AME-TS: Anchored Mixture-of-Experts for Time Series Forecasting

Rui Wang* **Renhao Xue** **Ray Razi** **Huan Song** **Hannah R. Marlowe**
Amazon Web Services

Abstract

Time series forecasting models are increasingly scaled through large Transformer backbones, yet most existing approaches process all series through a shared dense computation path despite substantial heterogeneity in temporal structure. Mixture-of-Experts (MoE) offers a natural alternative by enabling conditional computation, but standard MoE routing leaves expert specialization weakly identified and often unstable during downstream adaptation. We propose AME-TS, a structure-guided sparse time series foundation model that aligns expert routing with interpretable temporal structure. AME-TS first uses a lightweight regime predictor to estimate series-level descriptors, including forecastability, seasonality, trend, and sparsity, and maps them to a soft structural prior over experts. This series-level prior guides token-level routing during training, encouraging structure-aligned specialization. On the GIFT-Eval benchmark, AME-TS delivers a strong accuracy–efficiency tradeoff across model scales: it substantially outperforms existing time series foundation models at small model scales and remains competitive with the strongest models at larger scales, while activating substantially fewer parameters through sparse routing. We further show that AME-TS learns more interpretable routing geometry and substantially more stable expert specialization than standard MoE during fine-tuning on the M5 dataset. These results suggest that structure-aware routing is an effective and reliable way to realize the benefits of sparse expert models for time series forecasting.

1 Introduction

Time series forecasting is a fundamental problem in applications such as retail demand planning [5, 32], cloud operations[12, 27], healthcare[25], and weather prediction [17, 31]. Time series foundation models have achieved strong zero-shot and transfer performance by scaling Transformer-based architectures over large and diverse collections of time series [2, 7, 10, 3, 30]. Most of these models, however, process all series through a shared dense computation path. This is an inefficient use of model capacity, because real-world time series differ substantially in the temporal structure relevant for forecasting. A highly seasonal retail series, a strongly trending macroeconomic indicator, and a sparse cloud monitoring signal need not be processed in the same way. These structural differences directly influence which representations and computations are most useful for accurate forecasting.

Mixture-of-Experts (MoE) architectures [6] scale neural networks by conditionally activating only a subset of parameters for each input, and have been adopted successfully in both large language models [9, 16, 11, 28, 29] and, more recently, time series forecasting models [19, 30]. In principle, MoE is a natural fit for forecasting because different time series may benefit from different computational pathways. In practice, however, standard MoE routing leaves expert specialization weakly identified: because experts are permutation-symmetric and therefore do not begin with fixed semantic roles,

*Corresponding author: ruw020@ucsd.edu

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

the router may organize them according to random initialization, optimization dynamics, or fine-tuning dynamics rather than forecasting-relevant temporal structure. As a result, expert specialization can be fragile, difficult to interpret, and prone to drift during downstream adaptation. More broadly, recent studies across language, vision, diffusion, and multimodal learning suggest that stronger expert specialization and explicitly guided routing can improve MoE performance and interpretability, further motivating structure-aware routing mechanisms [14, 15, 34, 24, 29].

These observations suggest that MoE routing should be guided by meaningful problem structure, but such guidance must remain flexible in the foundation-model setting. Hard-coding expert roles or computation paths could limit generalization across heterogeneous domains. We therefore use temporal structure only as a soft training signal: it biases expert specialization toward interpretable regimes, while leaving the learned router free to adapt during broad pretraining, inference, and downstream transfer.

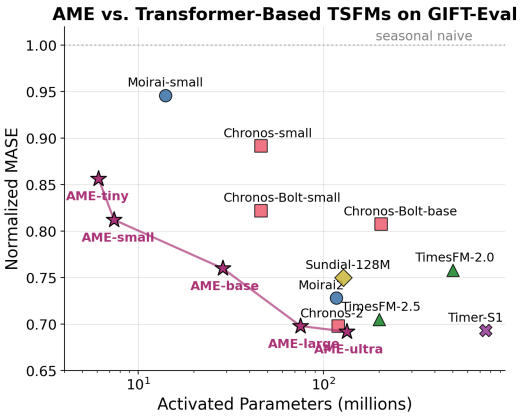


Figure 1: **MASE vs. activated parameter count on GIFT-Eval**. Each point shows a foundation model or an AME variant, with lower normalized MASE indicating better forecasting performance. AME-TS achieves a favorable accuracy–efficiency tradeoff across scales, matching or outperforming strong TSFMs while activating substantially fewer parameters through sparse routing.

Time series forecasting is a particularly natural setting for structure-aware routing because it provides interpretable axes along which expert specialization can be organized. Properties such as forecastability [32, 13], seasonality, trend, and sparsity capture distinct aspects of temporal structure that are strongly tied to forecasting performance. Motivated by this view, we propose Anchored Mixture-of-Experts (AME-TS), a structure-guided sparse time series foundation model that uses interpretable temporal descriptors to guide MoE routing. It first uses a lightweight regime predictor to estimate interpretable temporal descriptors of each input series, including forecastability, seasonality, trend, and sparsity. These regime scores are then mapped to a series-level soft prior over experts, which is used during training through a prior-alignment loss to guide token-level routing. This encourages structure-aligned specialization while preserving the flexibility of learned routing. As a result, AME-TS breaks the permutation symmetry of standard MoE, leading to more stable, interpretable, and selective expert specialization.

Our contributions are summarized as follows.

- We propose AME-TS, a time series foundation model with structure-aware MoE routing. It constructs a soft prior over expert usage from interpretable temporal descriptors, routes weakly structured series to shared experts through an entropy-based gate, and introduces a training-only prior-alignment loss to align token-level MoE routing with series-level temporal structure
- On the GIFT-Eval benchmark [1], we show that AME-TS achieves a strong accuracy–efficiency tradeoff, as summarized in Figure 1: it substantially outperforms existing small-scale time series foundation models and remains competitive with the strongest larger baselines, while using substantially fewer active parameters through sparse routing.
- We show that the routing learned by AME-TS is interpretable, with both routing space and representation space organizing around meaningful temporal regimes.
- We demonstrate that anchored routing improves fine-tuning stability: on the M5 dataset [22], AME-TS achieves strong zero-shot and fine-tuned performance while maintaining substantially more stable expert specialization than standard MoE during adaptation.

