EXAONE Finance 1.0:
An Attention-free Time Series Foundation Model for Financial Time Series

LG AI Research*
Seunghan Lee, Jaehoon Lee, Jun Seo, Tae Yoon Lim, Dongwan Kang, Hwanil Choi, Minjae Kim, Sungdong Yoo, Junhyeok Kang, Sangjun Han, Soonyoung Lee, Wonbin Ahn

This technical report presents **EXAONE Forecast for Finance** (EXAONE Finance), a financial time series foundation model (TSFM) tailored to financial forecasting. While recent TSFMs achieve strong zero-shot performance through large-scale pretraining, they are primarily developed for *general-domain* time series and largely rely on self-attention backbones whose computational cost grows quadratically with sequence length and variate count. Moreover, they assume fully observed inputs and are pretrained on corpora that fail to adequately capture the unique dynamics of financial markets. These limitations hinder their applicability to finance, where long, many-channel, intermittently observed panels are common. To address these challenges, EXAONE Finance adopts an *attention-free* architecture, replacing self-attention with two simple yet effective linear-time operators: (1) a *causal 1D convolution* for temporal mixing and (2) a *group-aware pooling multi-layer perceptron (MLP)* for variate mixing. Furthermore, a *masked-context augmentation* exposes the model to contiguous missing spans during training, improving robustness to the missingness pervasive in financial markets. EXAONE Finance is pretrained on a *synthetic financial* corpus whose generative process is designed to reproduce the properties of financial series such as heavy tails, volatility clustering, jumps, regime shifts, and cross-asset dependence, combined with a domain-agnostic synthetic source. On FinVerse, a financial forecasting benchmark covering diverse asset classes, EXAONE Finance attains state-of-the-art performance, ranking first across all three evaluation tiers: point-forecast accuracy, cross-sectional asset ranking, and portfolio profitability.

Model `https://huggingface.co/LG-AI-Research/EXAONE-Finance-1.0`

Code `https://github.com/LGAI-Research/EXAONE-Forecast`

1 Introduction

Time series (TS) forecasting underpins decision-making across domains such as finance [28, 32, 43] and energy [23]. To address these forecasting needs, deep learning models, including recurrent [42], convolutional [5, 38], and transformer-based [40, 52] architectures, capture temporal structure effectively, yet are typically trained for individual datasets and transfer poorly across domains. To address this limitation, time series foundation models (TSFMs) pursue generalizable forecasting through large-scale pretraining [3, 11, 18, 20, 29, 31, 48], achieving strong zero-shot performance across diverse TS benchmarks [1, 19].

Despite their success on general-domain benchmarks, *applying TSFMs to financial forecasting remains challenging* due to three gaps. **(i) Cost:** the self-attention backbones used by most TSFMs incur a cost that grows quadratically with the sequence length and the number of variates, which is expensive for the long, many-channel panels in finance. **(ii) Missingness:** financial series are interrupted by market closures and trading halts, yet most TSFMs assume contiguous inputs. **(iii) Domain mismatch:** pretraining corpora are dominated by non-financial domains, under-representing financial dynamics such as heavy tails and low signal-to-noise.

Broad view of financial assets. Crucially, *financial forecasting* is far broader than stock-price prediction. Financial decision-making spans a heterogeneous universe of assets with distinct sampling frequencies, scales, and dynamics: equities, exchange-traded funds (ETFs), foreign exchange (FX), commodities, crypto-assets, fixed-income instruments, and macroeconomic indicators. These asset classes exhibit diverse characteristics, ranging from high-frequency, heavy-tailed crypto series to low-frequency, smoothly varying macroeconomic indicators, and are often

Table 1: **Comparison of TSFMs.** EXAONE Finance is the only *attention-free* convolutional TSFM and the only model in this comparison pretrained on *financial-domain data*, which we generate synthetically. (CNN: convolutional neural network, RNN: recurrent neural network, SSM: state-space model, MLP: multi-layer perceptron)

Models	**Architecture**		**Dataset**
	Temporal mixing	**Variate mixing**	**Financial**
Chronos, Chronos-Bolt, TimesFM, Lag-Llama, TimeGPT, Timer, MOMENT, Sundial, TEMPO, ROSE, Time-MoE, PatchTST-FM, Kairos, YingLong, CleanTS, TabPFN-TS, VisionTS	Attention	–	×
Chronos-2, Moirai, Moirai-MoE, Toto, UniTS	Attention	Attention	×
TiRex (xLSTM), TempoPFN, FlowState, Reverso	RNN / SSM	–	×
TTM (TinyTimeMixer)	MLP	MLP	×
EXAONE Finance (Ours)	**CNN**	**MLP (Group pooling)**	✓

observed jointly as multi-channel panels (e.g., the open/high/low/close/volume fields of a single security). A *financial TSFM* should therefore support this spectrum of asset classes, enabling a single model to address the diverse forecasting needs encountered across financial markets.

To this end, we propose **EXAONE Finance**, a TSFM tailored to financial forecasting. Unlike general-purpose TSFMs, EXAONE Finance is designed to address the unique characteristics of financial TS. It adopts an 1) *attention-free* architecture by replacing self-attention with two linear-time operators, namely a causal 1D convolution for temporal mixing and a group-aware pooling MLP for variate mixing, and is trained with a 2) *masked-context augmentation* that improves robustness to missing observations. Furthermore, our method is pretrained on a 3) *synthetic financial corpus*, generated so that the sampled series carry the properties of financial markets such as heavy tails, volatility clustering, jumps, regime shifts, and cross-asset dependence. We evaluate it on FinVerse [26], a standardized financial benchmark that assesses models across three tiers (point accuracy, cross-sectional skill, and portfolio backtesting) over a broad range of financial assets. Table 1 contrasts EXAONE Finance with representative TSFMs in terms of their mixing architecture and pretraining domain. The main contributions are as follows:

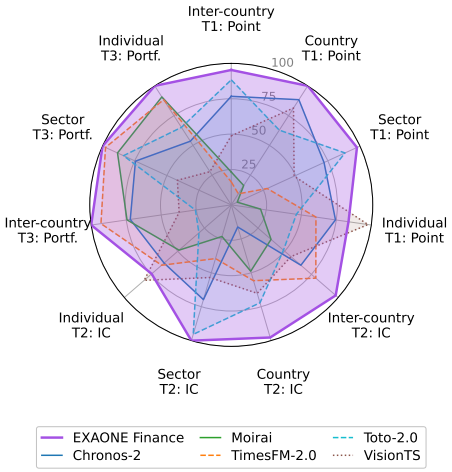

Figure 1: **FinVerse benchmark.** EXAONE Finance outperforms five representative TSFMs.

- We propose **EXAONE Finance**, an attention-free TSFM that mixes time with *causal convolutions* and variates with a *group-aware pooling MLP* for linear-time cost (with a patch-based forecasting head), made robust to missing data by a masked-context augmentation that masks contiguous spans of the input.
- We pretrain EXAONE Finance on a *synthetic financial corpus*, using a generator whose sampling process is designed to reproduce the properties of financial series such as heavy tails, volatility clustering, jumps, regime shifts, and cross-asset dependence. The model therefore sees no real market data during pretraining.
- Through evaluation on **FinVerse**, a financial benchmark that spans a broad range of asset classes and scores models on three tiers (point accuracy, cross-sectional skill, and portfolio backtesting), EXAONE Finance attains state-of-the-art (SoTA) performance across financial forecasting tasks. As shown in Figure 1, it traces a near-maximal envelope across the scope × tier grid, ranking in the top percentiles for nearly every cell.

2 Related Work

2.1 Time Series Foundation Models

Inspired by the success of large language models, recent work pretrains a single model on large, heterogeneous TS corpora for zero-shot forecasting. Chronos [3] tokenizes scalar values and reuses a T5 encoder–decoder language-model backbone, later distilled into the faster Chronos-Bolt and extended to multivariate inputs by Chronos-2 [2]. TimesFM [11] adopts a decoder-only, patch-based transformer, while Moirai [48] uses a masked encoder with any-

variate attention to support arbitrary numbers of covariates, and Moirai-MoE [35] sparsifies it with a mixture-of-experts. Beyond these, the design space broadens through decoder-only autoregressive models [18, 37, 41, 44], encoder-style representation models [20], further foundation models [6, 9, 17, 36, 46], and even the repurposing of the vision domain for forecasting [7]. Recently, Toto-2.0 [25] pushes this paradigm into the scaling era with a family of open models trained at multiple sizes. A common characteristic of most of these models is their reliance on *self-attention*, whose computational cost grows quadratically with the sequence length and the number of variates. EXAONE Finance departs from this paradigm by removing the attention mechanism, while keeping the patch-based, quantile-decoding interface that these models share. Even the few TSFMs built for finance retain an attention backbone. FinCast [53] adopts a channel-independent transformer that models series in isolation, and Delphyne [12] relies on quadratic any-variate attention, whereas EXAONE Finance mixes grouped variates jointly in *linear* time.

2.2 Attention-Free Sequence Models for Time Series

Attention is not the only way to mix information across time. Convolutional sequence models, from the original Temporal Convolutional Network (TCN) [5] to the more recent ModernTCN [38], use (dilated) causal convolutions to obtain large receptive fields at linear cost, and have proven competitive with transformers on long-horizon forecasting. Purely MLP-based models such as DLinear [50], TSMixer [8], and TiDE [10] show that even simple linear or token-/channel-mixing layers are strong baselines, and TTM (TinyTimeMixer) [14] carries this idea into the foundation-model regime with an MLP-Mixer backbone. State-space models, most notably Mamba [22] and its TS adaptations [27, 47], offer another linear-time alternative with selective recurrence. EXAONE Finance follows this attention-free lineage: it combines a *causal convolution* for temporal mixing with a *group-aware pooling MLP* for variate mixing. Unlike most prior attention-free work, trained per-dataset, it is a *pretrained foundation model*.

2.3 Large-scale Pretraining Datasets

Pretraining a TSFM requires large and diverse corpora, drawn from real archives, synthetic generators, or both. On the real side, the Monash archive [19] aggregates dozens of public datasets, LOTSA [48] scales this to over 27B observations across nine domains, GIFT-Eval [1] assembles a broad multi-domain corpus, and Time-300B [44] reaches over 300B time points. On the synthetic side, KernelSynth [3] composes Gaussian-process kernels to synthesize diverse trend and seasonal patterns, and CauKer [49] augments such kernel compositions with structural causal models for causally coherent inter-series interactions. In contrast, EXAONE Finance stays synthetic but targets a domain: its generator is built so that the sampled series reproduce the properties of financial markets (Section 5.1).

