FMoE: Frequency-Guided Multi-Period MoE Modeling for Time Series Forecasting

Taochen Chen[1], Jialiang Chen[1], Yiqun Zhang[1,*], Mengke Li[2], Yang Lu[3], and Yiu-ming Cheung[4]

[1]*School of Computer Science and Technology, Guangdong University of Technology, Guangzhou, China*

[2]*College of Computer Science and Software Engineering, Shenzhen University, Shenzhen, China*

[3]*Key Laboratory of Multimedia Trusted Perception and Efficient Computing, Ministry of Education of China, Xiamen University, Xiamen, China*

[4]*Department of Computer Science, Hong Kong Baptist University, Hong Kong, China*

Email: chentaochen1@mails.gdut.edu.cn, 3124004088@mail2.gdut.edu.cn, yqzhang@gdut.edu.cn*, mengkeli@szu.edu.cn, luyang@xmu.edu.cn, ymc@comp.hkbu.edu.hk

***Abstract*—Multivariate time-series forecasting is fundamental to decision-making in domains such as energy, transportation, and finance. To capture long-range dependencies within complex multivariate time series data, patch-based architectures have emerged as a dominant paradigm, significantly enhancing computational efficiency and semantic extraction. However, real-world time series inherently comprise superimposed periodic components with varying scales. Existing methods often overlook this intrinsic property by imposing a rigid, uniform patch granularity, thereby disrupting the semantic consistency across distinct periods. This misalignment induces scale mismatch and multi-period interference, limiting the model's ability to represent complex temporal dynamics. To address this, we propose FMoE, a frequency-guided network that aligns patch granularity with the intrinsic periodic structure. Specifically, FMoE extracts dominant frequencies from the Fourier spectrum to construct period-specific branches with adaptive patch lengths. It employs Transformer layers to learn representations within each branch, fuses outputs using weights from spectral peak strengths, and leverages a Heterogeneous Mixture-of-Experts (MoE) head to adaptively generate the final forecasts. By decomposing superimposed signals into dedicated, period-specific branches, this mechanism systematically mitigates interference caused by mixed-scale processing, enabling each dominant temporal pattern to be modeled with its optimal contextual scope. Extensive experiments on real-world benchmarks show that FMoE consistently outperforms mainstream models, demonstrating excellent performance.**

***Index Terms*—Multivariate time series, Fourier spectrum, multi-period modeling, MoE, Transformer, frequency domain**

I. Introduction

Multivariate Time Series Forecasting (MTSF) [1] is critical for decision-making in domains like energy [2], transportation [3], finance [4], and industrial monitoring [5]. By translating historical data into actionable future estimates, it enables effective resource allocation, risk management, and early warning. However, extended forecasting horizons and dynamic environments increasingly demand highly discriminative and robust representations. Consequently, MTSF remains a central challenge at the intersection of statistical modeling, signal processing, and machine learning.

This study was supported in part by the National Natural Science Foundation of China under Grants 62476063, 62376233, and 62306181; in part by the NSFC/Research Grants Council (RGC) Joint Research Scheme under Grant N_HKBU214/21; in part by the General Research Fund of RGC under grant: 12202924, and the Guangdong and Hong Kong Universities "1+1+1" Joint Research Collaboration Scheme with grant: 2025A0505000004; in part by the Natural Science Foundation of Fujian Province under Grant 2024J09001; in part by the Natural Science Foundation of Guangdong Province under Grant 2025A1515011293; in part by the Shenzhen Science and Technology Program under Grant RCBS20231211090659101; in part by the National Key Laboratory of Radar Signal Processing under Grant JKW202403.

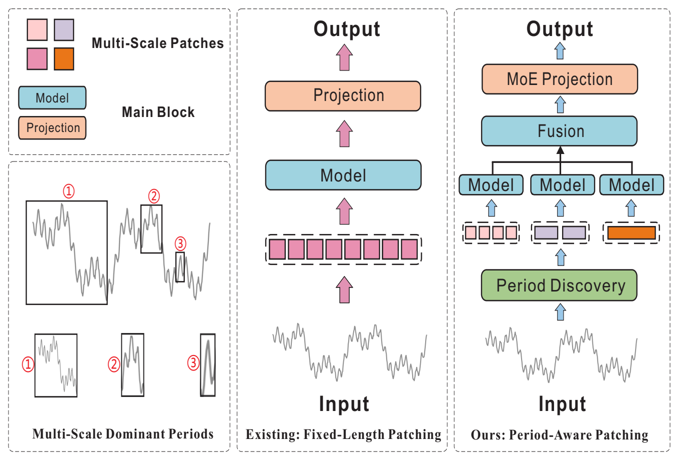

Fig. 1. Comparison between fixed-patch models and our multi-branch variable-length patch design. Fixed-patch methods treat patch length as a hyperparameter and ignore multi-period superposition, whereas our method explicitly models and fuses representations from different periods.