2 Related Work

Mixture-of-Experts for Scalable Sequence Modeling MoE architectures have become a central approach to scaling neural networks by conditionally activating only a subset of parameters for each

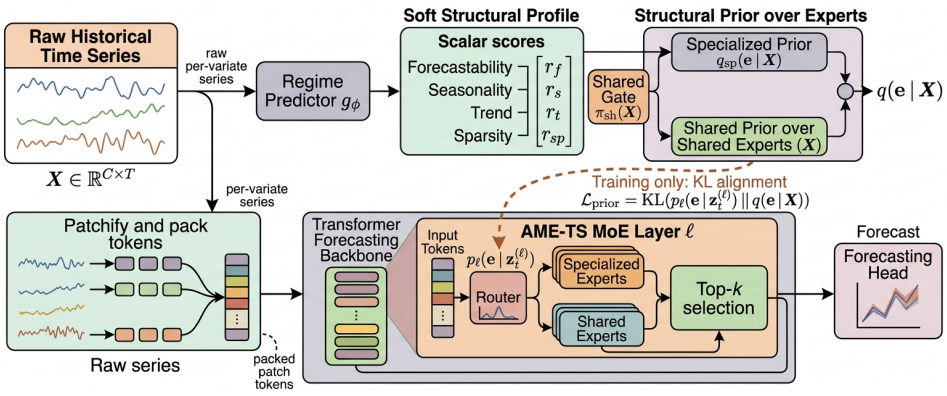


Figure 2: Overview of AME-TS. A regime predictor extracts a soft structural profile from raw time series to construct a structural prior over experts $q(e \mid X)$, while patchified tokens are processed by a Transformer forecasting backbone with AME-TS MoE layers. Training uses KL alignment between token-level routing and the structural prior; inference uses only the learned router.

input. Early sparse models such as Switch Transformers [11] and later large-scale systems such as Mixtral [16] and DeepSeekMoE [9] show that sparse routing can substantially increase model capacity without proportionally increasing computation. Other variants improve expert utilization through more expressive routing mechanisms or modified expert structures [36, 37]. Although these methods differ in routing design and expert architecture, their primary focus is on scaling efficiency and expert utilization. In most cases, however, expert identities remain emergent, with specialization shaped largely by training dynamics rather than explicit structural guidance. Our work is motivated by the observation that, for time series forecasting, sparse capacity alone is not enough, routing should also align with forecasting-relevant temporal structure.

Guided and Structure-Aware Routing in Mixture-of-Experts A growing body of work suggests that MoE performance depends not only on the presence of experts, but also on how expert specialization is induced [14]. Switch-NeRF [23] shows that spatially grounded routing improves scene decomposition in neural radiance fields. Routing Matters in MoE [34] shows that explicit routing guidance improves expert specialization in diffusion transformers. Guiding the Experts [24] uses semantic priors to encourage focused routing in Soft MoE vision models, while [15] uses multimodal temporal structure to guide expert allocation. In multimodal large language models, MoME [28] and MoVA [38] further show that routing aligned with task or modality structure can improve adaptation and generalization. Together, these works suggest that MoE routing benefits from alignment with meaningful problem structure rather than being learned entirely without guidance. More broadly, they support the view that structured routing can improve not only downstream performance, but also the interpretability and stability of expert specialization. Our work follows this line of research but differs in two ways. These methods rely on semantic, spatial, or modality priors in vision, diffusion, and multimodal models, whereas AME-TS derives its prior from interpretable temporal descriptors and, to our knowledge, is the first to bring structure-aware routing to time series foundation models. Moreover, the prior is used only as a training signal and never enters the router logits at inference.

Time Series Foundation Models Recent progress in time series forecasting has been driven by large pretrained forecasting models. Foundation models such as Chronos [3, 2], TimesFM [10], Moirai [35], and the recent xLSTM-based TiRex [4] show that scaling training data and model capacity can yield strong zero-shot and transfer performance across diverse forecasting tasks. More recent work, such as Moirai-MoE [19] and TimeMoE [30], has also begun to explore sparse expert architectures for forecasting. These works demonstrate the promise of MoE for forecasting, but they do not explicitly use interpretable temporal information to guide expert specialization. In contrast, our approach uses series-level structural descriptors to construct a soft prior over expert assignments.

3 Methodology

3.1 AME-TS Overview

AME-TS is a structure-guided sparse MoE forecasting model that aligns expert specialization with interpretable temporal structure. As illustrated in Figure 2, AME-TS balances a series-level structural prior with token-level learned routing. The series-level prior summarizes global temporal properties of the input series, while the token-level router selects experts from local latent representations inside the forecasting backbone. Given an input series, a lightweight regime predictor estimates a soft structural profile consisting of forecastability, seasonality strength, trend strength, and sparsity. This profile is mapped to a soft prior over experts, biasing expert specialization toward interpretable temporal regimes. During training, a prior-alignment loss encourages token-level routing to align with the series-level prior while still adapting to fine-grained token-level patterns. During inference, the prior is not explicitly injected and routing is determined only by the learned router.

3.2 Structural Profile and Expert Prior

Structural descriptors. A central idea of AME-TS is that expert routing should reflect the temporal structure of the time series. We therefore characterize each series using four complementary structural descriptors: forecastability, seasonality strength, trend strength, and sparsity. Together, these descriptors capture whether a series is spectrally regular, periodically structured, directionally changing, or intermittent, providing a compact and interpretable profile for guiding expert specialization.

Forecastability measures how concentrated a series is in the frequency domain, using the normalized entropy of its power spectral density [32, 13]. High forecastability indicates more regular spectral structure, while low forecastability corresponds to more noise-like dynamics. Seasonality strength measures the fraction of non-trend variation explained by the seasonal component in an STL decomposition [8]. Trend strength measures the magnitude of normalized linear change across the input window. Sparsity measures intermittency or repeated-value behavior using the fraction of non-unique values in the window. These descriptors are intentionally simple and are not intended to fully characterize a time series; rather, they provide a small, inexpensive set of interpretable structural signals that bias expert routing toward forecasting-relevant regimes. Because the prior is only a soft training signal, richer nonstationary, multi-scale, or local behavior is left to the learned token-level router and the backbone, so the descriptors need only be informative enough to seed specialization.

Detailed mathematical definitions are provided in Appendix A.2.

These four descriptors are complementary rather than redundant. Empirically, their pairwise correlations on the pre-training pool are all moderate, as shown in Table 5, indicating that they capture distinct structural axes rather than collapsing to a single structural factor. Our ablation study in Section 4.2 further shows that each descriptor contributes to forecasting performance.

Regime predictor. Computing these descriptors analytically for every training sample would be prohibitively expensive. We therefore compute them on a small subset of series from the pretraining pool and train a lightweight regime predictor g_ϕ to provide fast estimates during model training:

$$g_\phi(X) = [r_{\mathrm{f}}, r_{\mathrm{s}}, r_{\mathrm{t}}, r_{\mathrm{sp}}] \in [0,1]^4,$$

where $X = (x_1, \ldots, x_T)$ denotes a univariate input window of length T (for multivariate inputs, each variate is treated as its own univariate series and g_ϕ is applied to that variate's own context window), and r_{f}, r_{s}, r_{t}, and r_{sp} denote the predicted scores for forecastability, seasonality strength, trend strength, and sparsity, respectively. The regime predictor is trained separately and kept frozen during AME-TS training. Architectural and training details are provided in Appendix A.3.