3 Problem Definition

Time series forecasting. Let a single time series be $\boldsymbol{x} = (x_1, x_2, \ldots)$ observed at a fixed sampling *frequency* (e.g., daily, weekly, monthly). At a forecast origin t, the model is given a *context window* of the most recent L observations, $\boldsymbol{x}^{\text{ctx}} = (x_{t-L+1}, \ldots, x_t) \in \mathbb{R}^L$, and must predict the next H values, called the *forecast horizon*, $\boldsymbol{y} = (x_{t+1}, \ldots, x_{t+H}) \in \mathbb{R}^H$.

Multivariate (grouped) series. Financial assets are frequently observed as aligned multi-channel panels (e.g., the open/high/low/close/volume fields of one security, or a group of related assets). We therefore allow a *group* of C variates sharing common timestamps, $\boldsymbol{X}^{\text{ctx}} = [\boldsymbol{x}_1^{\text{ctx}}; \ldots; \boldsymbol{x}_C^{\text{ctx}}] \in \mathbb{R}^{C \times L}$, with a group-id vector $\boldsymbol{g} \in \{1, \ldots, G\}^C$ indicating which variates may exchange information ($g_i = g_j$ iff variates i, j belong to the same group). A univariate series is the special case $C = 1$.

Table 2: Notations for TS forecasting.

Symbol	Meaning
L	Context (Input) length
H	Forecast horizon length
C	Number of variates in a group
B	Batch size (# variate-rows)
P	Patch size
d	Model hidden dimension
$\boldsymbol{x}^{\text{ctx}} \in \mathbb{R}^L$	Univariate context input
$\boldsymbol{X}^{\text{ctx}} \in \mathbb{R}^{C \times L}$	Multivariate context input
$\boldsymbol{y} \in \mathbb{R}^H$	Ground-truth horizon values
$\boldsymbol{m} \in \{0,1\}^L$	Observation mask
$\boldsymbol{g} \in \{1, \ldots, G\}^C$	Variate group ids
$q_1, \ldots, q_Q$	Target quantile levels
$\hat{\boldsymbol{Y}} \in \mathbb{R}^{Q \times H}$	Predicted quantiles

Missingness. Financial series contain missing spans (e.g., market closures, trading halts, and unreported periods). We represent observability with an observation mask $\boldsymbol{m} \in \{0,1\}^L$, where $m_\tau = 0$ marks a missing/padded entry. The model must handle such missing spans and forecast through them without imputation.

Probabilistic objective. Rather than a single point estimate, we produce a set of Q quantile forecasts. Given target quantile levels $q_1, \ldots, q_Q \in (0,1)$, the model $f_{\boldsymbol{\theta}}$ maps the (masked) context to

$$\hat{\boldsymbol{Y}} = f_{\boldsymbol{\theta}}\big(\boldsymbol{X}^{\text{ctx}}, \boldsymbol{m}, \boldsymbol{g}\big) \in \mathbb{R}^{Q \times H}, \qquad \hat{y}_{q,h} \approx \text{Quantile}_q(x_{t+h}), \tag{1}$$

where $\hat{y}_{q,h}$ estimates the q-th quantile of the h-step-ahead value. Table 2 summarizes the notation.

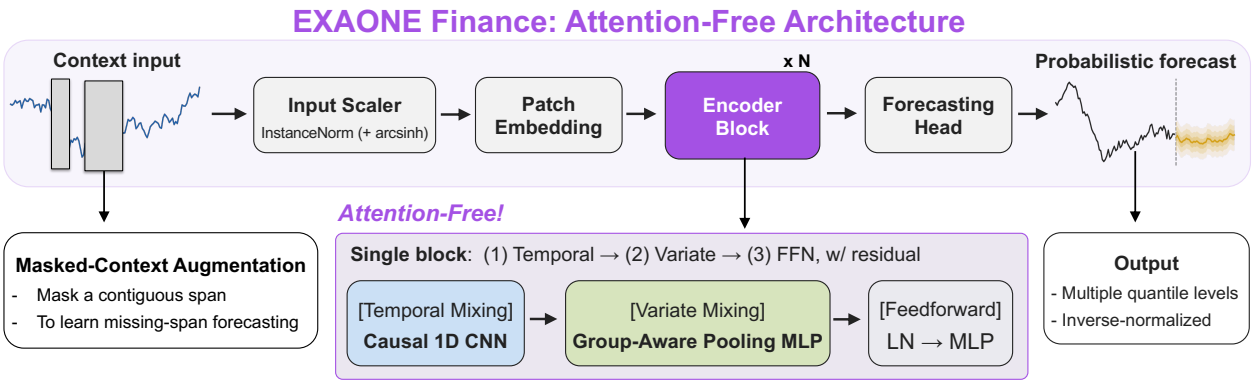

Figure 2: **Architecture of EXAONE Finance.** The context series is instance-normalized, split into patches, and embedded, then processed by N *attention-free* encoder blocks before a forecasting head produces a probabilistic forecast. Each encoder block mixes information with two linear-time operators: a *causal 1D convolution* for temporal mixing and a *group-aware pooling MLP* for variate mixing, followed by a feed-forward network. During training, a *masked-context augmentation* marks a contiguous input span as known-missing, so the model learns to forecast through the missing spans that occur at inference.

4 EXAONE Finance

4.1 Model Overview

EXAONE Finance is an attention-free TSFM that replaces self-attention with two simple yet effective linear-time mixers, where each encoder block combines (i) a *causal 1D convolution* over the temporal axis (Section 4.3) and (ii) a *group-aware pooling MLP* over the variate axis (Section 4.4), followed by a standard feed-forward network (FFN). This makes the per-block cost linear in both the sequence length L and the number of variates C. Missing observations are handled natively using an observation-mask channel that flags missing or padded entries in each patch token (Section 4.2). Instance normalization (Eq. 3) and the training loss (Section 4.6) ignore these positions, allowing the model to process and forecast through missing spans without a separate imputation step. The end-to-end pipeline is summarized as follows:

$$\boldsymbol{x}^{\mathrm{ctx}} \xrightarrow{\text{InstanceNorm}} \hat{\boldsymbol{x}} \xrightarrow{\text{Patch}} \boldsymbol{P} \xrightarrow{\text{Embed}} \boldsymbol{H}^{(0)} \xrightarrow{N\times\text{Encoder}} \boldsymbol{H}^{(N)} \xrightarrow{\text{Head}} \hat{\boldsymbol{Y}}. \tag{2}$$

We detail the input pipeline (Section 4.2), the two mixing operators (Sections 4.3–4.4), a masked-context training augmentation (Section 4.5), and the forecasting head (Section 4.6). The overall framework of EXAONE Finance is illustrated in Figure 2.

4.2 Input Pipeline

Instance normalization. Each context is standardized along the time axis using statistics:

$$\hat{\boldsymbol{x}} = \frac{\boldsymbol{x}^{\mathrm{ctx}} - \mu}{\sigma}, \qquad \mu = \mathrm{mean}(\boldsymbol{x}^{\mathrm{ctx}}),\ \sigma = \mathrm{std}(\boldsymbol{x}^{\mathrm{ctx}}), \tag{3}$$

where μ and σ are computed over the *observed* (non-missing) entries only and used to restore the predictions to the original scale. To accommodate the heavy-tailed nature of financial series, an area hyperbolic sine (arcsinh) transform is then applied to the standardized context:

$$\tilde{\boldsymbol{x}} \;=\; \mathrm{arcsinh}(\hat{\boldsymbol{x}}) \;=\; \log\!\left(\hat{\boldsymbol{x}} + \sqrt{\hat{\boldsymbol{x}}^2 + 1}\right). \tag{4}$$

This transform behaves linearly near zero ($\mathrm{arcsinh}(u) \approx u$) and logarithmically in the tails ($\mathrm{arcsinh}(u) \approx \mathrm{sgn}(u)\log 2|u|$), compressing extreme values while preserving sign. Unlike a log transform, it remains defined for negative and zero inputs. It reduces to the identity ($\tilde{\boldsymbol{x}} = \hat{\boldsymbol{x}}$) when disabled, and is inverted at inference by $\hat{\boldsymbol{x}} = \sinh(\tilde{\boldsymbol{x}})$ before de-standardization.

Patching. The transformed context $\tilde{\boldsymbol{x}}$ is unfolded into non-overlapping patches of size P (stride P). When the length is not a multiple of P, the sequence is left-padded with missing markers so that the most recent observations remain patch-aligned. Each patch token concatenates three components,

$$\boldsymbol{p}_\tau = \left[\, \boldsymbol{e}_\tau \,\|\, \tilde{\boldsymbol{x}}_\tau \,\|\, \boldsymbol{m}_\tau \,\right] \in \mathbb{R}^{3P}, \tag{5}$$

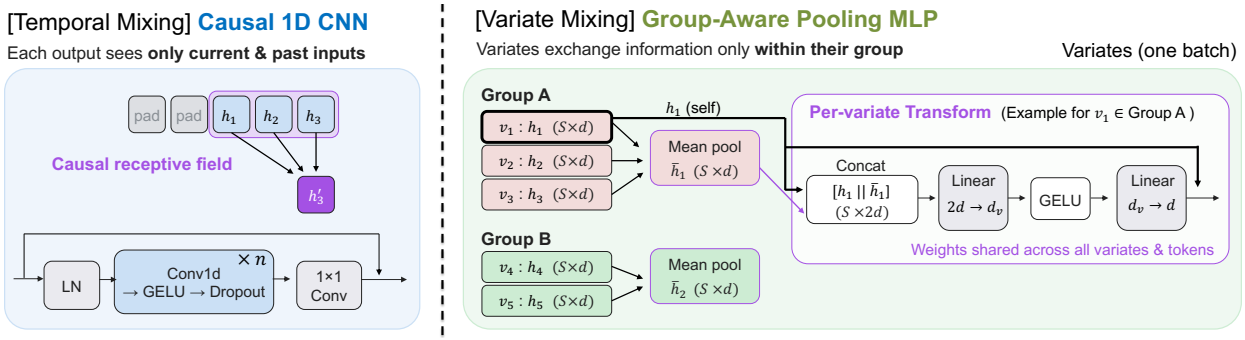

Figure 3: **The two attention-free mixing operators.** Each encoder block mixes information along two axes. [Left] *Temporal mixing* applies a causal 1D convolution, so that each position attends only to current and past inputs. [Right] *Variate mixing* is a group-aware pooling MLP, where variates exchange information only within their group, and each variate $\boldsymbol{h}_i$ is concatenated with its group mean $\bar{\boldsymbol{h}}_i$ and passed through a shared MLP with a residual connection.

a time encoding $\boldsymbol{e}_\tau$ (each position's index relative to the forecast origin, divided by a fixed scale), the patched values $\tilde{\boldsymbol{x}}_\tau$, and an observation mask $\boldsymbol{m}_\tau$ that marks padded/missing entries. Each token is then embedded into the model dimension by a residual MLP block: it takes the $3P$-dimensional token, passes it through a hidden layer of width d_{ff}, and outputs a d-dimensional representation. Stacking the embedded tokens gives the encoder input $\boldsymbol{H}^{(0)} \in \mathbb{R}^{T\times d}$, where T is the number of patches. An optional learnable *register* token is then prepended to this sequence.