The evolution of time series forecasting has progressed through several paradigms. Early statistical models such as AR, MA, and ARIMA [6]–[8] perform well under linear and stable assumptions, but struggle with nonlinear dynamics and distribution shifts. To overcome these limitations, neural sequence models such as RNNs, LSTMs [9], [10], and probabilistic frameworks like DeepAR [11] were introduced to better capture nonlinear temporal patterns and uncertainty. However, their recurrent nature limits parallelization and leads to error accumulation in multi-step forecasting. To improve efficiency, convolutional and hierarchical models such as TCN, N-

BEATS, and N-HiTS [12]–[14] adopted parallel computation and multi-scale designs, yet they remain constrained by fixed receptive fields in modeling long-range dependencies. To further address these issues, the field shifted toward Transformer-based methods for long-term forecasting [15]. Architectures such as Informer [16], Autoformer [17], FEDformer [18], PatchTST [19], and iTransformer [20] leverage attention, decomposition, and patching to model extended dependencies over long histories. Nevertheless, fixed segmentation and approximate attention can still mix overlapping periodic components and weaken cross-variable modeling. To better isolate dominant temporal patterns, frequency- and decomposition-based methods such as TimesNet [21] and LHRTF [22] further exploit spectral information. However, real-world series often exhibit dynamic and time-varying periodicities, making inter-period interference a persistent challenge for stable long-term multivariate forecasting.

These challenges are more severe in long-term forecasting, where models must summarize long histories while dominant periods vary across samples and variables. In real-world multivariate time series, the difficulty is mainly driven by multi-scale information and multi-period superposition: informative patterns emerge at different temporal resolutions, while multiple recurring cycles may coexist. Meanwhile, redundant or weakly informative timestamps, together with fixed or non-adaptive temporal processing, can obscure salient structures [23]. Under such conditions, fixed patch lengths become a major limitation, as misalignment between assumed and true dominant periods leads to scale mismatch, multi-period interference, and accumulated errors. Robust forecasting also requires reliable inter-variable modeling: overly decoupled designs miss informative couplings, whereas excessive channel mixing amplifies noise. In addition, distinct variable dynamics and noise levels make a fully shared output mapping too coarse and a fully independent one costly and hard to optimize. Therefore, an effective solution should capture multi-scale information, adapt patching to dynamic multi-period structures, model inter-variable dependencies reliably, and enable efficient heterogeneous output mapping within a unified architecture.

Therefore, we propose FMoE, a Frequency-Guided Multi-Period MoE framework for multivariate long-term forecasting. FMoE addresses the misalignment between fixed patch lengths and dynamically varying dominant periods, alleviating scale mismatch and inter-period interference. As illustrated in Fig. 1, unlike fixed patching that entangles multiple periodic components within shared tokens, FMoE identifies dominant periods to construct and adaptively fuse period-aware representations. Specifically, it integrates frequency-guided period discovery, temporal and cross-variable dependency refinement, and expert-based variable-specific prediction into a unified architecture. Extensive experiments demonstrate its effectiveness. Our main contributions are summarized as follows:

- This paper proposes a novel multi-period modeling approach to segment inputs into period-aligned patches using Fourier-extracted dominant periods. Fusing period-specific branches via spectral-strength weights substantially reduces inter-period interference.
- A simple but effective FMoE architecture is proposed to explicitly model period-specific branches at different patch lengths. This architecture captures temporal and channel dependencies for each period and leverages the heterogeneous mapping of a MoE Head, thereby enabling the model to focus on dominant periodic pattern shifts and enhance forecasting accuracy.
- Extensive experiments show FMoE's consistent superiority over mainstream baselines. Beyond these performance improvements, analysis of training dynamics and forecasting visualizations confirms its robustness: FMoE exhibits superior training stability and structurally separates distinct periods to capture sharp peaks, which can be easily smoothed out by fixed-patch methods.

