WIS), horizon-wise reliability curves, and conditional coverage tests are not reported. Table 11 provides a volatility-conditioned PICP summary.

Table 4. Reported FreqCast implementation settings. The same stated configuration is applied to all eight assets.

Component	Hyperparameter	Setting
Forecasting	Input length L	60 trading days
	Maximum horizon H	20 trading days
	Reported horizons	$\{1, 3, 5, 10, 20\}$
	Input dimension D	12 features
Decomposition	Frequency components	3
	Spectral intervals Q	12
	$c_{\min}/c_{\max}$	0.05/0.45
Statistical branch	Cyclic modes J	2
	State dimension	6
	Minimum scale $s_{\min}$	10^{-4}
Deep branch	Latent dimension D_m	64
	Multiscale layers M	3
	Kernel sizes/dilations	$\{3, 5, 7\}/\{1, 2, 4\}$
	Dropout	0.1
Output	Quantile levels	$\{0.1, 0.5, 0.9\}$
	Gate temperature	1.5 to 0.5
	Entropy annealing	30 epochs
Optimization	Optimizer	AdamW
	Learning rate/weight decay	$10^{-3}/10^{-4}$
	Batch size/maximum epochs	128/100
	Early-stopping patience	12 epochs

Likewise, valid Diebold–Mariano comparisons require paired origin-level loss differentials and dependence correction for overlapping h-step forecasts [45]; simultaneous comparisons additionally require a multiplicity adjustment. Only aggregated means are available here, so no DM statistic, block-bootstrap confidence interval, or multiplicity-adjusted p-value can be reconstructed. Tables 5 and 10 are therefore interpreted as point estimates, without statistical-superiority language.

5.2. Overall Multihorizon Forecasting Performance

Aggregate point forecasting. Table 5 reports macro-averaged return metrics for eight stocks and five retained horizons. Among the reported point estimates, TimeMixer is the strongest listed baseline, with MAE 82.9 basis points, RMSE 121.7 basis points, and DA 55.6%. The table contains neither forecast-origin confidence intervals nor significance markers because the necessary paired errors are unavailable.

Table 5. Descriptive macro-averaged cumulative-return forecasts across eight stocks and $h \in \{1, 3, 5, 10, 20\}$. Lower MAE/RMSE and higher DA are preferable; no statistical-significance inference is available.

Method	MAE ↓	RMSE ↓	DA (%) ↑
ARIMA	98.7	143.1	51.7
Exponential smoothing	96.5	140.8	51.9
Prophet	95.9	139.7	52.2
LSTM	91.3	133.2	53.1
TCN	89.4	130.1	53.8
Transformer	90.2	131.6	53.5
Informer	89.7	130.8	53.7
Autoformer	88.1	128.9	54.0
FEDformer	86.8	126.7	54.4
DLinear	87.5	127.6	54.1
PatchTST	84.9	124.0	55.0
TimesNet	84.2	123.2	55.1
iTransformer	83.7	122.5	55.4
TimeMixer	82.9	121.7	55.6
FreqCast	**78.6**	**115.3**	**57.2**

FreqCast has the lowest reported errors and highest DA in Table 5: relative to TimeMixer, its MAE is 5.2% lower, RMSE 5.3% lower, and DA 1.6 percentage points higher. These differences are descriptive associations within the chosen data, tuning record, and test interval. They motivate the component analysis below but do not by themselves confirm statistical superiority or isolate the responsible mechanism.

Performance across reported horizons. Figure 3 compares five horizon summaries for six methods. Error increases with h for every method. FreqCast's reported MAEs are 40.7, 54.9, 67.8, 90.7, and 138.9 basis points; its point-estimate gap from TimeMixer grows from 3.8% at $h = 1$ to 5.3% at $h = 20$. This pattern is consistent with, but does not prove, the intended role of structured propagation. Results for the remaining fifteen trained horizons are not reported.

Tables 7 and 8 associate short-horizon changes more strongly with removal of the intermediate/high branches and long-horizon changes with removal of the statistical branch. Because the ablations are not a factorial causal design, the pattern is interpreted as component-use evidence rather than proof that one branch generates the aggregate gap.

Prediction-interval diagnostics. Table 6 reports QL, central 80% PICP, and MPIW. FreqCast is trained with Laplace and quantile objectives, whereas baselines without a native quantile head use horizon-specific residual quantiles estimated after point training. This difference gives FreqCast a different training objective and prevents a controlled claim about intrinsic probabilistic superiority. The table is retained only as an operational comparison of the reported intervals under the stated procedures.

The reported FreqCast row has QL 6.3% below TimeMixer, PICP 80.3%, and MPIW 226.8 basis points. These three summaries do not establish calibration, especially under unequal training objectives. Equation (47) explains how expert dispersion can widen the model's scale, and Equations (50) and (51) prevent crossing; neither property guarantees empirical reliability. A controlled comparison would add identical probabilistic heads or the same conformal calibration layer to every point model and report CRPS/WIS and reliability curves.

Horizon-dependent allocation. Figure 4 shows the average reliability weights for the five retained horizons. At $h = 1$, the high-frequency expert has the largest reported weight; at $h = 20$, the statistical expert has the largest. This is a learned allocation diagnostic, not an economic attribution of returns.