4.3 Temporal Mixing

Causal convolution block. Self-attention along time is replaced by a stack of n causal 1D convolutions. Given block input $\boldsymbol{H} \in \mathbb{R}^{T\times d}$, we layer-normalize, transpose to channel-first, and apply each convolution with a left-only pad of $k-1$ so that token τ depends solely on tokens $\le \tau$ (ensuring causality):

$$\boldsymbol{U} = \text{GELU}\big(\text{Conv1d}_k(\text{LeftPad}_{k-1}(\cdot))\big), \quad \text{(repeated } n \text{ times)}, \tag{6}$$

where the first convolution maps $d \to c$ channels and the rest operate in c. A final 1×1 convolution restores the model width $c \to d$, and a residual connection is added:

$$\boldsymbol{H}' = \boldsymbol{H} + \text{Dropout}\big(\text{Conv1d}_{1\times1}(\boldsymbol{U})\big). \tag{7}$$

4.4 Variate Mixing

Group-aware pooling MLP. Information is exchanged across variates only *within the same group*, so that unrelated series are never mixed. Let g_i denote the group id of variate i and define the group mask $\Gamma_{ij} = \mathbb{1}[g_i = g_j]$. Each variate is augmented with its group-mean representation,

$$\bar{\boldsymbol{h}}_i = \frac{\sum_j \Gamma_{ij}\boldsymbol{h}_j}{\sum_j \Gamma_{ij}}, \qquad \boldsymbol{h}'_i = \boldsymbol{h}_i + \text{Dropout}\big(\text{MLP}([\boldsymbol{h}_i \,\|\, \bar{\boldsymbol{h}}_i])\big), \tag{8}$$

where the MLP projects $2d \to d_v \to d$ with a GELU nonlinearity. Since each variate is mixed only through its group mean, the output does not depend on the order of variates within a group. When a group contains only a single variate, no cross-variate information is available, and the operator reduces to a per-variate transformation. After the two mixers, a standard FFN is applied, that is, layer normalization (LN) followed by a two-layer MLP with hidden dimension d_{ff}, with a residual connection. Both mixers are shown in Figure 3.

4.5 Masked-Context Augmentation

Robustness to missingness. Financial series are routinely interrupted by market closures, trading halts, and unreported periods. To make the encoder robust to such missing spans, during training we randomly mark a single contiguous span of each context as *missing* before normalization,

$$\tilde{x}_\tau = \begin{cases} \text{missing}, & \tau \in [s, s+w) \\ x_\tau, & \text{otherwise}, \end{cases} \qquad w \sim \mathcal{U}\{1, \dots, \rho L\}, \tag{9}$$

with masking fraction $\rho = 0.25$ and a uniformly sampled start s. Since the normalization statistics, the observation mask, and the loss all treat missing entries consistently as "known-missing," the model learns to forecast through missing spans. Note that the augmentation is applied *only at training time*.

Comparison with masking in prior works. Most existing masking schemes in TS are reconstruction objectives [13, 30, 51], in which the masked span is itself the *prediction target*, i.e., masking is applied to the *output* to be reconstructed. In contrast, our augmentation masks part of the *input* rather than the target: the masked span is marked known-missing and excluded from the loss, so that the model becomes robust to the missing observations it encounters at inference, which are common in financial markets (e.g., market closures and trading halts).

4.6 Forecasting Head

Quantile decoding. After N encoder blocks and a final LN, the last $\lceil H/P \rceil$ tokens are decoded by the forecasting head (a residual MLP) into $Q \cdot P$ outputs and rearranged into quantile trajectories $\hat{\boldsymbol{Y}} \in \mathbb{R}^{Q \times H}$, which are mapped back to the original scale by inverting Eq. 3. We predict a set of Q quantile levels and train with the pinball (quantile) loss, evaluated only at finite target positions via a mask $\boldsymbol{m}^y$:

$$\mathscr{L} = \frac{1}{|\boldsymbol{m}^y|} \sum_q \sum_h m_h^y \max\big(q\,(y_h - \hat{y}_{q,h}),\ (q-1)(y_h - \hat{y}_{q,h})\big). \tag{10}$$

Masking the loss lets multivariate groups feed all channels as *inputs* while scoring only the designated target channel(s), matching the evaluation protocol.

4.7 Configuration, Inference, and Complexity

Model configuration. Table 3 summarizes the hyperparameters of the released model. The encoder stacks N=12 blocks of width d=1024, each pairing a temporal CNN (c=512 channels, kernel k=7, n=2 layers) with a group-aware variate MLP (hidden size d_v=512, up to G=16 groups). The forecasting head reads out patches of size P=16 as Q=21 quantile levels.

Single forward pass. At inference, EXAONE Finance decodes the entire horizon in *one* forward pass rather than autoregressively. For a requested horizon H, the model appends $\lceil H/P \rceil$ future tokens to the context, runs the N encoder blocks once, and reads out all Q quantiles for every horizon step simultaneously (Section 4.6). No step-by-step roll-out and no key/value cache are needed, as there is no attention to cache. Arbitrary horizons are supported by varying the number of output tokens, and the prediction is mapped back to the original scale by inverting Eq. 3. Missing or padded entries in the context are handled by the same observation mask used in training, so no special preprocessing (e.g., imputation) is required at inference. This scheme is identical during pretraining: the $\lceil H/P \rceil$ future tokens are all appended up front and predicted jointly in a single pass.

Table 3: Architecture configuration.

Hyperparameter	Value
Encoder	
Hidden dimension (d)	1024
Feed-forward dimension (d_{ff})	4096
Number of blocks (N)	12
Dropout	0.1
Encoder block	
Temporal CNN	
Channels (c)	512
Kernel size (k)	7
Conv layers (n)	2
Variate MLP	
Hidden dimension (d_v)	512
Max groups (G)	16
Patch & head	
Patch size (P)	16
Max input length (L)	512
Quantile levels (Q)	21

Computational complexity. Both mixers are linear in their respective axes. A causal convolution block costs $\mathcal{O}(L\,d\,c\,k)$ for sequence length L, and the group-aware pooling MLP costs $\mathcal{O}(C\,d)$ for C variates: each group mean is computed once by averaging within each group (written with a group mask in Eq. 8 only for clarity), not by a pairwise interaction. This contrasts with the $\mathcal{O}(L^2 d)$ time-axis attention and $\mathcal{O}(C^2 d)$ variate-axis attention in TSFMs that adopt self-attention, making EXAONE Finance linear in both the sequence length and the number of variates.

5 Experiments

We evaluate EXAONE Finance on **FinVerse**, a standardized, unified benchmark covering a broad range of financial assets. We first describe the pretraining dataset (Section 5.1) and the evaluation protocol, namely the three scoring tiers and their metrics (Section 5.2), then the baselines and aggregation (Section 5.3), the main results (Section 5.4), and an analysis of performance broken down by asset scope (Section 5.5).

Table 4: **Financial time series properties and their generating mechanisms.** Each property is induced by an explicit component of the sampling process. Properties are switched on independently with sampled intensities.

Family	Property	Generating mechanism
Temporal dependence	Trend	Deterministic drift plus a random-walk drift component
	Seasonality / cyclicality	Fourier components at calendar periods
	Autocorrelation	ARMA innovations
	Mean reversion	Ornstein–Uhlenbeck component
	Momentum	Positive short-horizon dependence in the drift
	Long-range dependence	Fractional Gaussian noise with sampled Hurst exponent
	Multi-scale dynamics	Superposition of components at several time scales
Conditional variance	Volatility	Sampled base volatility level
	Volatility clustering	GARCH-type conditional variance
	Heteroskedasticity	Time-varying variance schedules
Marginal distribution	Fat tails	Student-t innovations with sampled degrees of freedom
	Skewness	Skewed innovations and a leverage effect
	Non-normality	Mixture-of-normals innovation distribution
Discontinuity and regimes	Jump / discontinuity	Compound-Poisson jump component
	Shock / extreme event	Rare large-amplitude shocks with decaying response
	Regime switching	Markov-switching parameters
	Structural break	Change-points in the parameter path
Cross-series structure	Cross-asset correlation	Common factors with sampled loadings
	Time-varying correlation	Dynamic conditional correlation of the factors
	Lead–lag relationship	Lagged factor loadings across channels
	Cointegration	Shared stochastic trend with a stationary spread
Observation	Liquidity / activity	Intermittent observation and activity-driven volume proxies

5.1 Pretraining Dataset

Data mixture. We pretrain EXAONE Finance on a synthetic corpus designed around the nature of financial data. It combines a domain-agnostic source, KernelSynth [3], with a *synthetic financial* source whose sampling process we design so that the generated series carry the statistical regularities of financial markets. EXAONE Finance therefore sees *no real market data at any point during pretraining*, yet transfers to real assets at evaluation.

Synthetic financial datasets. Table 4 lists the properties we ask the generator to reproduce, grouped into six families, together with the mechanism that induces each. We sample a series as a superposition of these components: a trend and cyclical part sets the low-frequency shape, an ARMA / Ornstein–Uhlenbeck part sets the short-horizon dependence, a stochastic-volatility part with heavy-tailed, skewed innovations sets the conditional distribution, a jump and regime part introduces discontinuities and parameter changes, and a factor part ties channels together so that a group shares common shocks with lead–lag and cointegration structure. We switch every property on with a randomly sampled intensity. The corpus therefore spans series close to Gaussian random walks as well as series dominated by clustered volatility and abrupt regime changes. Figure 4 shows a representative property of each family as it appears in a sampled series, next to the reference behaviour that the corresponding component departs from.

Frequency-aware windowing. Synthetic financial series are generated at the frequencies of Table 5 and windowed with the geometry listed there, which is the same geometry used at evaluation. KernelSynth series carry no frequency, so they instead use a variable context length up to L and a horizon sampled uniformly up to $H_{\max}$, broadening the pretraining distribution. In this run every series is treated as an individual channel. The group-aware mixer (Section 4.4) natively supports richer cross-series grouping through the data pipeline, such as the OHLCV fields of a single security.

Table 5: Frequency-aware windows.