II. Related Work

A. Patch-based Architectures

Patch-based forecasting improves long-term efficiency by compressing sequences into patches and modeling dependencies at the patch level. PatchTST [19] treats patches as tokens and processes channels independently, offering good scalability but limited inter-variable modeling. To better capture inter-variable structure, iTransformer [20] instead treats variables as tokens to enhance cross-variable interactions but may weaken within-variable temporal modeling. STNet [24] captures both inter-variable and within-variable dependencies through a spatial-temporal design, yet it still relies on fixed patching. Recent variants further expand the design space: xPatch [25] improves robustness through patching, channel-independent processing, and dual-stream decomposition, while PatchMLP [26] adopts a patch-based MLP with multi-scale patching and component-wise processing to reduce backbone complexity. Despite strong empirical results, most patch-based methods still use fixed patch sizes or static multi-scale grids, which are suboptimal under overlapping periodicities and time-varying dominant scales. Aligning patch granularity with dominant periodic structures while handling scale mismatch and multi-period interference remains a key challenge.

B. Mixture-of-Experts Mechanisms

Mixture-of-Experts (MoE) [27] is a classical conditional computation paradigm that has regained attention with advances in large-scale models [28]. In time series forecasting, however, MoE remains relatively underexplored and is mostly used for specific subproblems. FiLM [29] applies MoE across input horizons to adapt to different temporal contexts but does not explicitly model output heterogeneity across variables and samples. FEDformer [18] incorporates an MoE-style design for data-dependent component fusion, yet mainly at the backbone and component levels. Time-MoE [30] further shows that sparse MoE can scale time series foundation models efficiently but still does not directly address variable- and sample-adaptive output mappings under heterogeneous multivariate dynamics. Overall, existing MoE designs in time series forecasting mainly serve backbone scaling or component

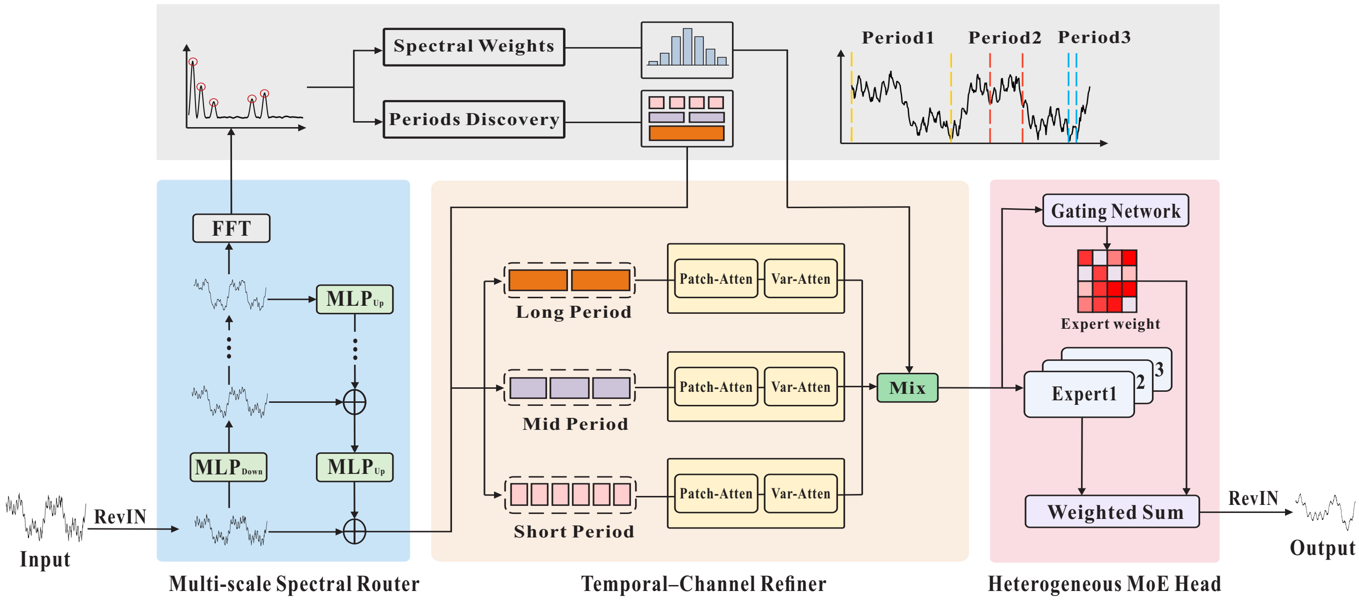

Fig. 2. FMoE Structure Overview, which consists of three components: (1) Multi-scale Spectral Router, (2) Temporal-Channel Refiner, and (3) Heterogeneous MoE Head, designed for time series forecasting.

fusion, while how to exploit expert specialization for variable- and sample-adaptive output mapping with parameter efficiency and balanced expert utilization remains a key challenge.