Table 6. Descriptive interval diagnostics over five retained horizons. QL is the quantile loss, PICP is the central-80% coverage, and MPIW is the width in basis points. Baseline intervals are post hoc residual intervals, whereas FreqCast has native uncertainty objectives; the rows are not training-condition-matched.

Method	QL ($\times 10^{-3}$) ↓	PICP (%)	MPIW ↓
FEDformer	4.67	75.8	244.6
PatchTST	4.55	77.1	238.7
TimesNet	4.51	77.8	237.2
iTransformer	4.48	78.4	236.1
TimeMixer	4.42	79.0	233.9
FreqCast	**4.16**	**80.3**	**226.8**

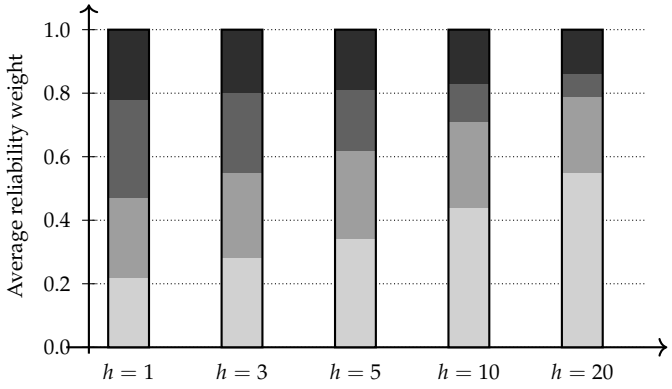


Figure 4. Average reliability weights over test windows. From bottom to top: statistical, intermediate-frequency, high-frequency, and joint-deep experts. The statistical share rises from 0.22 to 0.55 and the high-frequency share falls from 0.31 to 0.07 across the retained horizons; variability bands are unavailable.

The gate is not manually constrained to follow this pattern. Its average allocation is consistent with the architectural motivation, while nonzero weight alone does not demonstrate that an expert is independently predictive.

5.3. Ablation and Sensitivity Analysis

Table 7 reports the available ablations. It covers branch removal, adaptive versus fixed masks, fixed versus learned fusion, a joint uncertainty/disagreement removal, grouped decomposition regularization, grouped horizon-coherence losses, and probabilistic objectives. These are mostly module-level interventions and cannot identify every internal mechanism independently.

The largest reported branch-removal difference is for the statistical branch (MAE 82.5 versus 78.6 basis points). Removing the intermediate, high-frequency, or joint-deep branch gives MAE 81.2, 80.4, and 80.1, respectively. These changes show the dependence of the fitted system on each branch, but interactions among branches and re-optimization after removal preclude additive attribution.

Replacing adaptive masks with equal-width Fourier intervals gives MAE 81.7 basis points, providing the available within-architecture comparison for RQ1. Fixed fusion gives 81.0, and jointly removing scale/disagreement inputs gives 80.3. These rows do not reveal whether scale or disagreement is individually responsible and do not compare adaptive boundaries with wavelets, empirical mode decomposition (EMD), ensemble EMD (EEMD), or complete ensemble EMD with adaptive noise (CEEMDAN).

No isolated rows are reported for the residual refiner, cross-frequency attention, scale versus disagreement, horizon embedding, entropy regularization, or horizon-coherence

term separately from related horizon losses. Accordingly, Table 7 supports only the grouped interventions it reports. A complete mechanism-level factorial ablation would require new training runs and is not inferred from these numbers.

Table 7. Available module-level ablations averaged across eight stocks and five retained horizons. Several rows remove grouped mechanisms; differences are descriptive point estimates.

Variant	MAE ↓	RMSE ↓	DA (%) ↑	QL ↓
Without statistical branch	82.5	121.2	55.5	4.43
Without intermediate branch	81.2	119.1	55.9	4.34
Without high-frequency branch	80.4	118.0	56.1	4.30
Without joint deep expert	80.1	117.6	56.2	4.28
Fixed equal-width masks	81.7	119.8	55.8	4.38
Fixed fusion weights	81.0	118.7	56.0	4.33
Gate without uncertainty/disagreement	80.3	117.8	56.3	4.31
Without decomposition regularization	80.7	118.5	56.0	4.35
Without horizon-coherence losses	79.8	116.9	56.5	4.24
Without probabilistic objectives	79.4	116.2	56.6	4.62
Complete FreqCast	**78.6**	**115.3**	**57.2**	**4.16**

Table 8 gives the retained horizon-wise branch and routing ablations. Removal of the high-frequency branch has its largest listed gap at $h = 1$; removal of the statistical branch has an increasing gap through $h = 20$. Adaptive masks and learned fusion have nonzero point-estimate gaps at every reported horizon.

Table 8. Horizon-wise MAE for selected available ablations. Parentheses are descriptive increases relative to complete FreqCast; no uncertainty intervals are available.