Frequency	L	H
Daily	260	{5, 20, 65, 130, 260}
Weekly	52	{1, 4, 12, 26, 52}
Monthly	12	{1, 3, 6, 12, 24}
Quarterly	4	{1, 2, 4, 8}
Annual	10	{1, 2, 5, 10}

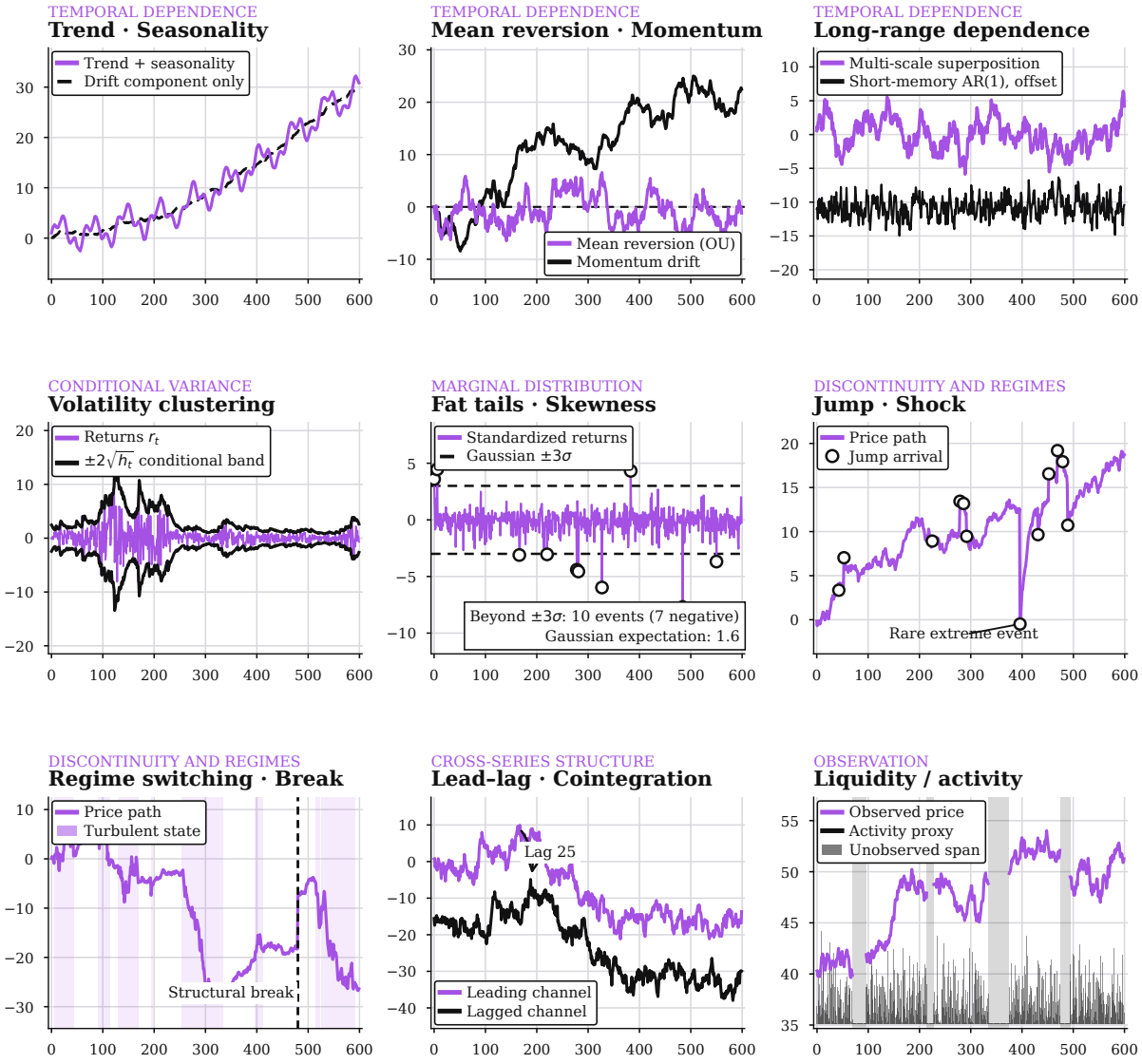

Figure 4: **Properties of the synthetic pretraining corpus.** Each panel simulates one property of Table 4 with the generator component that induces it. The purple curve is the series that shows the property. The black curve, band, or markers are there for comparison, and what they mean changes from panel to panel, so each panel has its own legend.

5.2 Evaluation Protocol: Tiers and Metrics

Why finance-specific metrics? Standard forecasting metrics such as the mean absolute error (MAE) measure average pointwise error and are scale-dependent. On a heterogeneous financial panel, a few high-variance series dominate the average error and reward predicting the raw level rather than the actionable signal. Financial decisions, however, hinge on different quantities: the *direction* of the next move, the *relative ordering* of assets at a given time, and the *risk-adjusted return* of a strategy built on the forecasts. FinVerse is therefore designed around these finance-specific desiderata and organizes evaluation into three tiers. Tier 1 uses scale-free, change-based accuracy. Tier 2 measures cross-sectional skill. Tier 3 measures realized economic value through a portfolio backtest.

Change-based targets. Tiers 1–2 are computed on a grid of a *change period* (how far back the change is measured) and a *forecast position* (how far ahead of the origin it is evaluated). For an origin t, denoting the change period by p

Table 6: **Tier metrics.** This table lists the metrics used by the three benchmark tiers.

Tier	Metric	Meaning
Tier 1 Point accuracy	MASE ↓	Mean absolute error scaled by an in-context naïve forecast
	HR / HR′ / HR″ ↑	Directional accuracy (Sign of change, stricter variants)
	MASE (Diff / raw) ↓	Differenced and raw-value MASE variants
Tier 2 Cross-sectional IC	IC ↑	Cross-sectional Spearman correlation of predicted vs. realized change
Tier 3 Portfolio backtest	Return ↑	Annualized return of the top-K% long portfolio
	Sharpe ↑	Annualized return divided by volatility
	Volatility ↓	Annualized volatility of the strategy
	MDD ↑	Maximum peak-to-trough loss (Closer to 0 is better)

and the forecast position by h, the *predicted* and *realized* changes are

$$\Delta^{\text{pred}} = \frac{\hat{y}_{t+h}}{x_{t-p}} - 1, \qquad \Delta^{\text{act}} = \frac{x_{t+h}}{x_{t-p}} - 1, \tag{11}$$

where x denotes the ground-truth series, so the base value x_{t-p} is measured a period p before the origin and both changes run to the same future point $t+h$. These changes underlie all three tiers.

The three tiers, whose metrics are summarized in Table 6, are defined as follows.

- **Tier 1 (Point accuracy).** Per series, we report (i) **MASE**, the mean absolute error scaled by an in-context naïve forecast, in both a relative-change and an absolute-difference variant, plus a raw all-position variant, and (ii) the **hit rate (HR)**, the directional accuracy $\text{HR} = \text{mean}\big[\,\text{sign}(\Delta^{\text{pred}}) = \text{sign}(\Delta^{\text{act}})\big]$, together with stricter variants HR′/HR″ that additionally require the sign of the first/second difference to match.
- **Tier 2 (Cross-sectional Information Coefficient, IC).** For each origin t, we rank assets within a subgroup by predicted change and compute the Spearman rank correlation against the realized change,

$$\text{IC}(t) = \text{SpearmanCorr}\big(\{\Delta_s^{\text{pred}}(t)\}_s, \{\Delta_s^{\text{act}}(t)\}_s\big), \qquad \text{IC} = \text{mean}_t\,\text{IC}(t). \tag{12}$$

  IC measures whether a model correctly orders assets cross-sectionally, which is the quantity that matters for relative-value strategies.
- **Tier 3 (Portfolio backtest).** We form a top-K%, equal-weight, long-only portfolio by selecting the assets with the highest predicted change, rebalanced at horizon h, in a multi-start overlapping backtest. We report the annualized **Return**, **Sharpe** ratio, **Volatility**, and **Maximum Drawdown (MDD)**. This tier turns forecasts into an executable strategy and scores the economic value of the predictions.

Overall ranking. We rank models *per cell*: for every (tier, spec, metric) triple, all M models are ranked from 1 (best) to M. We aggregate these ranks with the *geometric mean* rather than the arithmetic mean because it is scale-free and rewards *consistent* performance across heterogeneous cells, since a single very poor rank is penalized proportionally rather than absolutely. A model's score on a tier is

$$\bar{r} = \Big(\prod_c r_c\Big)^{1/|\mathscr{C}|} = \exp\Big(\tfrac{1}{|\mathscr{C}|}\textstyle\sum_c \ln r_c\Big), \tag{13}$$

where r_c is the model's rank on cell c and $\mathscr{C}$ is that tier's set of cells. The headline *rank-sum* is the sum of the three per-tier placements (lower is better).

5.3 Baselines and Aggregation

We compare against **43 baselines** covering the major public TSFM families, including Chronos [3] / Chronos-Bolt / Chronos-2 [2], Moirai (1.1, 2.0) [48], TimesFM (1.0/2.0/2.5) [11], Toto [9] / Toto-2.0 [25], TiRex (1.1, 2) [4], TTM [14], PatchTST-FM [40], FlowState [21], Timer [37], Sundial [36], Kairos [15], YingLong [45], Reverso [16], TempoPFN [39], VisionTS [7], CleanTS, PatchFM, Super-Linear, and T0-Alpha. These span univariate and multivariate-native APIs. Each model is scored on the same windows and ranked with the per-cell geometric-mean-rank procedure of Section 5.2 (Eq. 13). EXAONE Finance (Section 5.1) is thus evaluated against the baseline pool as a comparison set of 44 models. EXAONE Finance outputs $Q = 21$ quantile levels $\{0.01, 0.05, 0.1, \ldots, 0.9, 0.95, 0.99\}$, and all tier metrics are computed on the median forecast (q_{50}).

Table 7: **Main benchmark results on FinVerse.** Each tier gives the model's placement (#) and its *Avg. Rank* (The geometric mean of per-cell ranks, lower is better). **Rank Sum** is the sum of the three placements, and the 44 models are ordered from the lowest rank sum to the highest. EXAONE Finance is #1 in all three tiers, for a rank sum of 3.