III. Proposed Method

Given a multivariate time series $X = [\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_T] \in \mathbb{R}^{C\times T}$ with $\mathbf{x}_t \in \mathbb{R}^C$, multivariate long-term forecasting aims to predict the future sequence $Y = [\mathbf{y}_1, \mathbf{y}_2, \ldots, \mathbf{y}_H] \in \mathbb{R}^{C\times H}$ over a forecasting window of length H, where $\mathbf{y}_h \in \mathbb{R}^C$ denotes the prediction at forecast step h.

A. Architecture Overview

As shown in Fig. 2, FMoE consists of three components: the Multi-scale Spectral Router (MSR), the Temporal-Channel Refiner (TCR), and the Heterogeneous MoE Head (HMH). MSR enriches the input with multi-scale context and extracts period information from the Fourier spectrum of a coarse view to construct period-specific branches and compute instance-wise fusion weights. Each branch stacks TCR blocks to refine representations at the corresponding patch granularity. HMH then models heterogeneous variable-wise mappings with a mixture-of-experts head to generate forecasts. To mitigate time-series non-stationarity, we adopt Reversible Instance Normalization for normalization and denormalization.

B. Multi-scale Spectral Router

Given the normalized input $X^{(0)} \in \mathbb{R}^{C\times T}$, a pyramid of temporal views is generated by cascaded MLP downsampling $X^{(\ell)} = \mathrm{MLPDown}_\ell\big(X^{(\ell-1)}\big)$, which reduces the temporal length from $T/r^{\ell-1}$ to T/r^ℓ and produces $X^{(\ell)} \in \mathbb{R}^{C\times(T/r^\ell)}$ for $\ell \in \{1, \ldots, L\}$, where $r > 1$ is the downsampling ratio and L is the number of stages.

Subsequently, coarse-scale information is injected back into finer resolutions through stage-wise residual mixing. The fused hierarchy is initialized by $\widetilde{X}^{(L)} = X^{(L)}$, and the next finer fused representation is computed as:

$$\widetilde{X}^{(\ell-1)} = X^{(\ell-1)} + \mathrm{MLPUp}_\ell\Big(\widetilde{X}^{(\ell)}\Big), \quad (1)$$

where $\mathrm{MLPUp}_\ell(\cdot)$ projects $\widetilde{X}^{(\ell)}$ from length T/r^ℓ to $T/r^{\ell-1}$ for shape compatibility. This residual design enriches fine-scale structures with coarse-scale context, yielding the final multi-scale representation $U = \widetilde{X}^{(0)} \in \mathbb{R}^{C\times T}$.

MSR also provides spectral priors for period-aware modeling. An FFT is applied to the coarsest view $X^{(L)} \in \mathbb{R}^{C\times T_c}$ with $T_c = T/r^L$ to extract dominant frequencies and periods:

$$\mathbf{A}, \{f_1, \cdots, f_K\}, \{p_1, \cdots, p_K\} = \mathrm{FFT}\Big(X^{(L)}\Big), \quad (2)$$

where $\mathbf{A} = \{A_{f_1}, \ldots, A_{f_K}\}$ denotes the unnormalized amplitudes, and $\{f_1, \ldots, f_K\}$ are the top-K frequencies selected from the FFT spectrum. For each f_k, the period on the coarsest view is computed as $p_k = \left\lfloor \frac{T_c}{f_k} \right\rfloor$ and then rescaled to the original resolution by the overall downsampling factor $d = T/T_c$, giving $\tilde{p}_k = d\, p_k$.