Variant	$h = 1$	$h = 3$	$h = 5$	$h = 10$	$h = 20$
Without statistical	41.9 (+1.2)	57.0 (+2.1)	71.1 (+3.3)	96.5 (+5.8)	147.6 (+8.7)
Without intermediate	42.2 (+1.5)	57.2 (+2.3)	70.5 (+2.7)	94.8 (+4.1)	141.3 (+2.4)
Without high frequency	42.6 (+1.9)	56.7 (+1.8)	69.7 (+1.9)	93.1 (+2.4)	140.1 (+1.2)
Fixed masks	42.4 (+1.7)	57.0 (+2.1)	70.6 (+2.8)	94.9 (+4.2)	143.6 (+4.7)
Fixed fusion	42.1 (+1.4)	56.5 (+1.6)	69.5 (+1.7)	94.2 (+3.5)	142.6 (+3.7)
Complete FreqCast	**40.7**	**54.9**	**67.8**	**90.7**	**138.9**

The grouped "without decomposition regularization" row has a higher MAE and QL, but it does not separately identify reconstruction, overlap, and boundary-stability penalties. Likewise, the grouped horizon and probabilistic rows show changes in the reported metrics without establishing empirical calibration or isolating individual loss terms.

Learned-boundary diagnostics. The test windows are divided into within-benchmark volatility terciles using 20-day realized volatility. Figure 5 reports mean normalized cutoffs: $c_{t,1}$ decreases from 0.24 to 0.16 and $c_{t,2}$ from 0.61 to 0.48 between the low- and high-volatility groups. Mechanically, lower cutoffs reduce the frequencies assigned to the low and intermediate masks and expand the relative spectral range routed upward. This figure does not show representative component time series, boundary dispersion, or asset-specific distributions.

The state features in Equation (14) permit this movement. Relative to fixed equal-width Fourier masks, the reported MAE differences are 2.1%, 3.7%, and 5.6% in the low-, medium-, and high-volatility windows. Together with Tables 7 and 8, this is quantitative evidence that the learned routing changes with volatility and that removing bands changes forecasts. It

is not evidence that the bands universally correspond to economic trends, cycles, or shocks. Representative decompositions across stocks and direct wavelet/EMD/EEMD/CEEMDAN comparisons are not reported, so the statistical meaning and comparative advantage of the boundaries remain only partially assessed.

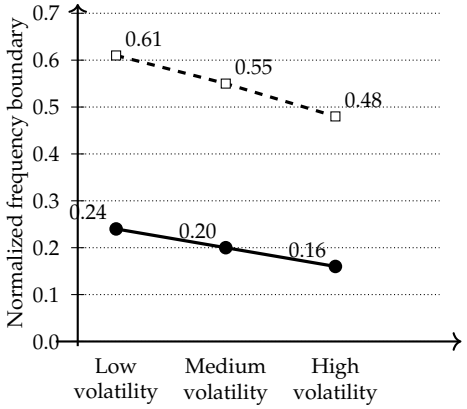


Figure 5. Mean normalized frequency boundaries by volatility tercile. Solid filled circles denote $c_{t,1}$ and dashed open squares $c_{t,2}$. Lower cutoffs expand the frequencies routed to higher-band pathways. Only means are available; no cross-window or cross-asset uncertainty bands are shown.

Sensitivity. Figure 6 reports retained point estimates for look-back L and cyclic modes J. MAE decreases through $L = 60$ and then rises; $J = 2$ has the lowest listed mean. Without variability estimates, explanations based on regime mixing or redundant modes remain interpretations rather than demonstrated causes.

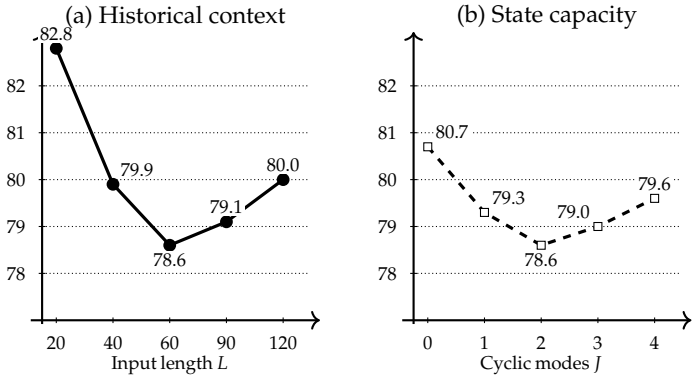


Figure 6. Sensitivity point estimates for historical context and statistical-state capacity. Panel (**a**) uses filled circles for input length; panel (**b**) uses open squares for cyclic modes. Both use the same MAE scale; uncertainty bands are unavailable.

Relative to the listed $L = 60$ result, $L = 40$ and $L = 90$ differ by 1.3 and 0.5 basis points; $J = 1$ and $J = 3$ are within one basis point of $J = 2$. These point estimates suggest local smoothness, but they do not establish insensitivity to the look-back window. Latent-dimension and layer-count series are not reported.

5.4. Computational Complexity and Reporting Scope

Table 9 gives an architecture-level per-window complexity account. Let D_m be the deep width, D_a the attention width, d_a the statistical-state dimension, S the number of convolution scales, M the deep layers, $k_{\max}$ the largest kernel, and $E = 4$ the experts. Constants and small output multilayers are suppressed.