Specialized and shared experts. Using the regime predictor, we construct a prior over experts that biases routing toward forecasting-relevant temporal structure. We partition the expert set into two groups: *specialized experts* $\mathcal{E}_{\mathrm{sp}}$, which are associated with the four structural descriptors, and *shared experts* $\mathcal{E}_{\mathrm{sh}}$, which provide fallback capacity when structural profile is weak, mixed, or ambiguous.

We define a fixed anchor distribution $q_{\mathrm{anchor}}(e \mid d)$ over specialized experts by assigning each descriptor d to a subset of specialized experts. In practice, $q_{\mathrm{anchor}}(e \mid d)$ is uniform over the experts assigned to descriptor d and zero elsewhere. When the number of specialized experts exceeds

the number of descriptors, experts are distributed as evenly as possible across descriptors. The regime-induced prior over specialized experts is defined as

$$q_{\mathrm{sp}}(e \mid X) \propto \sum_{d=1}^{D} q_{\mathrm{anchor}}(e \mid d)\, g_\phi(d \mid X), \qquad e \in \mathcal{E}_{\mathrm{sp}}.$$

Intuitively, higher regime scores place more prior mass on the experts anchored to those descriptors. Because the structural profile is soft, the resulting prior can simultaneously favor multiple expert groups when a series exhibits mixed structure.

Not all series admit strong descriptor-specific specialization. Some exhibit weak structural signals, while others lie between multiple regimes. To handle such cases, we allocate part of the prior mass to a shared expert pool through a shared gate $\pi_{\mathrm{sh}}(X) \in [0, 1]$, which increases when the regime profile is weak or uncertain. Let

$$H(X) = \frac{1}{D} \sum_{d=1}^{D} h(g_\phi(d \mid X)), \qquad S(X) = \max_{d \in \{1,\dots,D\}} g_\phi(d \mid X),$$

where h is the binary entropy. We define the shared gate as

$$\pi_{\mathrm{sh}}(X) = \big(1 - S(X)\big)\, \sigma(\alpha H(X) - b),$$

where $\sigma(\cdot)$ is the sigmoid function, and α and b are learnable or fixed parameters. This form reflects two intuitions: shared experts should receive more mass when the structural profile is uncertain, and when no descriptor is strongly activated. The factor $1 - S(X)$ suppresses shared mass when at least one descriptor is strongly activated, while the entropy term increases shared mass when the predicted descriptor scores are diffuse.

The final prior over all experts is given by

$$q(e \mid X) = \begin{cases} \dfrac{\pi_{\mathrm{sh}}(X)}{|\mathcal{E}_{\mathrm{sh}}|}, & e \in \mathcal{E}_{\mathrm{sh}}, \\ \big(1 - \pi_{\mathrm{sh}}(X)\big)\, q_{\mathrm{sp}}(e \mid X), & e \in \mathcal{E}_{\mathrm{sp}}. \end{cases}$$

This prior is soft and interpretable: it breaks the permutation symmetry of standard MoE routing by anchoring expert preferences to meaningful regimes, while allowing uncertain cases to be absorbed by shared experts. Rather than hard-assigning experts, it provides a series-level structural bias that is later used to guide token-level routing.

3.3 Training with Prior Alignment

The key training mechanism of AME-TS is a prior-alignment loss that transfers series-level temporal structure into token-level sparse routing. The structural prior $q(e \mid X)$ defined in Section 3.2 is constructed at the series level, whereas routing in the MoE layers operates at the token level. Given a token representation $z_t^{(\ell)}$ at layer ℓ, where t indexes patch tokens, the router produces logits

$$s_\ell(z_t^{(\ell)}) = W_r^{(\ell)} z_t^{(\ell)},$$

which define a token-level routing distribution

$$p_\ell(e \mid z_t^{(\ell)}) = \mathrm{softmax}\big(s_\ell(z_t^{(\ell)})\big).$$

As in standard sparse MoE, only the top-k experts are activated per token. Rather than injecting the structural prior directly into the router logits, we use it only during training as a soft signal that encourages alignment between token-level routing and series-level structure. Specifically, we introduce a Kullback–Leibler (KL) regularization term:

$$\mathcal{L}_{\mathrm{prior}} = \frac{1}{N_L} \sum_{\ell=0}^{N_L - 1} \lambda_\ell\, \mathbb{E}_t \Big[\mathrm{KL}\big(p_\ell(e \mid z_t^{(\ell)}) \,\|\, q(e \mid X)\big) \Big],$$

where N_L is the number of Transformer layers. We use the forward KL divergence $\mathrm{KL}(p \,\|\, q)$, which encourages the token-level router to remain consistent with the soft structural prior. This prior-alignment loss is the main mechanism that transfers interpretable series-level structure into token-level sparse routing. It encourages experts to develop stable structural roles, while still allowing the learned router to adapt to fine-grained token-level patterns.

Layer-wise prior weighting. To control where specialization emerges in the network, we vary the strength of prior regularization across layers. We adopt a linearly increasing schedule:

$$\lambda_\ell = \lambda_{\max} \frac{\ell}{N_L - 1},$$

where $\ell \in \{0, \ldots, N_L - 1\}$ is the layer index. This design applies minimal regime pressure to early layers, allowing them to learn shared representations, while encouraging deeper layers to specialize according to the regime prior. This reflects the intuition that high-level structural information is more useful in later stages of computation.

Orthogonality loss. When multiple experts are associated with the same descriptor, we further include an orthogonality loss to promote diversity among their outputs:

$$\mathcal{L}_{\text{ortho}} = \mathbb{E}_{i \neq j} \left[|\langle h_i, h_j \rangle| \right],$$

where h_i and h_j are outputs of co-activated experts within the same group. In practice, we find that the prior-alignment term substantially mitigates expert collapse and stabilizes routing, while the orthogonality loss provides additional but smaller gains, especially when using larger expert pools.

3.4 Forecasting Backbone and Prediction Loss

AME-TS is built on an encoder-only Transformer forecasting backbone. Given an input window, each variate is treated as a univariate series, partitioned into non-overlapping patches, and projected into latent token representations. For multivariate inputs, tokens from all variates are packed into a single sequence with variate identity embeddings and processed by a stack of Transformer encoder layers. Forecasting is formulated as masked prediction over the forecast horizon: historical tokens are provided as observed input, while tokens in the future horizon are masked. The encoder outputs at these masked positions are then used to predict the target values. We replace the dense feed-forward layers in each encoder block with MoE layers, while keeping the rest of the backbone unchanged. This backbone design follows the packed-sequence formulation of [35].