#	Model	Tier 1		Tier 2		Tier 3		Rank Sum
		#	Avg. Rank	#	Avg. Rank	#	Avg. Rank	
1	**EXAONE Finance**	**1**	**6.53**	**1**	**7.45**	**1**	**7.43**	**3**
2	Chronos-2 (Synthetic)	9	11.68	3	8.74	2	9.90	14
3	Reverso (Small)	6	9.60	2	8.64	6	13.04	14
4	TiRex-1.1	2	6.98	4	9.71	14	15.76	20
5	Chronos-2	8	10.14	13	16.12	9	15.17	30
6	TimesFM-2.5-200M	7	9.96	5	10.66	20	17.10	32
7	Chronos-Bolt (Base)	11	12.88	15	16.48	7	14.54	33
8	Chronos-Bolt (Small)	12	12.94	12	15.85	16	16.10	40
9	T0-Alpha	13	13.47	25	18.73	3	11.11	41
10	TiRex-2 (Pretrain)	3	7.48	6	11.56	34	20.12	43
11	TiRex-2 (Zero-shot)	4	8.37	7	11.67	33	20.06	44
12	Toto-2.0-2.5B	18	15.74	17	16.78	10	15.19	45
13	VisionTS	10	12.05	8	13.00	29	19.31	47
14	PatchFM	5	9.08	14	16.15	30	19.56	49
15	Toto-2.0-4M	15	14.44	24	18.21	13	15.66	52
16	TimesFM-2.0-500M	39	29.32	10	14.38	4	11.12	53
17	TempoPFN	14	14.11	18	16.95	27	18.76	59
18	Kairos-50M	20	16.04	29	20.21	11	15.26	60
19	Toto-2.0-1B	16	15.00	23	17.96	22	17.74	61
20	FlowState (Granite R1)	21	17.00	27	19.12	15	16.09	63
21	TimesFM-1.0-200M	22	17.06	16	16.58	26	18.55	64
22	Super-Linear	24	18.11	11	15.15	31	19.58	66
23	Toto-2.0-22M	19	15.90	26	18.97	21	17.30	66
24	Kairos-10M	28	20.85	28	19.80	12	15.61	68
25	FlowState	23	17.49	30	20.22	17	16.34	70
26	Toto-2.0-313M	17	15.34	31	20.55	23	17.96	71
27	Timer-S1	25	18.45	9	13.50	39	22.81	73
28	Moirai-1.1-R (Large)	38	27.84	35	22.17	5	13.00	78
29	Toto-Open-Base-1.0	27	19.94	33	21.36	19	17.07	79
30	Moirai-1.1-R (Base)	37	27.45	38	24.21	8	14.57	83
31	Moirai-1.1-R (Small)	33	25.62	32	21.15	18	16.87	83
32	PatchTST-FM-R1	34	26.30	22	17.92	28	19.03	84
33	Sundial-Base-128M	26	19.73	20	17.66	41	23.48	87
34	PatchTST-FM (Granite R1)	40	29.35	19	17.34	32	20.05	91
35	Kairos-23M	29	21.69	40	24.41	24	18.35	93
36	TTM-R3	35	27.10	21	17.73	43	24.22	99
37	Moirai-2.0-R (Small)	31	23.87	34	21.91	36	21.39	101
38	TTM-R1	32	24.77	44	27.22	25	18.50	101
39	CleanTS-65M	36	27.13	36	22.70	35	21.19	107
40	TTM-R2	30	22.19	43	27.05	44	25.18	117
41	YingLong-50M	42	35.16	37	24.16	38	22.50	117
42	YingLong-300M	44	35.46	39	24.29	37	21.84	120
43	YingLong-110M	43	35.39	41	24.47	40	23.23	124
44	YingLong-6M	41	34.84	42	24.87	42	23.70	125

5.4 Main Results

Overall performance. Table 7 and Figure 5 present the headline result: *EXAONE Finance ranks #1 in all three tiers*, for a perfect rank-sum of 3, while the best baseline reaches only 14. The margin holds across very different objectives, namely point accuracy (Tier 1), cross-sectional ordering (Tier 2), and realized portfolio performance (Tier 3), indicating that the gains are not an artifact of any single metric. Notably, no single baseline is consistently strong across tiers. For example, TiRex [4] is strong on Tier 1 but drops on Tier 3, whereas EXAONE Finance leads every tier simultaneously. The two closest baselines, Chronos-2 (Synthetic) and Reverso (Small), never lead a tier.

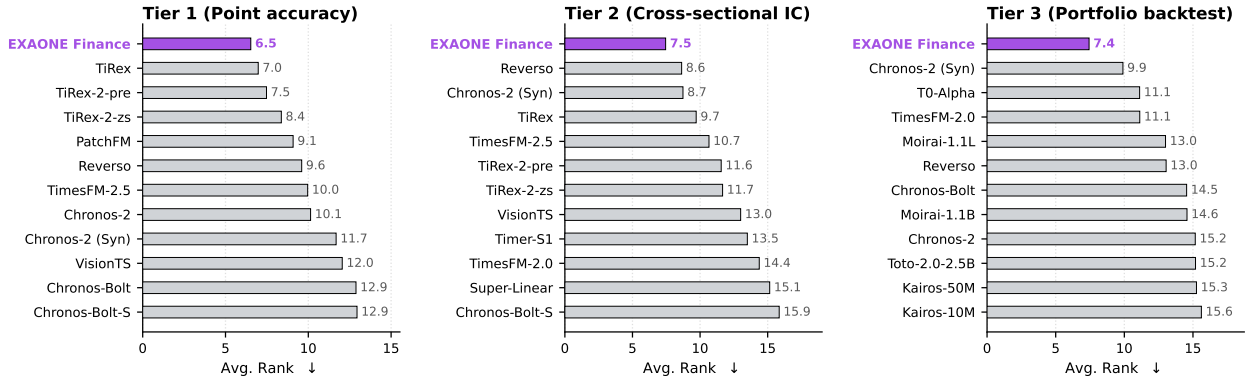

Figure 5: **Per-tier ranking.** Each panel ranks the twelve best models of one tier by their aggregate rank (The geometric mean of per-cell ranks). EXAONE Finance attains the lowest aggregate rank in all three tiers.

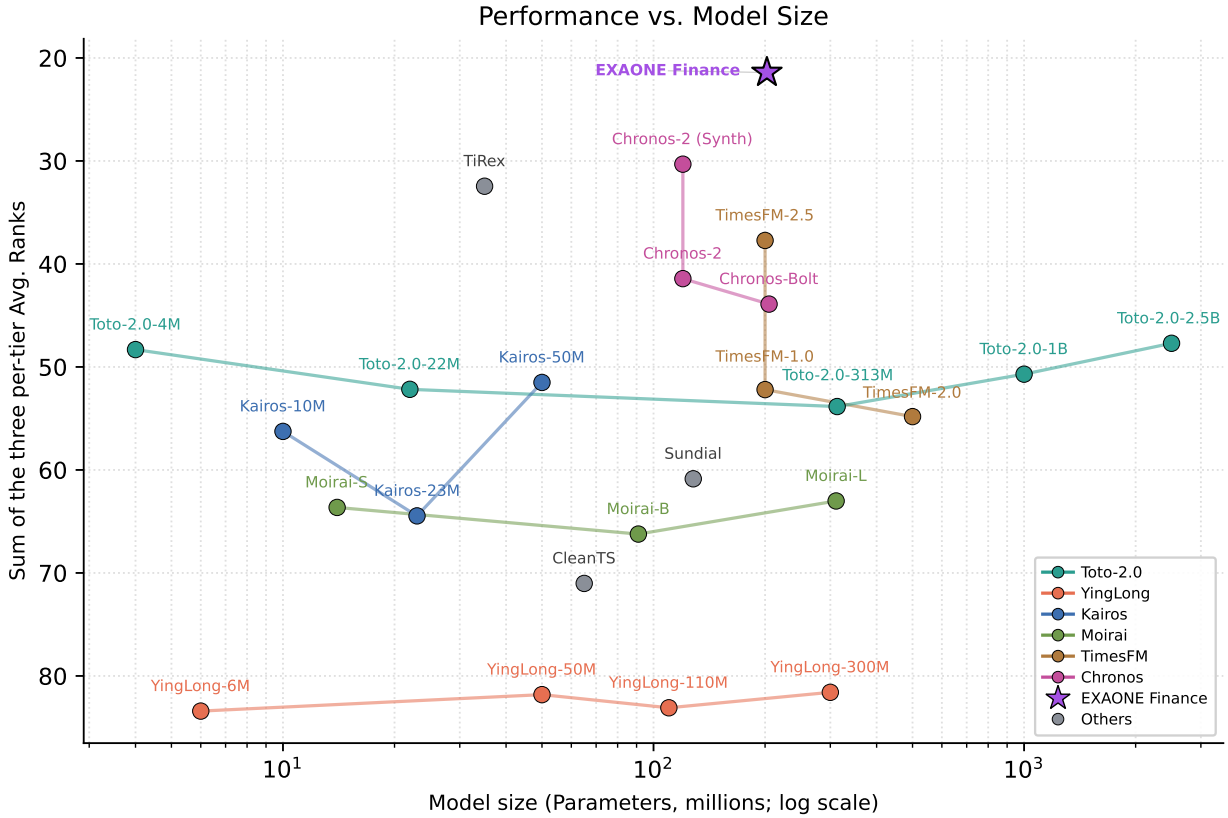

Figure 6: **Performance vs. model size.** Each point is a model, with parameter count on the x-axis (Log scale) and, on the y-axis, the sum of its three per-tier Avg. Ranks. Lower values are better, and the axis is inverted so that higher is better. EXAONE Finance attains the best aggregate rank at a modest size.

Performance vs. model size. Figure 6 plots each model's aggregate rank against its parameter count, where the aggregate is the sum of the model's three per-tier Avg. Ranks from Table 7. EXAONE Finance sits on the Pareto frontier: at only 202M parameters it attains the best overall rank, outperforming models more than an order of magnitude larger (e.g., the billion-scale Toto-2.0 [25] variants), which is consistent with its linear-time, attention-free design. The same trend appears across the other families in the figure, where a larger model size does not reliably improve the rank, suggesting that the scaling law common in general-domain forecasting does not yet hold in finance.