C. Temporal-Channel Refiner

From the MSR output, we obtain dominant periods $\{\tilde{p}_k\}_{k=1}^K$ and corresponding instance-wise spectral strengths $\{s_k(x)\}_{k=1}^K$. Each period is mapped to a feasible patch length $P_k = \mathrm{Snap}(\tilde{p}_k)$ with $T \bmod P_k = 0$, where $\mathrm{Snap}(\cdot)$ returns the nearest divisor of T. The resulting representation $U \in \mathbb{R}^{C\times T}$ is then partitioned into $N_k = T/P_k$ non-overlapping patches for branch k. Each branch first models intra-channel inter-patch dependencies and then cross-variable interactions:

$$U_k^{\mathrm{patch}} = U + \mathrm{PatchAttn}(U; P_k)\,, \quad (3)$$

$$U_k^{\mathrm{out}} = U_k^{\mathrm{patch}} + \gamma\, \mathrm{VarAttn}\Big(U_k^{\mathrm{patch}}\Big)\,, \quad (4)$$

TABLE I

COMPARING FMOE WITH SOTA BASELINES FOR MULTIVARIATE LONG-TERM FORECASTING RESULTS. WE USE PREDICTION LENGTHS $H \in \{96, 192, 336, 720\}$, AND INPUT LENGTH $T = 96$. THE BEST RESULTS ARE IN **BOLD** AND THE SECOND BEST ARE $\underline{\text{UNDERLINED}}$.

Models		**FMoE (ours)**		**xPatch AAAI'25**		**PatchMLP AAAI'25**		**TimeMixer ICLR'24**		**iTransformer ICLR'24**		**TimesNet ICLR'23**		**PatchTST ICLR'23**		**DLinear AAAI'23**	
Metric		MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE
ETTm1	96	**0.316**	**0.351**	0.332	0.369	0.329	0.367	$\underline{0.324}$	$\underline{0.363}$	0.341	0.376	0.335	0.400	0.328	0.368	0.346	0.374
	192	**0.359**	**0.380**	0.371	$\underline{0.387}$	0.377	0.394	0.372	0.390	0.381	0.395	0.407	0.414	$\underline{0.369}$	0.391	0.382	0.391
	336	**0.394**	**0.400**	$\underline{0.399}$	$\underline{0.407}$	0.411	0.413	**0.394**	0.408	0.419	0.419	0.414	0.422	0.400	0.408	0.415	0.415
	720	$\underline{0.461}$	**0.438**	0.463	$\underline{0.443}$	0.485	0.454	**0.454**	$\underline{0.443}$	0.486	0.456	0.512	0.473	0.457	0.444	0.473	0.451
ETTm2	96	**0.175**	$\underline{0.258}$	$\underline{0.177}$	0.262	0.178	0.262	**0.175**	**0.257**	0.184	0.267	0.189	0.266	0.183	0.267	0.193	0.293
	192	**0.238**	**0.300**	0.244	0.306	0.244	0.307	$\underline{0.239}$	$\underline{0.301}$	0.253	0.312	0.263	0.312	0.245	0.305	0.285	0.361
	336	$\underline{0.296}$	**0.336**	0.304	0.342	0.307	0.343	**0.295**	$\underline{0.338}$	0.315	0.352	0.307	0.340	0.309	0.349	0.385	0.429
	720	$\underline{0.397}$	**0.394**	0.404	$\underline{0.399}$	0.420	0.413	**0.396**	$\underline{0.399}$	0.412	0.406	0.418	0.405	0.420	0.415	0.556	0.523
ETTh1	96	**0.376**	**0.390**	0.383	0.401	0.392	0.405	0.383	$\underline{0.400}$	0.395	0.410	0.389	0.412	$\underline{0.377}$	0.397	0.396	0.411
	192	**0.426**	**0.419**	$\underline{0.431}$	$\underline{0.427}$	0.442	0.434	0.441	0.430	0.449	0.441	0.439	0.441	**0.426**	0.431	0.445	0.440
	336	**0.470**	**0.440**	0.483	$\underline{0.455}$	0.483	$\underline{0.455}$	0.499	0.459	0.492	0.465	0.494	0.471	$\underline{0.474}$	0.463	0.487	0.465
	720	**0.474**	**0.463**	0.486	$\underline{0.474}$	0.502	0.484	$\underline{0.480}$	$\underline{0.474}$	0.522	0.504	0.518	0.494	0.529	0.508	0.513	0.510
ETTh2	96	**0.281**	**0.335**	$\underline{0.291}$	$\underline{0.343}$	0.309	0.359	0.301	0.348	0.301	0.349	0.337	0.371	$\underline{0.291}$	0.345	0.341	0.395
	192	$\underline{0.374}$	**0.389**	**0.372**	$\underline{0.392}$	0.405	0.419	0.387	0.406	0.382	0.400	0.405	0.415	0.382	0.406	0.482	0.479
	336	**0.410**	**0.424**	0.423	0.433	0.432	0.441	0.428	0.443	0.424	$\underline{0.432}$	0.454	0.451	$\underline{0.416}$	0.434	0.593	0.542
	720	**0.416**	**0.438**	0.430	0.451	0.460	0.446	0.476	0.473	$\underline{0.427}$	$\underline{0.445}$	0.434	0.448	0.435	0.454	0.840	0.661
Exchange	96	**0.085**	**0.204**	$\underline{0.089}$	0.208	0.093	0.213	0.093	0.212	$\underline{0.089}$	0.210	0.107	0.236	$\underline{0.089}$	$\underline{0.207}$	0.098	0.233
	192	$\underline{0.175}$	**0.296**	0.183	0.304	0.185	0.307	**0.174**	$\underline{0.297}$	0.186	0.305	0.224	0.343	0.182	0.303	0.186	0.325
	336	**0.325**	**0.414**	0.338	0.422	0.339	0.423	0.346	0.424	0.334	$\underline{0.418}$	0.346	0.432	0.342	0.426	$\underline{0.327}$	0.435
	720	$\underline{0.810}$	**0.680**	0.876	0.707	0.880	0.706	0.919	0.718	0.875	0.711	0.930	0.734	0.934	0.726	**0.789**	$\underline{0.695}$
Weather	96	0.175	0.220	0.178	0.221	**0.165**	$\underline{0.217}$	0.175	0.219	0.174	**0.214**	$\underline{0.169}$	0.219	0.175	$\underline{0.217}$	0.196	0.256
	192	0.222	0.260	0.231	0.263	**0.212**	$\underline{0.258}$	0.226	0.263	0.247	0.281	0.225	0.265	$\underline{0.221}$	**0.256**	0.239	0.299
	336	**0.277**	**0.294**	0.283	0.300	0.283	0.309	0.283	0.304	0.283	0.301	0.282	0.305	$\underline{0.281}$	$\underline{0.298}$	$\underline{0.281}$	0.331
	720	$\underline{0.351}$	**0.344**	0.360	0.350	0.357	0.357	0.355	0.351	0.359	0.351	0.354	0.351	0.357	$\underline{0.349}$	**0.345**	0.382
1st Ct.		15	21	1	0	2	0	6	1	0	1	0	0	1	1	2	0