Table 9. Symbolic per-window inference complexity of FreqCast modules. This is not a measured runtime or memory benchmark.

Module	Time Complexity	Principal Activation Memory
DFT, masks, inverse DFTs	$O(DL \log L)$	$O(DL)$ complex coefficients/components
Residual and multiscale encoders	$O(MSLD_m k_{\max} + MLD_m^2)$	$O(MLD_m)$
Cross-frequency attention	$O(L^2 D_a)$	$O(L^2 + LD_a)$
Structured statistical branch	$O((L+H)d_a^2)$	$O(d_a + Hd_a)$
Expert heads and reliability gate	$O(HED_m^2)$	$O(HED_m)$
Dominant combined order	$O(DL \log L + MLD_m^2 + L^2 D_a + HED_m^2)$	$O(MLD_m + L^2 + HED_m)$

The cross-frequency attention term retains quadratic dependence on L, as in full-attention sequence models, while PatchTST reduces the attended length through patching and TimeMixer avoids this exact attention matrix through multiscale mixing. FreqCast additionally pays for Fourier transforms, three refiners, multiple convolutional paths, four expert heads, and auxiliary objectives. Table 9 therefore cannot show that its point-estimate accuracy differences justify its added cost. Trainable-parameter counts, wall-clock training/inference time, peak device memory, hardware, batch-level profiling, and matched measurements for principal baselines are not reported.

5.5. Cross-Asset and Volatility-Conditioned Results

Aggregate results can conceal concentration in a few series. Table 10 therefore lists the retained FreqCast and TimeMixer point estimates by asset. FreqCast has a lower reported MAE and higher DA for all eight sample stocks, with relative MAE differences of 4.2–6.4%. This is within-sample-universe consistency, not evidence for other exchanges, capitalization groups, indices, or periods.

The largest listed relative difference is for NVIDIA, followed by Amazon; the smallest is for Johnson & Johnson. Explanations based on transience or persistence are hypotheses consistent with the architecture and cannot be established from Table 10 alone. The positive DA differences indicate that the lower reported magnitude error is accompanied by sign differences, but do not establish economic profitability.

Figure 7 visualizes these eight relative MAE differences. Every displayed bar is positive, so the reported macro-average is not produced by one asset; the figure provides no confidence intervals and cannot support population-level generalization.

Table 10. Descriptive asset-level comparison averaged over $h \in \{1, 3, 5, 10, 20\}$. The eight large U.S. stocks do not constitute a global equity sample.

Company	Ticker	MAE		Reduction	DA (%)	
		TimeMixer	FreqCast	(%)	TimeMixer	FreqCast
Apple	AAPL	79.1	**75.0**	5.2	56.1	**57.8**
Microsoft	MSFT	76.8	**72.9**	5.1	56.0	**57.4**
Amazon	AMZN	90.2	**85.2**	5.5	55.2	**57.0**
NVIDIA	NVDA	98.4	**92.1**	6.4	54.8	**56.9**
JPMorgan Chase	JPM	78.5	**74.8**	4.7	55.7	**57.2**
Exxon Mobil	XOM	84.6	**80.2**	5.2	55.4	**56.8**
Johnson & Johnson	JNJ	74.3	**71.2**	4.2	56.0	**57.3**
Caterpillar	CAT	81.3	**77.4**	4.8	55.6	**57.2**
Macro average	–	82.9	**78.6**	5.2	55.6	**57.2**

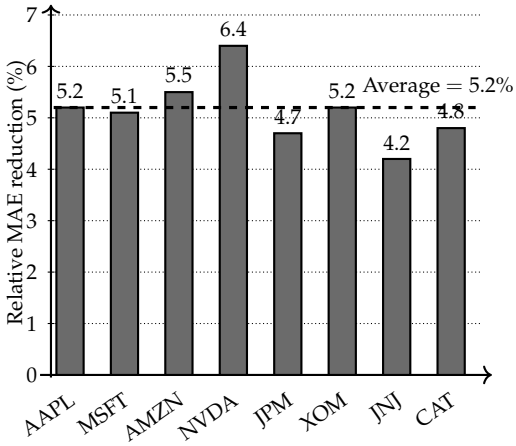


Figure 7. Reported relative MAE difference between FreqCast and TimeMixer for each sampled stock. Dark-gray bars are assets, and the dashed line the eight-asset macro-average; no uncertainty intervals are available.

The reported average across-seed MAE standard deviation is 0.74 basis points for FreqCast and 0.81 for TimeMixer over five runs, with lower FreqCast variability on six assets. Exact seeds and run-level values are not reported, so confidence intervals and paired seed tests cannot be reconstructed; the summary does not prove optimization stability.

Volatility-conditioned summaries. Test windows are grouped by cross-asset terciles of 20-day realized volatility. Table 11 shows higher reported errors for both methods at higher volatility. The relative MAE differences between FreqCast and TimeMixer are 3.8%, 5.5%, and 5.6% in the three groups.

Table 11. Descriptive performance by volatility tercile. PICP is empirical coverage of the nominal central 80% interval and is not a conditional-coverage test.