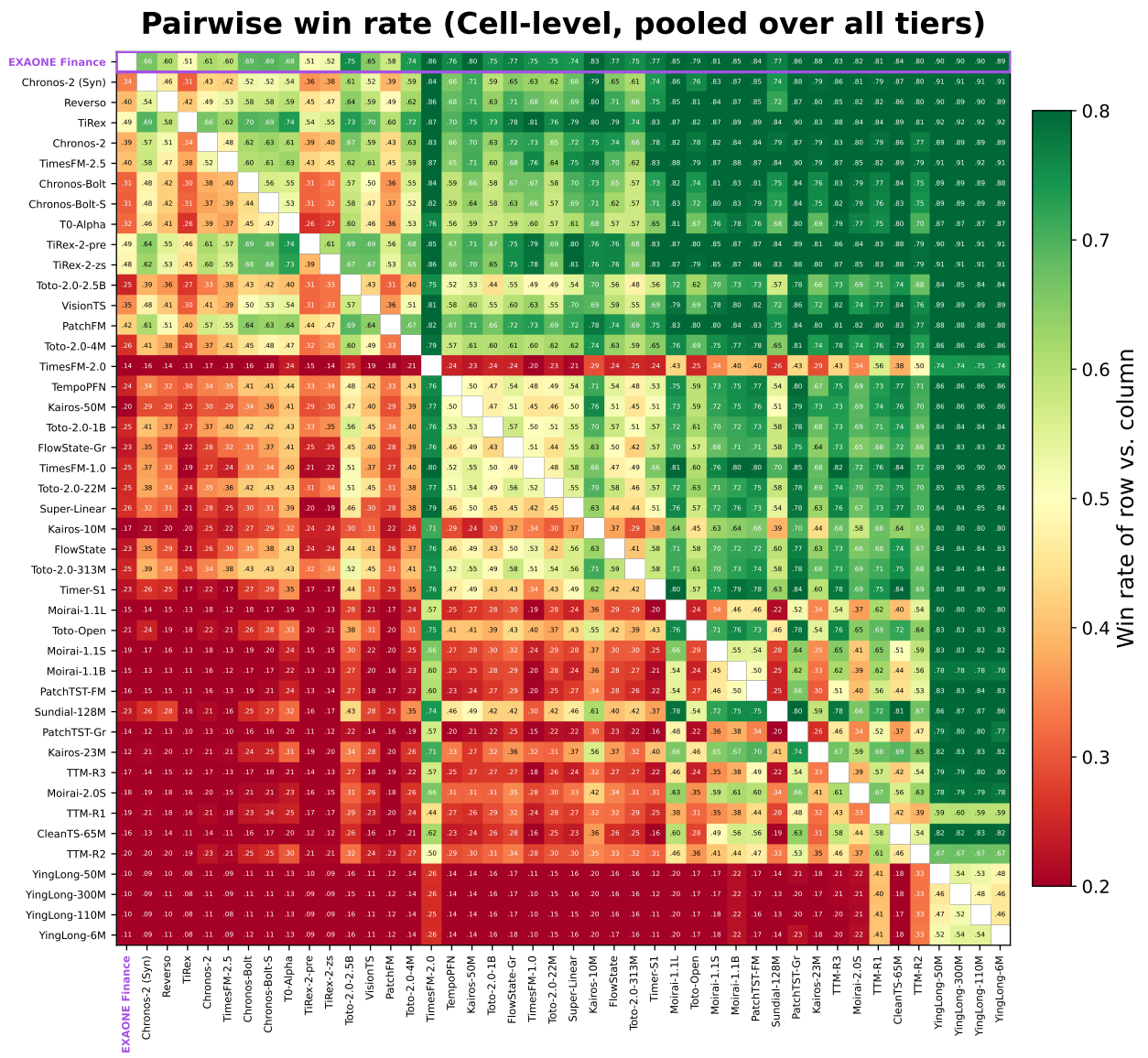

Figure 7: **Pairwise win rate.** Each cell reports the cell-level win rate of the row model against the column model, pooled over all (tier, spec, metric) cells. Models are ordered by rank-sum, as in Table 7, so EXAONE Finance is the first row and column. EXAONE Finance is the only model that beats every other model head-to-head.

Head-to-head dominance. To probe robustness of the ranking, we also compute the *pairwise* win rate between every pair of models: the fraction of shared (tier, spec, metric) cells on which one model beats the other. Ties are split evenly between the two models. Rows and columns are ordered by rank-sum, best first, as in Table 7. As shown in Figure 7, EXAONE Finance is the only model whose row is entirely above 0.5, i.e., it wins its head-to-head comparison against all 43 baselines (per-opponent win rate ranging from 0.51 to 0.90). Within that range, the narrowest margins are against the TiRex family (TiRex-2-pretrain and TiRex-1.1, both 0.51), the only baselines that come close, yet both still trail EXAONE Finance on every tier of Table 7.

5.5 Analysis by Asset Scope

Profile across all scopes. Because the benchmark spans a broad universe of asset classes, Figure 1 profiles the model as a percentile over the four data *scopes* (inter-country, country, sector, individual) crossed with the three tiers, where 100 denotes the best of the 44 models. EXAONE Finance traces a near-maximal envelope, scoring in the 95–100 percentile on ten of the eleven populated scope×tier cells.

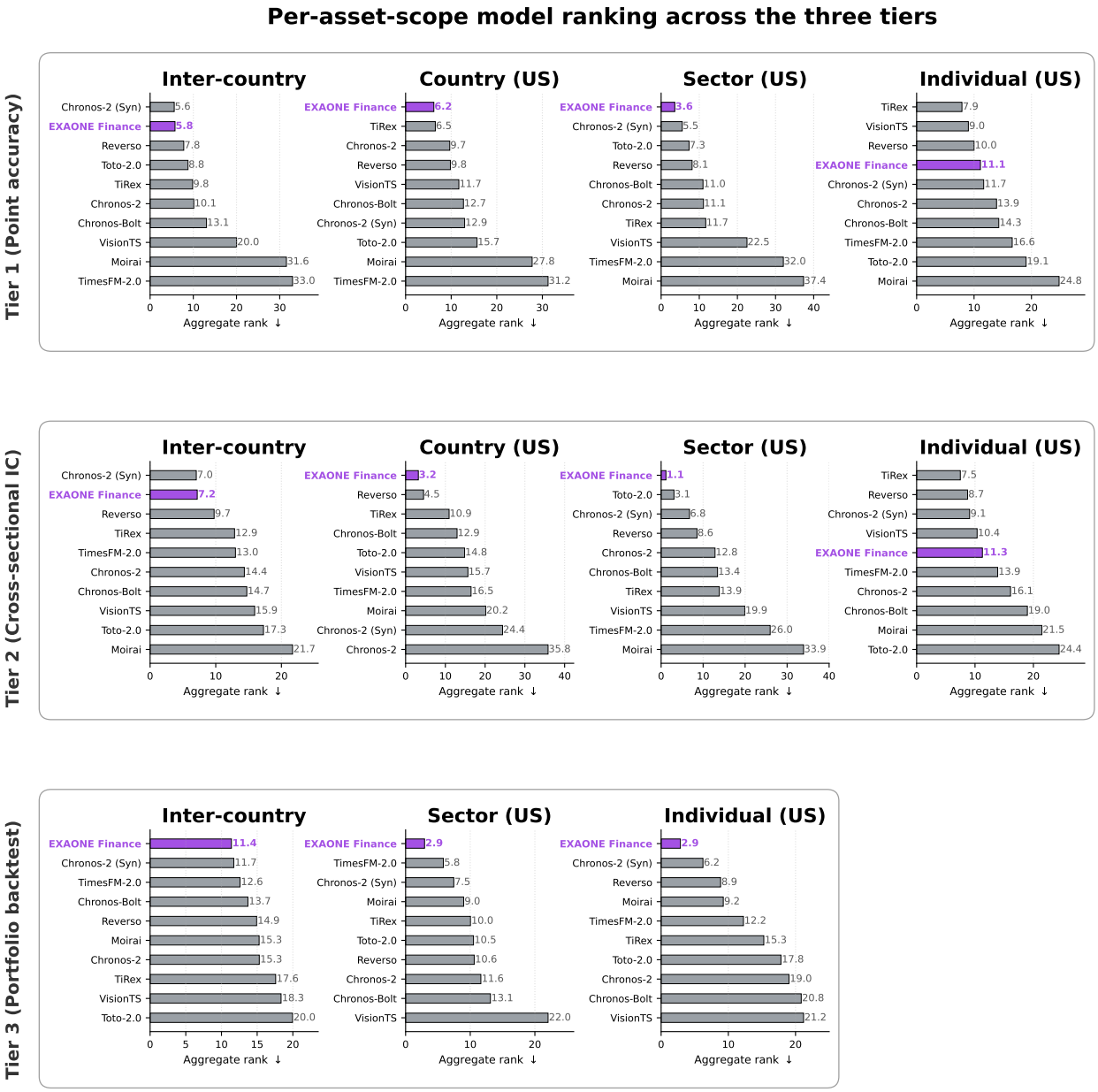

Figure 8: **Comparison of the top-10 models across the three tiers.** Each panel shows the aggregate rank (The geometric mean of per-(Spec × metric) cell ranks, where lower is better) within the comparison set for one asset scope. The rows correspond to Tier 1 (Point accuracy), Tier 2 (Cross-sectional IC), and Tier 3 (Portfolio backtest).

Ranking within each scope. To make the comparison concrete, Figure 8 breaks the ranking down model-by-model. Each row corresponds to one tier and contains a panel per asset scope, plotting the aggregate rank (lower is better) of EXAONE Finance against representative baselines. Across point accuracy (Tier 1), cross-sectional IC (Tier 2), and portfolio backtest (Tier 3), EXAONE Finance is top-3 in ten of the eleven scope×tier panels, with Chronos-2 (Synthetic), the 2nd-ranked model in Table 7, as its closest competitor.

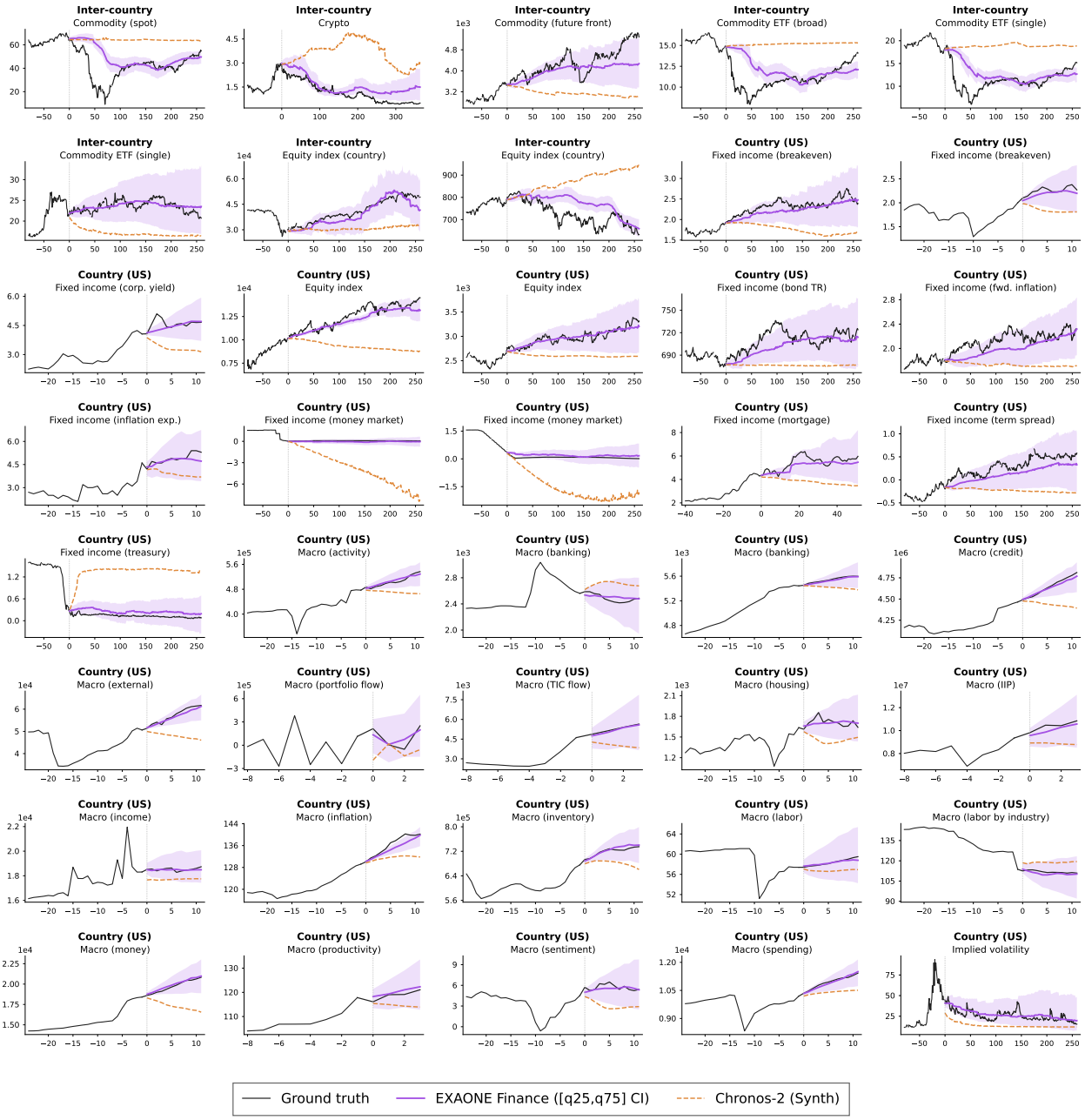

Figure 9: **Forecast visualizations across asset classes (Inter-country, Country).** Each panel shows a context history, the ground truth, the EXAONE Finance median forecast with its $[q_{25}, q_{75}]$ interval, and the median forecast of the 2nd-ranked Chronos-2 (Synthetic). EXAONE Finance follows the ground truth more closely than the 2nd-ranked model across diverse sectors.