The primary forecasting objective $\mathcal{L}_{\text{task}}$ is a masked prediction loss over the forecast horizon. Given predicted values $\hat{y}$ and ground-truth targets y, we use an ℓ_1 loss:

$$\mathcal{L}_{\text{task}} = \mathbb{E} \left[\|\hat{y} - y\|_1 \right].$$

The training objective combines the prediction loss with the prior-alignment and orthogonality losses:

$$\mathcal{L} = \mathcal{L}_{\text{task}} + \lambda_{\text{prior}} \mathcal{L}_{\text{prior}} + \lambda_{\text{ortho}} \mathcal{L}_{\text{ortho}}.$$

4 Experiment

4.1 Experimental Setup

We pre-train AME-TS on a pre-training pool spanning 96 dataset configurations across 8 domains (Table 3), comprising approximately 3.5 million individual time series and 18 billion observations, with frequencies ranging from seconds to yearly. A detailed breakdown is provided in Appendix A.4. In addition to the foundation-model setting, we also evaluate AME-TS in a dataset-specific setting, where a smaller model is trained independently for each GIFT-Eval task.

We train using AdamW with a linear decay learning-rate schedule and linear warmup. Training is performed on a single p4 instance with 8 A100 GPUs. For each task, we sweep over eight context lengths, select the best setting on the validation set, and use that context length for the final test-set evaluation. Following the official protocol [1], we report four metrics, MASE, sMAPE, MAE, and RMSE, computed per series and then aggregated across the 97 tasks after normalization by the Seasonal Naive baseline. Detailed architecture configurations for all AME-TS variants are provided in Appendix Table 4, and additional training details are provided in Appendix A.5.

4.2 GIFT-Eval Results and Ablation

Main results We evaluate AME-TS on the GIFT-Eval benchmark, which comprises 97 forecasting tasks spanning diverse datasets, frequencies, and prediction horizons. We compare against published

Table 1: **Summary GIFT-Eval results.** Scores are geometric means over 97 tasks normalized by the Seasonal Naive baseline; lower is better. Full results across all four metrics are provided in Appendix Table 6. † Active parameters per token due to sparse MoE routing.

Model	#Params	MASE ↓	RMSE ↓
Moirai-Small [35]	14M	0.946	1.056
Chronos-Small [3]	46M	0.892	0.866
Chronos-2 [2]	120M	0.698	0.704
Sundial-128M [21]	128M	0.750	0.739
TimesFM-2.5 [10]	231M	0.705	0.694
Moirai2 [18]	300M	0.728	0.739
AME-TS Tiny	14M†6M	0.856	0.808
AME-TS Small	30M†7M	0.812	0.767
AME-TS Base	62M†29M	0.765	0.737
AME-TS Large	162M†75M	0.700	0.690
AME-TS Ultra	540M†133M	**0.692**	**0.687**

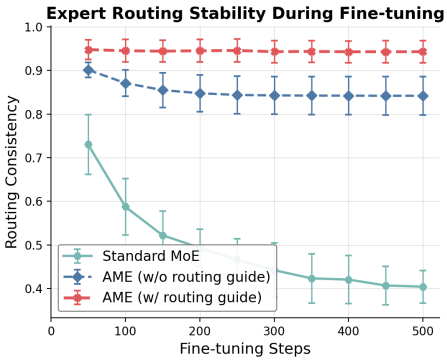


Figure 3: **Routing stability during fine-tuning on M5.** AME-TS maintains substantially more stable expert specialization than standard MoE, and routing guidance further improves stability during adaptation.

time series foundation models at three parameter scales, as well as task-specific per-dataset models. For each scale group, we report the top published models at that scale from the official leaderboard. Although Moirai-MoE [19] and TimeMoE [30] are relevant forecasting MoE models in the literature, they do not appear as directly comparable named entries on the public GIFT-Eval leaderboard at the time of writing, so our main comparison follows the top publicly listed leaderboard baselines.

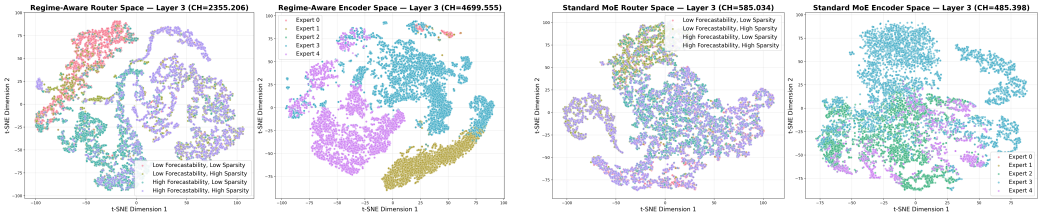
Figure 1 and Table 1 summarize the main GIFT-Eval comparison. Figure 1 highlights the accuracy–efficiency tradeoff in terms of MASE versus activated parameter count, while Table 1 reports the corresponding MASE and RMSE values. AME-TS achieves the strongest MASE and RMSE among the listed models, with AME-TS Ultra reaching 0.692 MASE and 0.687 RMSE while activating 133M parameters per token. The gains are especially pronounced at smaller scales: AME-TS Small outperforms both Moirai-Small and Chronos-Small while activating only 7M parameters per token. Returns diminish at the largest scale: from Large to Ultra, MASE improves only from 0.700 to 0.692 despite a roughly 3.3× increase in total parameters (162M to 540M), so we did not scale beyond Ultra. Full results across all four metrics, including sMAPE and MAE, as well as task-specific per-dataset model comparisons, are provided in Appendix Table 6.

Ablation study Beyond this main comparison, we conduct controlled ablations to isolate the contribution of routing design, as shown in Table 7. Replacing a dense model with standard MoE yields modest improvements, confirming that sparse expert capacity alone is beneficial. Introducing regime-aware routing yields further gains, showing that the improvement comes not only from increased capacity, but from better alignment between routing decisions and temporal structure. Among prior-integration strategies, KL-guided routing consistently outperforms additive prior injection, indicating that soft alignment is more effective than direct prior injection. Forward and reverse KL achieve similar performance, with forward KL showing slightly more consistent gains across tasks.

We further study the contribution of individual regime features by dropping one feature at a time. Removing any single feature degrades performance, confirming that the four regime descriptors provide complementary routing signals. The largest drops are observed when forecastability or sparsity is removed, suggesting that these features provide especially informative cues for expert allocation. Taken together, these ablations show that the gains of AME-TS arise from structure-aligned routing rather than sparse capacity alone.

4.3 Routing Interpretability and Representation Analysis

We evaluate whether AME-TS learns routing patterns and internal representations that align with interpretable temporal structure. To this end, we compare AME-TS against a standard MoE model with the same backbone and expert architecture but without regime-aware routing. From the same layer in both models, we extract router logits as routing embeddings, encoder representations, top-1 expert assignments, and regime predictions from the regime predictor.



(a) Regime-Aware (AME) (b) Standard MoE

Figure 4: t-SNE visualizations comparing AME-TS and standard MoE at the same layer. Each subfigure contains two panels: router space on the left and encoder space on the right. In the router space, points are colored by regime profiles derived from forecastability and sparsity labels. In the encoder space, points are colored by top-1 expert assignment. AME-TS yields clearer regime-aligned structure in router space and substantially stronger expert separation in encoder space, as also reflected by the higher Calinski–Harabasz (CH) scores reported in the subplot titles.