Regime	Method	MAE ↓	RMSE ↓	DA (%) ↑	PICP (%)
Low	TimeMixer	54.8	78.2	58.1	81.2
	FreqCast	**52.7**	**75.3**	**59.4**	80.5
Medium	TimeMixer	78.6	114.9	55.8	79.4
	FreqCast	**74.3**	**108.9**	**57.5**	80.2
High	TimeMixer	115.3	172.0	52.9	76.4
	FreqCast	**108.8**	**161.7**	**54.7**	80.2

Figure 5 shows lower mean cutoffs in higher-volatility groups, while Table 11 reports FreqCast PICP between 80.2% and 80.5%. TimeMixer's post hoc residual intervals have 76.4% coverage in the high-volatility group. These grouped PICP values describe marginal behavior within terciles; they are not formal conditional-coverage tests and do not replace horizon-wise reliability analysis.

Figure 8 displays the regime-wise point estimates. The absolute MAE difference grows from 2.1 basis points in the low group to 6.5 in the high group. The association is compatible with market-conditioned decomposition but does not isolate it from the other FreqCast modules.

A transition window is defined descriptively when its volatility category differs from its category ten trading days earlier. The reported FreqCast MAEs are 76.2 and 91.7 basis points in stable and transition windows, versus 80.0 and 98.0 for TimeMixer. Equation (26) permits faster boundary movement when adjacent state vectors differ, which is one possible explanation; no module-specific transition experiment identifies causality.

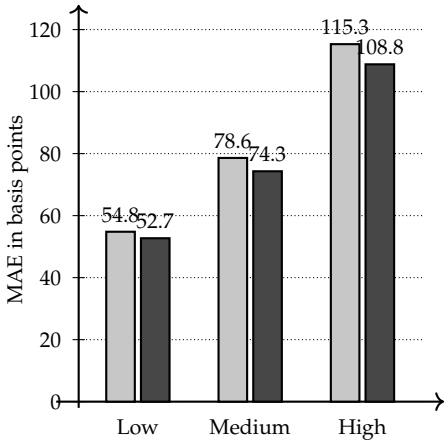


Figure 8. Reported MAE by volatility tercile. In each pair, the left light-gray bar is TimeMixer and the right dark-gray bar FreqCast. The displayed difference is larger in the high-volatility group; no uncertainty intervals are available.

Average cross-expert standard deviation is reported as 36.5 basis points in stable windows and 51.8 in transition windows. Equations (45) and (47) show how dispersion affects scale and weights. A disagreement-only transition ablation is not reported, so its isolated effect is unknown.

The qualitative interpretation associates larger short-horizon high/joint-deep weights with NVIDIA and Amazon, larger statistical weights with Johnson & Johnson and JPMorgan Chase, and larger intermediate weights with Exxon Mobil and Caterpillar. Asset-level weight distributions are not reported, so these observations are not treated as a separate quantitative result.

5.6. Evidential Scope and Limitations

The evidence has seven principal limits. First, the eight surviving large U.S. companies studied do not represent international exchanges, indices, small- and mid-capitalization equities, or other asset classes. Second, the single 2021–2024 holdout is a past contiguous period; multiple rolling origins and later regimes are needed to assess temporal generalization. Third, FreqCast has greater architectural complexity than several baselines, while parameter counts, wall-clock times, peak memory, and matched hardware profiles are unavailable. Fourth, the selected 60-day look-back can mix regimes, and the sensitivity evidence contains only point estimates. Fifth, inputs contain only historical price/volume transformations; news, fundamentals, order flow, market-wide factors, and macroeconomic variables are absent. Sixth, learned frequency boundaries may not transfer across assets, sampling frequencies, or market structures, and no wavelet/EMD-family decomposition comparison or representative component panel is available. Seventh, exact baseline searches, implementation commits, software/hardware configuration, seed identities, raw missingness counts, origin-level predictions, and complete probabilistic outputs are not preserved.

These limits constrain interpretation. The reported results describe this benchmark and do not establish broad financial applicability, distribution-shift robustness, full probabilistic calibration, economic value, or statistical superiority. Interpretability refers to inspectable bands, states, expert forecasts, and gate weights; causal economic interpretation would require independent validation. Reproducible extension requires a versioned data manifest, code and configurations, complete per-origin forecasts, exact seeds, matched probabilistic baseline heads or calibration, DM/block-bootstrap inference with multiplicity control, CRPS/WIS and reliability plots, all twenty horizons, targeted ablations, and matched resource profiling.

6. Conclusions

FreqCast forecasts cumulative log returns and reconstructs prices afterward; it is not trained on price-level error. Its primary contribution is the coordinated use of sample-conditioned soft spectral boundaries, a structured state-space expert, component-specific causal encoders, and a horizon-conditioned gate using predictive scale and disagreement.