Visualization of time series forecasts. Figures 9 and 10 visualize forecasts from EXAONE Finance across diverse asset classes, overlaid with the 2nd-ranked model, Chronos-2 (Synthetic). Our median forecast (purple) tracks the ground truth more closely than the 2nd-ranked model (orange), and the ground-truth value stays within the predicted $[q_{25}, q_{75}]$ interval in most windows. Notably, several series in Figure 10 contain missing spans in their context (visible as gaps in the ground-truth history), yet EXAONE Finance still forecasts through them accurately, consistent with its masked-context design. Overall, these visualizations mirror the quantitative results: EXAONE Finance is accurate and well calibrated across the diverse assets of a financial panel, even under missing observations.

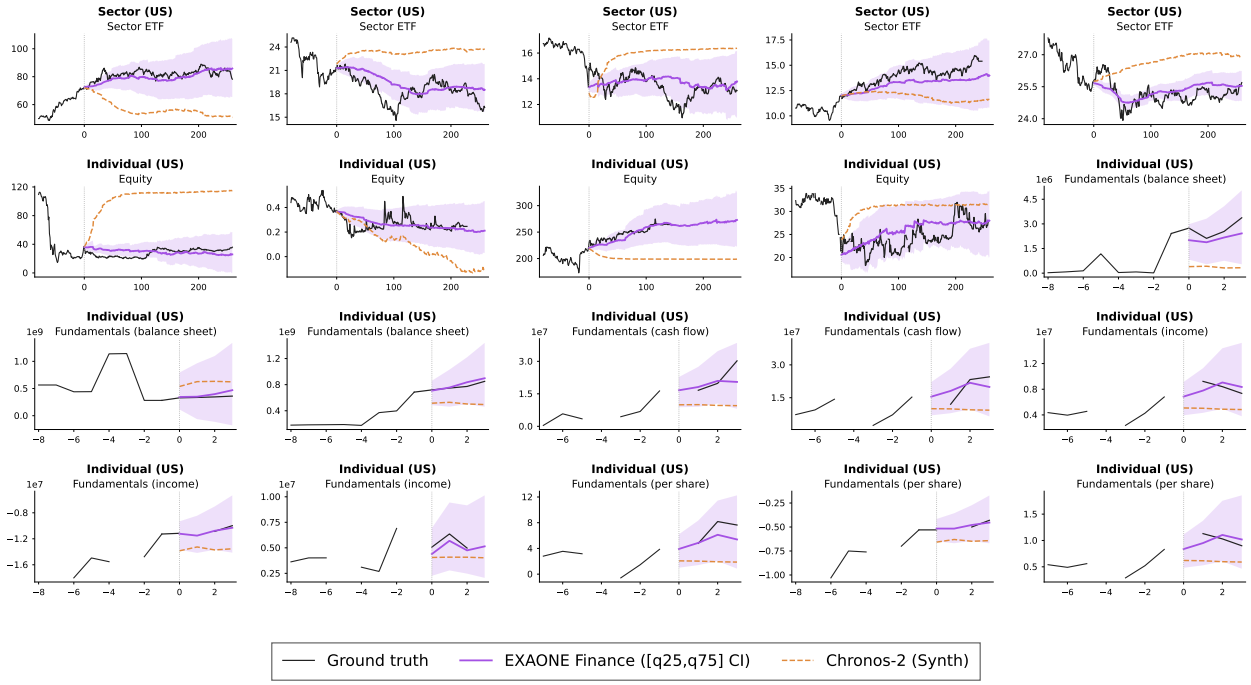

Figure 10: **Forecast visualizations across asset classes (Sector, Individual).** Same setup as Figure 9, for the sector and individual asset scopes.

6 Conclusion

We present **EXAONE Finance**, an attention-free time series foundation model for financial forecasting. By replacing self-attention with a causal 1D convolution for temporal mixing and a group-aware pooling MLP for variate mixing, the model attains linear-time cost in both sequence length and variate count while retaining a patch-based, multi-quantile decoding interface. A masked-context augmentation makes it robust to the missingness pervasive in financial markets. Pretrained on a *synthetic financial* corpus, EXAONE Finance ranks #1 in all three tiers of the FinVerse benchmark and is the only model that wins its head-to-head comparison against every baseline.

Limitations and future works.

- **Richer variate grouping.** The group-aware pooling mixer natively supports arbitrary cross-series grouping structures, from the multiple fields of a single security to related instruments within a portfolio. Exploring these grouping schemes, and scaling to broader multivariate financial panels, is a promising direction for further gains.
- **Multimodal extension.** Extending the model beyond numerical series to auxiliary modalities, such as the news, disclosures, and earnings reports that accompany a price series, is a natural next direction [24, 33, 34].
- **Broader-domain evaluation.** Results are on a financial benchmark, and broad general-domain benchmarking (e.g., GIFT-Eval [1]) is complementary future work, since transfer beyond finance remains untested.

References

[1] Taha Aksu, Gerald Woo, Juncheng Liu, Xu Liu, Chenghao Liu, Silvio Savarese, Caiming Xiong, and Doyen Sahoo. GIFT-Eval: A Benchmark For General Time Series Forecasting Model Evaluation. In *NeurIPS 2024 TSALM Workshop*, 2024.

[2] Abdul Fatir Ansari, Oleksandr Shchur, Jaris Küken, Andreas Auer, Boran Han, Pedro Mercado, Syama Sundar Rangapuram, Huibin Shen, Lorenzo Stella, Xiyuan Zhang, Mononito Goswami, Shubham Kapoor, Danielle C. Maddix, Pablo Guerron, Tony Hu, Junming Yin, Nick Erickson, Prateek Mutalik Desai, Hao Wang, Huzefa Rangwala, George Karypis, Yuyang Wang, and Michael Bohlke-Schneider. Chronos-2: From Univariate to Universal Forecasting. `https://arxiv.org/abs/2510.15821`, 2025.

[3] Abdul Fatir Ansari, Lorenzo Stella, Caner Turkmen, Xiyuan Zhang, Pedro Mercado, Huibin Shen, Oleksandr Shchur, Syama Sundar Rangapuram, Sebastian Pineda Arango, Shubham Kapoor, Jasper Zschiegner, Danielle C. Maddix, Hao Wang, Michael W. Mahoney, Kari Torkkola, Andrew Gordon Wilson, Michael Bohlke-Schneider, and Yuyang Wang. Chronos: Learning the Language of Time Series. *TMLR*, 2024.

[4] Andreas Auer, Patrick Podest, Daniel Klotz, Sebastian Böck, Günter Klambauer, and Sepp Hochreiter. TiRex: Zero-Shot Forecasting Across Long and Short Horizons with Enhanced In-Context Learning. In *NeurIPS*, 2025.

[5] Shaojie Bai, J. Zico Kolter, and Vladlen Koltun. An Empirical Evaluation of Generic Convolutional and Recurrent Networks for Sequence Modeling. `https://arxiv.org/abs/1803.01271`, 2018.

[6] Defu Cao, Furong Jia, Sercan O. Arik, Tomas Pfister, Yixiang Zheng, Wen Ye, and Yan Liu. TEMPO: Prompt-based Generative Pre-trained Transformer for Time Series Forecasting. In *ICLR*, 2024.

[7] Mouxiang Chen, Lefei Shen, Zhuo Li, Xiaoyun Joy Wang, Jianling Sun, and Chenghao Liu. VisionTS: Visual Masked Autoencoders Are Free-Lunch Zero-Shot Time Series Forecasters. In *ICML*, 2025.

[8] Si-An Chen, Chun-Liang Li, Nate Yoder, Sercan O. Arik, and Tomas Pfister. TSMixer: An All-MLP Architecture for Time Series Forecasting. *TMLR*, 2023.

[9] Ben Cohen, Emaad Khwaja, Kan Wang, Charles Masson, Elise Ramé, Youssef Doubli, and Othmane Abou-Amal. Toto: Time Series Optimized Transformer for Observability. `https://arxiv.org/abs/2407.07874`, 2024.

[10] Abhimanyu Das, Weihao Kong, Andrew Leach, Shaan Mathur, Rajat Sen, and Rose Yu. Long-term Forecasting with TiDE: Time-series Dense Encoder. *TMLR*, 2023.

[11] Abhimanyu Das, Weihao Kong, Rajat Sen, and Yichen Zhou. A decoder-only foundation model for time-series forecasting. In *ICML*, 2024.

[12] Xueying Ding, Aakriti Mittal, and Achintya Gopal. DELPHYNE: A Pre-Trained Model for General and Financial Time Series. `https://arxiv.org/abs/2506.06288`, 2025.

[13] Jiaxiang Dong, Haixu Wu, Haoran Zhang, Li Zhang, Jianmin Wang, and Mingsheng Long. SimMTM: A simple pre-training framework for masked time-series modeling. In *NeurIPS*, 2023.

[14] Vijay Ekambaram, Arindam Jati, Pankaj Dayama, Sumanta Mukherjee, Nam H. Nguyen, Wesley M. Gifford, Chandra Reddy, and Jayant Kalagnanam. Tiny Time Mixers (TTMs): Fast Pre-trained Models for Enhanced Zero/Few-Shot Forecasting of Multivariate Time Series. In *NeurIPS*, 2024.

[15] Kun Feng, Shaocheng Lan, Yuchen Fang, Wenchao He, Sihan Lu, Shuqi Gu, Lintao Ma, Xingyu Lu, and Kan Ren. Kairos: Toward Adaptive and Parameter-Efficient Time Series Foundation Models. `https://arxiv.org/abs/2509.25826`, 2025.