Table 2: WRMSSE on the M5 dataset across all 12 hierarchical levels. Lower is better. "Rank 1" denotes the first-place M5 competition result, reported here as a task-specific reference.

Model	Tot	St	Str	Cat	Dept	St-Cat	St-Dept	Str-Cat	Str-Dept	Prod	Prod-St	Prod-Str	Average
M5 Rank 1 [22]	0.199	0.310	0.400	0.277	0.365	0.390	0.474	0.480	0.573	0.966	0.929	0.884	0.520
AME-TS (Zero-shot)	0.367	0.470	0.536	0.424	0.553	0.517	0.620	0.606	0.684	0.851	0.860	0.867	0.613
AME-TS (Fine-tuned)	0.226	0.357	0.430	0.318	0.441	0.419	0.521	0.502	0.579	0.754	0.781	0.765	**0.506**

To visualize the learned spaces, we project both routing embeddings and encoder representations into two dimensions using t-SNE. In the router space, points are colored by regime profiles obtained by thresholding the regime predictor outputs. In Figure 4, we focus on binary labels derived from forecastability and sparsity, which yields four regime groups. In encoder space, points are colored by top-1 expert assignment. This lets us separately examine whether routing geometry aligns with temporal structure and whether encoder representations organize according to expert specialization.

Figure 4 shows a clear qualitative difference between AME-TS and standard MoE. In the router space, AME-TS forms more coherent regions associated with different regime labels, whereas standard MoE exhibits substantially greater mixing across groups. In the encoder space, the contrast is even stronger: AME-TS produces sharply separated expert-specific regions, while standard MoE yields weaker and more entangled clusters. These patterns suggest that the regime prior not only affects routing decisions directly, but also reshapes the representation geometry learned by the backbone.

We quantify this effect using the Calinski–Harabasz (CH) index [33], which measures cluster compactness and separation. Higher CH values indicate better-defined clusters. As reported in the subplot titles of Figure 4, AME-TS achieves substantially higher CH scores than standard MoE in both router and encoder space. The gain is especially pronounced in the encoder space, where expert assignments under AME-TS form highly separable regions. This indicates that regime-aware routing leads to more structured routing behavior and stronger expert specialization.

Overall, these results suggest that the gains of AME-TS are not merely due to increased sparse capacity. Instead, the regime prior induces a more interpretable routing geometry, in which series with similar temporal structure are routed more consistently, and expert-specific computation becomes more clearly separated in representation space.

4.4 Zero-Shot and Fine-Tuning on M5

We evaluate AME-TS-Base on the M5 Walmart dataset [22], which is not included in pre-training. M5 contains 30,490 daily retail time series organized into 12 hierarchical levels and is evaluated using WRMSSE over a 28-day forecasting horizon. Because many recent forecasting foundation models include M5 in pre-training, we avoid direct zero-shot comparison to such models and instead report results against the first-place M5 competition result as a task-specific reference.

We first evaluate zero-shot forecasting without any task-specific adaptation. Even without fine-tuning, AME-TS outperforms the competition winner on all three item-level aggregations (Prod, Prod-St,

Prod-Str). This is especially notable because lower-level M5 series are more heterogeneous and often exhibit higher sparsity and less regular temporal structure. These results suggest that regime-aware routing transfers effectively to an unseen retail dataset with diverse structural characteristics.

We then fine-tune AME-TS on M5 training data for 20K steps with a batch size of 16, incorporating a day-of-week calendar covariate as a second input variate. Because the model treats each variate as a separate token sequence sharing the same time index, the day-of-week signal is fully observed at both context and forecast positions. Fine-tuned AME-TS achieves an average WRMSSE of 0.506, improving over the first-place M5 result (0.520), which relied on ensembles of gradient-boosted models with extensive hand-crafted features and hierarchical reconciliation. The gains are strongest at the item level, where AME-TS reduces WRMSSE by 13–22% relative to Rank 1. Fine-tuned AME-TS still falls short of Rank 1 on some higher aggregation levels, where strong seasonality and hierarchical reconciliation favor heavily engineered task-specific pipelines. Nevertheless, the item-level improvements are large enough to yield a lower average WRMSSE overall.

4.5 Routing Stability During Fine-Tuning

We evaluate whether expert specialization remains stable during fine-tuning on the M5 dataset. We compare AME-TS against two baselines with identical architectures: standard MoE (no prior) and an ablated variant of AME-TS in which routing guidance is removed during fine-tuning.

We define routing consistency as the agreement between current routing decisions and those at the initial checkpoint. For a fixed probe set $\mathcal{D}_{\text{probe}}$, let $e^{(0)}_{i,t,\ell}$ and $e^{(k)}_{i,t,\ell}$ denote the top-1 expert assigned to token t of series i at layer ℓ at the initial and k-th fine-tuning step, respectively. We compute

$$\mathrm{RC}(k) = \frac{1}{|\mathcal{S}|} \sum_{(i,t,\ell)\in\mathcal{S}} \mathbf{1}\left[e^{(k)}_{i,t,\ell} = e^{(0)}_{i,t,\ell}\right],$$

where $\mathcal{S}$ denotes the set of tracked series-token-layer tuples derived from $\mathcal{D}_{\text{probe}}$. Higher values indicate more stable expert specialization. We construct a fixed probe set of 1,000 time series sampled across all hierarchy levels of M5 to capture diverse temporal behaviors.

Figure 3 shows routing consistency over fine-tuning steps. Confidence intervals are computed over five independent fine-tuning runs with learning rates sampled from $[1 \times 10^{-5}, 5 \times 10^{-5}]$. AME-TS with regime-aware routing guidance maintains consistently high stability throughout training, indicating that expert roles are preserved during fine-tuning under the guidance of the regime predictor.

Notably, the ablated variant without routing guidance also remains stable, with only a modest decrease from $\sim$0.90 to $\sim$0.84. This suggests that AME-TS learns meaningful and well-structured expert specialization during pre-training, which is largely preserved even without explicit guidance during fine-tuning. In contrast, standard MoE exhibits substantial drift, with consistency dropping from $\sim$0.73 to $\sim$0.40, indicating that expert assignments are continuously reconfigured during fine-tuning.

These results show that AME-TS substantially mitigates routing instability in MoE during fine-tuning. By learning regime-aligned routing, AME-TS produces expert specialization that remains coherent during adaptation, avoiding the collapse and drift observed in standard MoE.

5 Discussion

AME-TS shows that aligning expert routing with temporal structure can improve both forecasting efficiency and adaptation stability in Mixture-of-Experts for time series forecasting. On GIFT-Eval, it delivers a strong accuracy–efficiency tradeoff across model scales, with especially large gains at small model sizes and competitive performance at larger scales while activating substantially fewer parameters. On M5, it shows strong transfer in both zero-shot and fine-tuned settings, with the largest gains appearing at lower hierarchical levels where the series are more diverse, sparse, and weakly structured. Beyond predictive accuracy, AME-TS learns structured expert specialization that remains substantially more stable than standard MoE during fine-tuning. Together, these results suggest that domain-informed routing priors can make sparse forecasting models both more reliable and more interpretable. Importantly, this does not require hard-coding expert assignments or forecasts. Temporal descriptors provide a structural bias for organizing sparse capacity during training, while the learned router retains the flexibility needed for broad pretraining and downstream adaptation.