For RQ1, fixed-mask and volatility-tercile summaries indicate that learned routing changes with market state and is associated with lower reported error than equal-width Fourier masks, while the absence of representative decompositions and wavelet/EMD comparisons limits interpretation. For RQ2, branch-removal summaries associate the high-frequency path more strongly with the shortest retained horizon and the statistical path with longer horizons, but several internal mechanisms remain unisolated. For RQ3, mean gate weights shift from the high-frequency expert toward the statistical expert as the horizon grows, and boundary/interval summaries vary with volatility; these are descriptive diagnostics rather than causal or statistically tested findings. Across eight large U.S. stocks in the single 2021–2024 holdout, FreqCast has the strongest reported point estimates among the listed methods. Unequal probabilistic training, incomplete baseline-tuning records, missing origin-level errors, and absent computational logs preclude claims of calibrated superiority, statistical significance, or favorable cost–benefit. Broader assets, rolling periods, exogenous information, matched probabilistic baselines, complete horizon reporting, mechanism-level ablations, and reproducible resource measurements are necessary before generalizing beyond this benchmark.

Author Contributions: Conceptualization, Y.L. and H.Z.; Methodology, Y.L. and H.Z.; Software, Y.L. and H.Z.; Validation, Y.L. and H.Z.; Formal analysis, Y.L. and H.Z.; Investigation, Y.L. and H.Z.; Resources, Y.L. and H.Z.; Data curation, Y.L. and H.Z.; Writing—original draft, Y.L. and H.Z.; Writing—review and editing, Y.L. and H.Z. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: The source market pages are listed in Table 3. This manuscript version does not distribute the downloaded raw files, derived dataset, origin-level forecasts, training logs, source code, or configuration files; exact numerical reproduction is therefore not possible from the article files alone.

Conflicts of Interest: The author declares no conflicts of interest.

References

1. Box, G.E.P.; Jenkins, G.M.; Reinsel, G.C.; Ljung, G.M. *Time Series Analysis: Forecasting and Control*, 5th ed.; John Wiley & Sons: Hoboken, NJ, USA, 2015.
2. Engle, R.F. Autoregressive Conditional Heteroscedasticity with Estimates of the Variance of United Kingdom Inflation. *Econometrica* **1982**, *50*, 987–1007. [CrossRef]
3. Bollerslev, T. Generalized Autoregressive Conditional Heteroskedasticity. *J. Econom.* **1986**, *31*, 307–327. [CrossRef]
4. Kim, H.Y.; Won, C.H. Forecasting the Volatility of Stock Price Index: A Hybrid Model Integrating LSTM with Multiple GARCH-Type Models. *Expert Syst. Appl.* **2018**, *103*, 25–37. [CrossRef]
5. Iftikhar, H.; Khan, F.; Torres Armas, E.A.; Rodrigues, P.C.; López-Gonzales, J.L. A Novel Hybrid Framework for Forecasting Stock Indices Based on the Nonlinear Time Series Models. *Comput. Stat.* **2025**, *40*, 4163–4186. [CrossRef]
6. Nasir, J.; Iftikhar, H.; Aamir, M.; Iftikhar, H.; Rodrigues, P.C.; Rehman, M.Z. A Hybrid LMD–ARIMA–Machine Learning Framework for Enhanced Forecasting of Financial Time Series: Evidence from the NASDAQ Composite Index. *Mathematics* **2025**, *13*, 2389. [CrossRef]
7. Hochreiter, S.; Schmidhuber, J. Long Short-Term Memory. *Neural Comput.* **1997**, *9*, 1735–1780. [CrossRef]
8. Bai, S.; Kolter, J.Z.; Koltun, V. An Empirical Evaluation of Generic Convolutional and Recurrent Networks for Sequence Modeling. *arXiv* **2018**, arXiv:1803.01271.

9. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, Ł.; Polosukhin, I. Attention Is All You Need. In Proceedings of the Advances in Neural Information Processing Systems, Long Beach, CA, USA, 4–9 December 2017; Volume 30, pp. 5998–6008.
10. Zhou, H.; Zhang, S.; Peng, J.; Zhang, S.; Li, J.; Xiong, H.; Zhang, W. Informer: Beyond Efficient Transformer for Long Sequence Time-Series Forecasting. In Proceedings of the AAAI Conference on Artificial Intelligence, Online, 2–9 February 2021; Volume 35, pp. 11106–11115.
11. Wu, H.; Xu, J.; Wang, J.; Long, M. Autoformer: Decomposition Transformers with Auto-Correlation for Long-Term Series Forecasting. In Proceedings of the Advances in Neural Information Processing Systems, Virtually, 6–14 December 2021; Volume 34, pp. 22419–22430.
12. Zhou, T.; Ma, Z.; Wen, Q.; Wang, X.; Sun, L.; Jin, R. FEDformer: Frequency Enhanced Decomposed Transformer for Long-Term Series Forecasting. In Proceedings of the 39th International Conference on Machine Learning, PMLR, Proceedings of Machine Learning Research, Baltimore, MD, USA, 17–23 July 2022; Volume 162, pp. 27268–27286.
13. Nie, Y.; Nguyen, N.H.; Sinthong, P.; Kalagnanam, J. A Time Series Is Worth 64 Words: Long-Term Forecasting with Transformers. In Proceedings of the International Conference on Learning Representations, Kigali, Rwanda, 1–5 May 2023.
14. Wu, H.; Hu, T.; Liu, Y.; Zhou, H.; Wang, J.; Long, M. TimesNet: Temporal 2D-Variation Modeling for General Time Series Analysis. In Proceedings of the International Conference on Learning Representations, Kigali, Rwanda, 1–5 May 2023.
15. Liu, Y.; Hu, T.; Zhang, H.; Wu, H.; Wang, S.; Ma, L.; Long, M. iTransformer: Inverted Transformers Are Effective for Time Series Forecasting. In Proceedings of the International Conference on Learning Representations, Vienna, Austria, 7–11 May 2024.
16. Wang, S.; Wu, H.; Shi, X.; Hu, T.; Luo, H.; Ma, L.; Zhang, J.Y.; Zhou, J. TimeMixer: Decomposable Multiscale Mixing for Time Series Forecasting. In Proceedings of the International Conference on Learning Representations, Vienna, Austria, 7–11 May 2024.
17. Bao, W.; Yue, J.; Rao, Y. A Deep Learning Framework for Financial Time Series Using Stacked Autoencoders and Long Short-Term Memory. *PLoS ONE* **2017**, *12*, e0180944. [CrossRef]
18. Fischer, T.; Krauss, C. Deep Learning with Long Short-Term Memory Networks for Financial Market Predictions. *Eur. J. Oper. Res.* **2018**, *270*, 654–669. [CrossRef]
19. Zhang, L.; Aggarwal, C.C.; Qi, G.J. Stock Price Prediction via Discovering Multi-Frequency Trading Patterns. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Halifax, NS, Canada, 13–17 August 2017*; Association for Computing Machinery: New York, NY, USA; pp. 2141–2149.
20. Salinas, D.; Flunkert, V.; Gasthaus, J.; Januschowski, T. DeepAR: Probabilistic Forecasting with Autoregressive Recurrent Networks. *Int. J. Forecast.* **2020**, *36*, 1181–1191. [CrossRef]
21. Oreshkin, B.N.; Carpov, D.; Chapados, N.; Bengio, Y. N-BEATS: Neural Basis Expansion Analysis for Interpretable Time Series Forecasting. In Proceedings of the International Conference on Learning Representations, Addis Ababa, Ethiopia, 26 April–1 May 2020.
22. Lim, B.; Arık, S.Ö.; Loeff, N.; Pfister, T. Temporal Fusion Transformers for Interpretable Multi-Horizon Time Series Forecasting. *Int. J. Forecast.* **2021**, *37*, 1748–1764. [CrossRef]
23. Zeng, A.; Chen, M.; Zhang, L.; Xu, Q. Are Transformers Effective for Time Series Forecasting? In Proceedings of the AAAI Conference on Artificial Intelligence, Washington, DC, USA, 7–14 February 2023; Volume 37, pp. 11121–11128.
24. Chen, P.; Zhang, Y.; Cheng, Y.; Shu, Y.; Wang, Y.; Wen, Q.; Yang, B.; Guo, C. Pathformer: Multi-scale Transformers with Adaptive Pathways for Time Series Forecasting. In Proceedings of the International Conference on Learning Representations, Vienna, Austria, 7–11 May 2024.
25. Lu, J.; Han, X.; Sun, Y.; Yang, S. CATS: Enhancing Multivariate Time Series Forecasting by Constructing Auxiliary Time Series as Exogenous Variables. In Proceedings of the 41st International Conference on Machine Learning, PMLR, Proceedings of Machine Learning Research, Vienna, Austria, 21–27 July 2024; Volume 235, pp. 32990–33006.
26. Wang, Y.; Wu, H.; Dong, J.; Qin, G.; Zhang, H.; Liu, Y.; Qiu, Y.; Wang, J.; Long, M. TimeXer: Empowering Transformers for Time Series Forecasting with Exogenous Variables. In Proceedings of the Advances in Neural Information Processing Systems, Vancouver, BC, Canada, 9–15 December 2024; Volume 37.
27. Fei, J.; Yi, K.; Fan, W.; Zhang, Q.; Niu, Z. Amplifier: Bringing Attention to Neglected Low-Energy Components in Time Series Forecasting. In Proceedings of the AAAI Conference on Artificial Intelligence, Philadelphia, PA, USA, 25 February–4 March 2025; Volume 39, pp. 11645–11653.
28. Zhang, Z.; Pham, T.D.; An, Y.; Doan, N.P.; Alsharari, M.; Tran, V.H.; Hoang, A.T.; Vandierendonck, H.; Mai, S.T. WaveletMixer: A Multi-Resolution Wavelets Based MLP-Mixer for Multivariate Long-Term Time Series Forecasting. In Proceedings of the AAAI Conference on Artificial Intelligence, Philadelphia, PA, USA, 25 February–4 March 2025; Volume 39, pp. 22741–22749.
29. Wang, S.; Li, J.; Shi, X.; Ye, Z.; Mo, B.; Lin, W.; Ju, S.; Chu, Z.; Jin, M. TimeMixer++: A General Time Series Pattern Machine for Universal Predictive Analysis. In Proceedings of the International Conference on Learning Representations, Singapore, 24–28 April 2025.