[16] Xinghong Fu, Yanhong Li, Georgios Papaioannou, and Yoon Kim. Reverso: Efficient Time Series Foundation Models for Zero-shot Forecasting. `https://arxiv.org/abs/2602.17634`, 2026.

[17] Shanghua Gao, Teddy Koker, Owen Queen, Thomas Hartvigsen, Theodoros Tsiligkaridis, and Marinka Zitnik. UniTS: A Unified Multi-Task Time Series Model. In *NeurIPS*, 2024.

[18] Azul Garza, Cristian Challu, and Max Mergenthaler-Canseco. TimeGPT-1. `https://arxiv.org/abs/2310.03589`, 2023.

[19] Rakshitha Godahewa, Christoph Bergmeir, Geoffrey I. Webb, Rob J. Hyndman, and Pablo Montero-Manso. Monash Time Series Forecasting Archive. In *NeurIPS Datasets and Benchmarks Track*, 2021.

[20] Mononito Goswami, Konrad Szafer, Arjun Choudhry, Yifu Cai, Shuo Li, and Artur Dubrawski. MOMENT: A Family of Open Time-series Foundation Models. In *ICML*, 2024.

[21] Lars Graf, Thomas Ortner, Stanisław Woźniak, and Angeliki Pantazi. FlowState: Sampling-Rate-Equivariant Time-Series Forecasting. In *ICML*, 2026.

[22] Albert Gu and Tri Dao. Mamba: Linear-Time Sequence Modeling with Selective State Spaces. In *COLM*, 2024.

[23] Tao Hong and Shu Fan. Probabilistic Electric Load Forecasting: A Tutorial Review. *International Journal of Forecasting*, 32(3):914–938, 2016.

[24] Ming Jin, Shiyu Wang, Lintao Ma, Zhixuan Chu, James Y. Zhang, Xiaoming Shi, Pin-Yu Chen, Yuxuan Liang, Yuan-Fang Li, Shirui Pan, and Qingsong Wen. Time-LLM: Time Series Forecasting by Reprogramming Large Language Models. In *ICLR*, 2024.

[25] Emaad Khwaja, Chris Lettieri, Gerald Woo, Eden Belouadah, Marc Cenac, Guillaume Jarry, Enguerrand Paquin, Xunyi Zhao, Viktoriya Zhukov, Othmane Abou-Amal, Chenghao Liu, Ameet Talwalkar, and David Asker. Toto 2.0: Time Series Forecasting Enters the Scaling Era. `https://arxiv.org/abs/2605.20119`, 2026.

[26] Jaehoon Lee, Jun Seo, Seunghan Lee, Tae Yoon Lim, Dongwan Kang, Hwanil Choi, Minjae Kim, Sungdong Yoo, Junhyeok Kang, Sangjun Han, et al. FinVerse: Financial Time-Series Benchmark. `https://arxiv.org/abs/2608.03259`, 2026.

[27] Seunghan Lee, Juri Hong, Kibok Lee, and Taeyoung Park. Sequential Order-Robust Mamba for Time Series Forecasting. In *NeurIPS 2024 TSALM Workshop*, 2024.

[28] Seunghan Lee, Jaehoon Lee, Jun Seo, Junhyeok Kang, Sangjun Han, Sungdong Yoo, Minjae Kim, Tae Yoon Lim, Dongwan Kang, Hwanil Choi, et al. Beyond Magnitude and Shape: A Direction-Aware Loss for Time Series Forecasting. `https://arxiv.org/abs/2608.01857`, 2026.

[29] Seunghan Lee, Jaehoon Lee, Jun Seo, Sungdong Yoo, Minjae Kim, Tae Yoon Lim, Dongwan Kang, Hwanil Choi, Soonyoung Lee, and Wonbin Ahn. Not All Retrievals are Useful: Cross-Attention for Input-Aware RAG in Time Series Forecasting. In *KDD 2026 Workshop on Mining and Learning from Time Series*, 2026.

[30] Seunghan Lee, Taeyoung Park, and Kibok Lee. Learning to Embed Time Series Patches Independently. In *ICLR*, 2024.

[31] Seunghan Lee, Taeyoung Park, and Kibok Lee. Dataset-Driven Channel Masks in Transformers for Multivariate Time Series. In *ICASSP*, pages 2401–2405, 2026.

[32] Seunghan Lee, Jun Seo, Jaehoon Lee, Sungdong Yoo, Minjae Kim, Tae Yoon Lim, Dongwan Kang, Hwanil Choi, Soonyoung Lee, and Wonbin Ahn. FinSTaR: Towards Financial Reasoning with Time Series Reasoning Models. In *EMNLP Industry Track*, 2026.

[33] Seunghan Lee, Jun Seo, Jaehoon Lee, Sungdong Yoo, Minjae Kim, Tae Yoon Lim, Dongwan Kang, Hwanil Choi, Soonyoung Lee, and Wonbin Ahn. Rethinking Multimodal Fusion for Time Series: Auxiliary Modalities Need Constrained Fusion. In *KDD 2026 Workshop on Mining and Learning from Time Series*, 2026.

[34] Haoxin Liu, Shangqing Xu, Zhiyuan Zhao, Lingkai Kong, Harshavardhan Kamarthi, Aditya B. Sasanur, Megha Sharma, Jiaming Cui, Qingsong Wen, Chao Zhang, and B. Aditya Prakash. Time-MMD: Multi-Domain Multimodal Dataset for Time Series Analysis. In *NeurIPS Datasets and Benchmarks Track*, 2024.

[35] Xu Liu, Juncheng Liu, Gerald Woo, Taha Aksu, Yuxuan Liang, Roger Zimmermann, Chenghao Liu, Silvio Savarese, Caiming Xiong, and Doyen Sahoo. Moirai-MoE: Empowering Time Series Foundation Models with Sparse Mixture of Experts. In *ICML*, 2025.

[36] Yong Liu, Guo Qin, Zhiyuan Shi, Zhi Chen, Caiyin Yang, Xiangdong Huang, Jianmin Wang, and Mingsheng Long. Sundial: A Family of Highly Capable Time Series Foundation Models. In *ICML*, 2025.

[37] Yong Liu, Haoran Zhang, Chenyu Li, Xiangdong Huang, Jianmin Wang, and Mingsheng Long. Timer: Generative Pre-trained Transformers Are Large Time Series Models. In *ICML*, 2024.

[38] Donghao Luo and Xue Wang. ModernTCN: A Modern Pure Convolution Structure for General Time Series Analysis. In *ICLR*, 2024.

[39] Vladyslav Moroshan, Julien Siems, Arber Zela, Timur Carstensen, and Frank Hutter. TempoPFN: Synthetic Pre-training of Linear RNNs for Zero-shot Time Series Forecasting. `https://arxiv.org/abs/2510.25502`, 2025.

[40] Yuqi Nie, Nam H. Nguyen, Phanwadee Sinthong, and Jayant Kalagnanam. A Time Series is Worth 64 Words: Long-term Forecasting with Transformers. In *ICLR*, 2023.

[41] Kashif Rasul, Arjun Ashok, Andrew Robert Williams, Hena Ghonia, Rishika Bhagwatkar, Arian Khorasani, Mohammad Javad Darvishi Bayazi, George Adamopoulos, Roland Riachi, Nadhir Hassen, Marin Biloš, Sahil Garg, Anderson Schneider, Nicolas Chapados, Alexandre Drouin, Valentina Zantedeschi, Yuriy Nevmyvaka, and Irina Rish. Lag-Llama: Towards Foundation Models for Probabilistic Time Series Forecasting. In *NeurIPS 2023 R0-FoMo Workshop*, 2023.

[42] David Salinas, Valentin Flunkert, Jan Gasthaus, and Tim Januschowski. DeepAR: Probabilistic Forecasting with Autoregressive Recurrent Networks. *International Journal of Forecasting*, 36(3):1181–1191, 2020.

[43] Omer Berat Sezer, Mehmet Ugur Gudelek, and Ahmet Murat Ozbayoglu. Financial Time Series Forecasting with Deep Learning: A Systematic Literature Review: 2005–2019. *Applied Soft Computing*, 90:106181, 2020.

[44] Xiaoming Shi, Shiyu Wang, Yuqi Nie, Dianqi Li, Zhou Ye, Qingsong Wen, and Ming Jin. Time-MoE: Billion-Scale Time Series Foundation Models with Mixture of Experts. In *ICLR*, 2025.

[45] Xue Wang, Tian Zhou, Jinyang Gao, Bolin Ding, and Jingren Zhou. Output Scaling: YingLong-Delayed Chain of Thought in a Large Pretrained Time Series Forecasting Model. `https://arxiv.org/abs/2506.11029`, 2025.

[46] Yihang Wang, Yuying Qiu, Peng Chen, Kai Zhao, Yang Shu, Zhongwen Rao, Lujia Pan, Bin Yang, and Chenjuan Guo. Towards a General Time Series Forecasting Model with Unified Representation and Adaptive Transfer. In *ICML*, 2025.

[47] Zihan Wang, Fanheng Kong, Shi Feng, Ming Wang, Xiaocui Yang, Han Zhao, Daling Wang, and Yifei Zhang. Is Mamba Effective for Time Series Forecasting? *Neurocomputing*, 619:129178, 2025.

[48] Gerald Woo, Chenghao Liu, Akshat Kumar, Caiming Xiong, Silvio Savarese, and Doyen Sahoo. Unified Training of Universal Time Series Forecasting Transformers. In *ICML*, 2024.

[49] Shifeng Xie, Vasilii Feofanov, Ambroise Odonnat, Lei Zan, Marius Alonso, Jianfeng Zhang, Themis Palpanas, Lujia Pan, Keli Zhang, and Ievgen Redko. CauKer: Classification Time Series Foundation Models Can Be Pretrained on Synthetic Data. In *ICLR*, 2026.

[50] Ailing Zeng, Muxi Chen, Lei Zhang, and Qiang Xu. Are Transformers Effective for Time Series Forecasting? In *AAAI*, 2023.

[51] George Zerveas, Srideepika Jayaraman, Dhaval Patel, Anuradha Bhamidipaty, and Carsten Eickhoff. A Transformer-based Framework for Multivariate Time Series Representation Learning. In *KDD*, pages 2114–2124, 2021.

[52] Haoyi Zhou, Shanghang Zhang, Jieqi Peng, Shuai Zhang, Jianxin Li, Hui Xiong, and Wancai Zhang. Informer: Beyond Efficient Transformer for Long Sequence Time-Series Forecasting. In *AAAI*, volume 35, pages 11106–11115, 2021.

[53] Zhuohang Zhu, Haodong Chen, Qiang Qu, and Vera Chung. FinCast: A Foundation Model for Financial Time-Series Forecasting. `https://arxiv.org/abs/2508.19609`, 2025.