A limitation of the current work is that AME-TS focuses on historical time series inputs, while future dynamics may also depend on external context such as text, events, or metadata. A natural next direction is multimodal forecasting, where structure-aware routing could combine temporal descriptors with such external context to further improve forecasting performance and interpretability.

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. *arXiv preprint arXiv:2410.10393*, 2024.

[2] Abdul Fatir Ansari, Oleksandr Shchur, Jaris Küken, Andreas Auer, Boran Han, Pedro Mercado, Syama Sundar Rangapuram, Huibin Shen, Lorenzo Stella, Xiyuan Zhang, et al. Chronos-2: From univariate to universal forecasting. *arXiv preprint arXiv: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, et al. Chronos: Learning the language of time series. *arXiv preprint arXiv:2403.07815*, 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. *arXiv preprint arXiv:2505.23719*, 2025.

[5] Konstantinos Benidis, Syama Sundar Rangapuram, Valentin Flunkert, Yuyang Wang, Danielle Maddix, Caner Turkmen, Jan Gasthaus, Michael Bohlke-Schneider, David Salinas, Lorenzo Stella, et al. Deep learning for time series forecasting: Tutorial and literature survey. *ACM Computing Surveys*, 55(6):1–36, 2022.

[6] Weilin Cai, Juyong Jiang, Fan Wang, Jing Tang, Sunghun Kim, and Jiayi Huang. A survey on mixture of experts in large language models. *IEEE Transactions on Knowledge and Data Engineering*, 2025.

[7] Defu Cao, Michael Gee, Jinbo Liu, Hengxuan Wang, Wei Yang, Rui Wang, and Yan Liu. Conversational time series foundation models: Towards explainable and effective forecasting. *arXiv preprint arXiv:2512.16022*, 2025.

[8] Robert B Cleveland, William S Cleveland, Jean E McRae, Irma Terpenning, et al. Stl: A seasonal-trend decomposition. *J. off. Stat*, 6(1):3–73, 1990.

[9] Damai Dai, Chengqi Deng, Chenggang Zhao, RX Xu, Huazuo Gao, Deli Chen, Jiashi Li, Wangding Zeng, Xingkai Yu, Yu Wu, et al. Deepseekmoe: Towards ultimate expert specialization in mixture-of-experts language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1280–1297, 2024.

[10] Abhimanyu Das, Weihao Kong, Rajat Sen, and Yichen Zhou. A decoder-only foundation model for time-series forecasting. In *Forty-first international conference on machine learning*, 2024.

[11] William Fedus, Barret Zoph, and Noam Shazeer. Switch transformers: Scaling to trillion parameter models with simple and efficient sparsity. *Journal of Machine Learning Research*, 23(120):1–39, 2022.

[12] Rakshitha Godahewa, Christoph Bergmeir, Geoffrey I Webb, Rob J Hyndman, and Pablo Montero-Manso. Monash time series forecasting archive. *arXiv preprint arXiv:2105.06643*, 2021.

[13] Georg Goerg. Forecastable component analysis. In *International conference on machine learning*, pages 64–72. PMLR, 2013.

[14] Hongcan Guo, Haolang Lu, Guoshun Nan, Bolun Chu, Jialin Zhuang, Yuan Yang, Wenhao Che, Xinye Cao, Sicong Leng, Qimei Cui, et al. Advancing expert specialization for better moe. *arXiv preprint arXiv:2505.22323*, 2025.

[15] Xing Han, Hsing-Huan Chung, Joydeep Ghosh, Paul Pu Liang, and Suchi Saria. Guiding mixture-of-experts with temporal multimodal interactions. In *The Fourteenth International Conference on Learning Representations*, 2026.

[16] Albert Q Jiang, Alexandre Sablayrolles, Antoine Roux, Arthur Mensch, Blanche Savary, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Emma Bou Hanna, Florian Bressand, et al. Mixtral of experts. *arXiv preprint arXiv:2401.04088*, 2024.

[17] Zongyi Li, Nikola Borislavov Kovachki, Kamyar Azizzadenesheli, Burigede liu, Kaushik Bhattacharya, Andrew Stuart, and Anima Anandkumar. Fourier neural operator for parametric partial differential equations. In *International Conference on Learning Representations*, 2021.

[18] Chenghao Liu, Taha Aksu, Juncheng Liu, Xu Liu, Hanshu Yan, Quang Pham, Silvio Savarese, Doyen Sahoo, Caiming Xiong, and Junnan Li. Moirai 2.0: When less is more for time series forecasting. *arXiv preprint arXiv:2511.11698*, 2025.

[19] 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. *arXiv preprint arXiv:2410.10469*, 2024.

[20] Yong Liu, Tengge Hu, Haoran Zhang, Haixu Wu, Shiyu Wang, Lintao Ma, and Mingsheng Long. itransformer: Inverted transformers are effective for time series forecasting. *arXiv preprint arXiv:2310.06625*, 2023.

[21] 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. *arXiv preprint arXiv:2502.00816*, 2025.

[22] Spyros Makridakis, Evangelos Spiliotis, and Vassilios Assimakopoulos. M5 accuracy competition: Results, findings, and conclusions. *International journal of forecasting*, 38(4):1346–1364, 2022.

[23] Zhenxing MI and Dan Xu. Switch-neRF: Learning scene decomposition with mixture of experts for large-scale neural radiance fields. In *The Eleventh International Conference on Learning Representations*, 2023.

[24] Chengxi Min, Wei Wang, Yahui Liu, Weixin Ye, Enver Sangineto, Qi Wang, and Yao Zhao. Guiding the experts: Semantic priors for efficient and focused moe routing. *arXiv preprint arXiv:2505.18586*, 2025.

[25] Mohammad Amin Morid, Olivia R Liu Sheng, and Joseph Dunbar. Time series prediction using deep learning methods in healthcare. *ACM Transactions on Management Information Systems*, 14(1):1–29, 2023.

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

[27] Oleksandr Shchur, Abdul Fatir Ansari, Caner Turkmen, Lorenzo Stella, Nick Erickson, Pablo Guerron, Michael Bohlke-Schneider, and Yuyang Wang. fev-bench: A realistic benchmark for time series forecasting. *arXiv preprint arXiv:2509.26468*, 2025.

[28] Leyang Shen, Gongwei Chen, Rui Shao, Weili Guan, and Liqiang Nie. MoME: Mixture of multimodal experts for generalist multimodal large language models. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024.