30. Jin, M.; Wang, S.; Ma, L.; Chu, Z.; Zhang, J.Y.; Shi, X.; Chen, P.Y.; Liang, Y.; Li, Y.F.; Pan, S.; et al. Time-LLM: Time Series Forecasting by Reprogramming Large Language Models. In Proceedings of the International Conference on Learning Representations, Vienna, Austria, 7–11 May 2024.
31. Goswami, M.; Szafer, K.; Choudhry, A.; Cai, Y.; Li, S.; Dubrawski, A. MOMENT: A Family of Open Time-series Foundation Models. In Proceedings of the 41st International Conference on Machine Learning. PMLR, Proceedings of Machine Learning Research, Vienna, Austria, 21–27 July 2024; Volume 235, pp. 16115–16152.
32. Das, A.; Kong, W.; Sen, R.; Zhou, Y. A decoder-only foundation model for time-series forecasting. In Proceedings of the 41st International Conference on Machine Learning, PMLR, Proceedings of Machine Learning Research, Vienna, Austria, 21–27 July 2024; Volume 235, pp. 10148–10167.
33. Woo, G.; Liu, C.; Kumar, A.; Xiong, C.; Savarese, S.; Sahoo, D. Unified Training of Universal Time Series Forecasting Transformers. In Proceedings of the 41st International Conference on Machine Learning, PMLR, Proceedings of Machine Learning Research, Vienna, Austria, 21–27 July 2024; Volume 235,pp. 53140–53164.
34. Liu, Y.; Zhang, H.; Li, C.; Huang, X.; Wang, J.; Long, M. Timer: Generative Pre-trained Transformers Are Large Time Series Models. In Proceedings of the 41st International Conference on Machine Learning, PMLR, Proceedings of Machine Learning Research, Vienna, Austria, 21–27 July 2024; Volume 235, pp. 32369–32399.
35. Dong, J.; Wu, H.; Wang, Y.; Qiu, Y.Z.; Zhang, L.; Wang, J.; Long, M. TimeSiam: A Pre-Training Framework for Siamese Time-Series Modeling. In Proceedings of the 41st International Conference on Machine Learning, PMLR, Proceedings of Machine Learning Research, Vienna, Austria, 21–27 July 2024; Volume 235, pp. 11412–11436.
36. Ansari, A.F.; Stella, L.; Turkmen, C.; Zhang, X.; Mercado, P.; Shen, H.; Shchur, O.; Rangapuram, S.S.; Arango, S.P.; Kapoor, S.; et al. Chronos: Learning the Language of Time Series. *arXiv* **2024**, arXiv:2403.07815.
37. Liu, Y.; Qin, G.; Huang, X.; Wang, J.; Long, M. Timer-XL: Long-Context Transformers for Unified Time Series Forecasting. In Proceedings of the International Conference on Learning Representations, Singapore, 24–28 April 2025.
38. Shi, X.; Wang, S.; Nie, Y.; Li, D.; Ye, Z.; Wen, Q.; Jin, M. Time-MoE: Billion-Scale Time Series Foundation Models with Mixture of Experts. In Proceedings of the International Conference on Learning Representations, Singapore, 24–28 April 2025.
39. Liu, Y.; Qin, G.; Shi, Z.; Chen, Z.; Yang, C.; Huang, X.; Wang, J.; Long, M. Sundial: A Family of Highly Capable Time Series Foundation Models. In Proceedings of the 42nd International Conference on Machine Learning, PMLR, Proceedings of Machine Learning Research, Vancouver, BC, Canada, 13–19 July 2025; Volume 267, pp. 39295–39317.
40. Aksu, I.T.; Woo, G.; Liu, J.; Liu, X.; Liu, C.; Savarese, S.; Xiong, C.; Sahoo, D. GIFT-Eval: A Benchmark for General Time Series Forecasting Model Evaluation. In Proceedings of the NeurIPS 2024 Workshop on Time Series in the Age of Large Models, Vancouver, BC, Canada, 15 December 2024.
41. Liu, P.; Guo, H.; Dai, T.; Li, N.; Bao, J.; Ren, X.; Jiang, Y.; Xia, S.T. CALF: Aligning LLMs for Time Series Forecasting via Cross-modal Fine-Tuning. In Proceedings of the AAAI Conference on Artificial Intelligence, Philadelphia, PA, USA, 25 February–4 March 2025; Volume 39, pp. 18915–18923.
42. Liu, C.; Xu, Q.; Miao, H.; Yang, S.; Zhang, L.; Long, C.; Li, Z.; Zhao, R. TimeCMA: Towards LLM-Empowered Multivariate Time Series Forecasting via Cross-Modality Alignment. In Proceedings of the AAAI Conference on Artificial Intelligence, Philadelphia, PA, USA, 25 February–4 March 2025; Volume 39, pp. 18780–18788.
43. Taylor, S.J.; Letham, B. Forecasting at Scale. *Am. Stat.* **2018**, *72*, 37–45. [CrossRef]
44. Ravishankar, S.; Battineni, G. A Survey on Recent Advancements in Auto-Machine Learning with a Focus on Feature Engineering. *J. Comput. Cogn. Eng.* **2025**, *4*, 56–63. [CrossRef]
45. Diebold, F.X.; Mariano, R.S. Comparing Predictive Accuracy. *J. Bus. Econ. Stat.* **1995**, *13*, 253–263. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.