[29] Sheng Shen, Le Hou, Yanqi Zhou, Nan Du, Shayne Longpre, Jason Wei, Hyung Won Chung, Barret Zoph, William Fedus, Xinyun Chen, Tu Vu, Yuexin Wu, Wuyang Chen, Albert Webson, Yunxuan Li, Vincent Y Zhao, Hongkun Yu, Kurt Keutzer, Trevor Darrell, and Denny Zhou. Mixture-of-experts meets instruction tuning: A winning combination for large language models. In *The Twelfth International Conference on Learning Representations*, 2024.

[30] 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 *The Thirteenth International Conference on Learning Representations*, 2025.

[31] Rui Wang, Karthik Kashinath, Mustafa Mustafa, Adrian Albert, and Rose Yu. Towards physics-informed deep learning for turbulent flow prediction. In *Proceedings of the 26th ACM SIGKDD international conference on knowledge discovery & data mining*, pages 1457–1466, 2020.

[32] Rui Wang, Steven Klee, and Alexis Roos. Time series forecastability measures. *KDD 2025 Workshop on AI for Supply Chain*, 2025.

[33] Xu Wang and Yusheng Xu. An improved index for clustering validation based on silhouette index and calinski-harabasz index. In *IOP conference series: materials science and engineering*, volume 569, page 052024. IOP Publishing, 2019.

[34] Yujie Wei, Shiwei Zhang, Hangjie Yuan, Yujin Han, Zhekai Chen, Jiayu Wang, Difan Zou, Xihui Liu, Yingya Zhang, Yu Liu, and Hongming Shan. Routing matters in moe: Scaling diffusion transformers with explicit routing guidance. In *The Fourteenth International Conference on Learning Representations*, 2026.

[35] Gerald Woo, Chenghao Liu, Akshat Kumar, Caiming Xiong, Silvio Savarese, and Doyen Sahoo. Unified training of universal time series forecasting transformers. In *Forty-first International Conference on Machine Learning*, 2024.

[36] Xun Wu, Shaohan Huang, Wenhui Wang, Shuming Ma, Li Dong, and Furu Wei. Multi-head mixture-of-experts. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024.

[37] Minjia Zhang, Conglong Li, Xiaoxia Wu, Zhewei Yao, and Yuxiong He. Samoe: Parameter efficient moe language models via self-adaptive expert combination, 2023.

[38] Zhuofan Zong, Bingqi Ma, Dazhong Shen, Guanglu Song, Hao Shao, Dongzhi Jiang, Hongsheng Li, and Yu Liu. MoVA: Adapting mixture of vision experts to multimodal context. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024.

A Additional Experimental and Implementation Details

A.1 Model Architecture Details

We evaluate five AME-TS variants at different scales: Tiny, Small, Base, Large, and Ultra. Their detailed configurations are reported in Table 4. In the dataset-specific setting, model size is selected separately for each dataset and is generally much smaller than in the foundation-model setting.

A.2 Structural Descriptor Definitions

We compute four structural descriptors for each input series: forecastability, seasonality strength, trend strength, and sparsity. These descriptors provide complementary signals for constructing the structural prior used in AME-TS.

Forecastability. Forecastability measures how predictable a series is from its frequency-domain structure. Following prior work [32, 13], we quantify it using the entropy of the normalized power spectral density. Given an input series $X = (x_0, x_1, \ldots, x_{T-1})$, let $\widetilde{X} = \mathrm{Detrend}(X)$ denote its detrended version, and let p_i denote the normalized power of the i-th frequency bin of $\widetilde{X}$. The spectral entropy is

$$H_a(\widetilde{X}) = -\sum_i p_i \log_a p_i,$$

and the forecastability score is

$$\mathrm{Forecastability}(X) = 1 - \frac{H_a(\widetilde{X})}{\log_a N_f},$$

where N_f is the number of frequency bins.

Seasonality strength. We quantify seasonality strength using STL decomposition [8]. Each series is decomposed into trend, seasonal, and remainder components, where the seasonal period is determined from the dominant Fast Fourier Transform peak frequency. Let S and R denote the seasonal and residual components, respectively. We define

$$\mathrm{SeasonalityStrength}(X) = 1 - \frac{\mathrm{Var}(R)}{\mathrm{Var}(S+R)}.$$

Trend strength. Trend strength measures the magnitude of linear change across the input window. We first min–max normalize the series to $[0, 1]$, fit a linear regression, and let $\hat{\beta}$ denote the fitted slope. We define

$$\mathrm{TrendStrength}(X) = \min(1, |\hat{\beta}|T).$$

This measures the directional change over the input window relative to the series range.

Sparsity. Sparsity captures how intermittent or repeated-value dominated a series is. We define

$$\mathrm{Sparsity}(X) = 1 - \frac{N_{\mathrm{unique}}(X)}{T},$$

where $N_{\mathrm{unique}}(X)$ is the number of unique values in the input window.

A.3 Regime Predictor Architecture and Training

The regime predictor g_ϕ is trained separately from the forecasting model and kept frozen during AME-TS training. Its role is to map a raw input series to a soft structural profile over the four structural descriptors: forecastability, seasonality strength, trend strength, and sparsity. Because these properties are not mutually exclusive, g_ϕ predicts them independently rather than assigning each series to a single discrete class. The resulting output is a soft structural profile that summarizes the forecasting-relevant characteristics of the input series.

The predictor is trained on a subset of the same pre-training pool used for the forecasting model. Computing the analytical structural descriptors for every series in the full pre-training pool would be

prohibitively expensive, so we instead construct a sampled training set by drawing approximately 98,910 random crops. For each crop, ground-truth structural descriptor targets are computed using the analytical feature pipeline described in Section 3.2.

Rather than using a single multi-output network, we train four separate single-feature predictors, one for each structural descriptor: forecastability, seasonality strength, trend strength, and sparsity. Each predictor has approximately 450K parameters, for a total of roughly 1.8M parameters across all four predictors.

The architecture consists of a multi-scale 1D CNN encoder with three parallel branches using kernel sizes 5, 11, and 21. Each branch contains two Conv1d layers with GroupNorm, GELU activations, and MaxPool(2). In parallel, we apply a self-attention block to the first-scale feature map, consisting of LayerNorm, 4-head MultiheadAttention, and mean pooling. The pooled outputs from the three CNN branches and the attention block are concatenated into a 512-dimensional representation, which is fed to an MLP head of the form $512 \to 128 \to 64 \to 1$, with GELU activations, dropout of 0.1 after the first hidden layer, and a sigmoid output in $[0, 1]$.

The four structural descriptors exhibit heterogeneous label distributions. For example, forecastability is often concentrated in a relatively narrow range, whereas sparsity spans a much broader portion of $[0, 1]$. Training directly on raw targets can therefore bias the predictors toward densely populated regions of the label distribution. To mitigate this, we apply per-feature quantile normalization to the regime targets before training. For each feature, target values are mapped to their empirical ranks within the sampled training pool, yielding approximately uniform targets on $[0, 1]$. This encourages the predictor to use the full target range rather than focusing disproportionately on highly concentrated regions. The same training quantiles are used to normalize validation targets.

We train the regime predictors using mean squared error loss on the quantile-normalized targets. Optimization uses AdamW with learning rate 10^{-3}, weight decay 10^{-4}, cosine annealing for 100 epochs, batch size 256, and early stopping with patience 15 based on validation MSE.

A.4 Pre-training Data Composition

We pre-train AME-TS on a heterogeneous corpus assembled from three public sources. First, we draw a subset of datasets from the LOTSA archive [35], which contains more than 170 datasets totaling 27 billion observations; our subset represents less than 20% of the full archive and spans traffic, weather, energy, retail, healthcare, web/cloud, and economics. Second, we incorporate several unique datasets from the Chronos pre-training corpus [3], including a one-million-series synthetic KernelSynth corpus and several real-world datasets not otherwise available in LOTSA. Third, we generate a synthetic dataset of 87,000 time series from diverse parametric models, including seasonal, trend, sparse, and noise processes, with known structural descriptor labels across four dimensions (forecastability, seasonality, trend, sparsity) to encourage expert specialization during routing. We explicitly exclude the M5 retail hierarchy from pre-training, reserving it for downstream fine-tuning and evaluation, and we verify that our corpus contains no overlap with the held-out test windows of standard forecasting benchmarks, so all reported results are free of test-data leakage. In total, the pre-training pool spans 96 dataset configurations across 8 domains (Table 3), comprising approximately 3.5 million individual time series and 18 billion observations, with frequencies ranging from 10-second to yearly. We use domain-balanced sampling with per-dataset weights to prevent large datasets (e.g., buildings_900k, kernel_synth_1m) from dominating training.

A.5 Additional Training Hyperparameters

For pre-training, we optimize AME-TS using AdamW with $\beta_1 = 0.9$, $\beta_2 = 0.98$, and weight decay 0.01. We use a peak learning rate of 5×10^{-4} with a cosine-with-restarts schedule over three cycles and a warmup of 5,000 steps. Training runs for 200 epochs with 2,000 steps per epoch, for a total of 400K optimization steps. We use a batch size of 32 per GPU on 8 GPUs, giving an effective batch size of 256, and apply gradient clipping with maximum norm 2.0. During masked forecasting pre-training, 15%–50% of input tokens are randomly masked for prediction. The maximum sequence length is 512 tokens, training uses TF32 precision, and the regime predictor is kept frozen, with its input length capped at 192 timesteps.

Table 3: Pre-training data composition (96 configurations, 8 domains).

Domain	#Datasets	#Series	#Obs.
Econ/Fin	15	104K	28M
Energy	23	1.80M	16.16B
Transport	16	5K	148M
Weather/Climate	17	70K	65M
Sales/Retail	7	246K	227M
Healthcare	6	1K	0.2M
Web/Cloud	10	207K	371M
Synthetic	2	1.09M	1.04B
Total	**96**	**3.52M**	**18.04B**

For fine-tuning on M5, we use a substantially smaller learning rate of 10^{-5} and no learning-rate schedule. The batch size is set between 8 and 16 depending on the configuration, and models are fine-tuned for 6,000–10,000 steps. We apply level-balanced sampling so that each M5 hierarchy level is sampled with equal probability during training. The regime predictor remains active, and the KL alignment loss uses the same coefficient as in pre-training.

A.6 Feature Correlation Analysis

To assess whether the proposed structural descriptors provide complementary information, Table 5 reports their Pearson correlation matrix on 98,910 training samples from the pre-training pool. The off-diagonal correlations are uniformly moderate in magnitude, with the largest absolute correlation equal to 0.296. These results suggest that forecastability, seasonality, trend, and sparsity capture distinct temporal properties and therefore provide complementary, non-redundant signals for both the regime predictor and the routing prior.

Table 4: Architecture details of AME-TS variants.

Model	d	L	E/k	Params	Active
tiny	256	4	5/2	14M	6M
small	256	5	10/2	30M	7M
base	512	5	5/2	62M	29M
large	768	6	5/2	162M	75M
ultra	1024	6	10/2	540M	133M

Table 5: Pearson correlation between the four structural descriptors across 98,910 training samples. Moderate correlations ($|r| < 0.5$) indicate that the features capture distinct temporal properties.

	Forecast.	**Season.**	**Trend**	**Sparsity**
Forecastability	1.000	0.111	0.200	−0.296
Seasonality	0.111	1.000	−0.280	−0.031
Trend	0.200	−0.280	1.000	−0.277
Sparsity	−0.296	−0.031	−0.277	1.000

A.7 Additional Results Tables

Table 6 reports the full GIFT-Eval results across all four metrics, including sMAPE and MAE, and includes the task-specific per-dataset comparison omitted from the compact main-paper summary in Table 1. Additional ablations are reported in Table 7. All public datasets and benchmarks are used according to their released licenses and terms. GIFT-Eval is released under Apache-2.0. We use public datasets from LOTSA, Chronos, and M5 through their official releases and cite their original sources.

Table 6: Forecasting performance on GIFT-Eval (97 tasks). Scores are the geometric mean of each metric normalized by the Seasonal Naive baseline (lower is better). Best results within each scale group are shown in **bold**. † Active parameters per token due to sparse MoE routing.

Scale	Model	#Params	MASE ↓	sMAPE ↓	RMSE ↓	MAE ↓
Small	Moirai-Small [35]	14M	0.946	1.109	1.056	0.906
	Chronos-Small [3]	46M	0.892	1.059	0.866	0.847
	AME-TS Tiny	**14M†6M**	0.856	1.043	0.808	0.834
	AME-TS Small	**30M†7M**	0.812	0.990	0.767	0.741
Base	Chronos-2 [2]	120M	0.698	0.886	0.704	0.679
	Sundial-128M [21]	128M	0.750	0.945	0.739	0.739
	AME-TS Base	**62M†29M**	0.765	0.926	0.737	0.725
Large	TimesFM-2.5 [10]	231M	0.705	0.856	0.694	0.687
	Moirai2 [18]	300M	0.728	0.925	0.739	0.705
	AME-TS Large	**162M†75M**	0.700	0.852	0.690	0.680
	AME-TS Ultra	**540M†133M**	0.692	0.857	0.687	0.680
Per-dataset	PatchTST [26]	–	0.849	1.020	0.839	0.810
	iTransformer [20]	–	0.893	1.074	0.858	0.850
	AME-TS	–	0.844	0.946	0.832	0.800

Table 7: Ablation study of AME-TS Tiny on GIFT-Eval benchmark.

Configuration	MASE ↓
(a) Architecture	
AME-TS Tiny (proposed)	**0.856**
Standard MoE (no prior)	0.929
Dense (no MoE)	0.958
(b) Prior integration	
Reverse KL	0.917
Additive prior	1.060
(c) Regime ablation	
Drop Forecastability	0.919
Drop Seasonality	0.889
Drop Trend	0.882
Drop Sparsity	0.910