where U_k^{out} denotes the output of the k-th TCR branch with patch length $P_k = \text{Snap}(\tilde{p}_k)$. $\text{PatchAttn}(\cdot; P_k)$ captures global dependencies across the N_k patch tokens, while γ controls the strength of cross-channel mixing in $\text{VarAttn}(\cdot)$.

The multi-branch representations are fused into a unified period-aware representation using sample-wise spectral strengths. For a sample x, let $\{s_k(x)\}_{k=1}^{K}$ denote its spectral strengths. The fusion weights are computed by softmax:

$$\alpha_k(x) = \frac{\exp(s_k(x))}{\sum_{j=1}^{K} \exp(s_j(x))}. \tag{5}$$

The final representation is obtained by weighted fusion:

$$Z(x) = \sum_{k=1}^{K} \alpha_k(x)\, U_k^{\text{out}}(x) \in \mathbb{R}^{C \times T}. \tag{6}$$

D. Heterogeneous MoE Head

Given the fused representation $Z(x) \in \mathbb{R}^{C \times T}$, HMH performs variable-wise gating. For each variable c, the gating network takes $Z_{c,:}(x)$ as input and outputs expert weights:

$$\begin{gathered} \mathbf{g}_c(x) = \text{Softmax}(\phi(\text{Gate}(Z_{c,:}(x)))) \in \mathbb{R}^{E}, \\ \sum_{e=1}^{E} g_{c,e}(x) = 1, \end{gathered} \tag{7}$$

where $\text{Gate}(\cdot)$ maps the input of length T to E gating logits, and $g_{c,e}(x)$ is the weight assigned to expert e. $\phi(\cdot)$ is a top-k-biased transformation that sharpens the post-softmax expert distribution. During training, Gaussian noise is added to the gating logits to improve exploration and robustness.

Each expert f_e is implemented as a two-layer MLP that maps the history of variable c to an H-step forecast:

$$\hat{\mathbf{y}}_{c,:}^{(e)}(x) = f_e(Z_{c,:}(x)) \in \mathbb{R}^{H}. \tag{8}$$

The final prediction is obtained by a weighted mixture of expert outputs:

$$\hat{\mathbf{Y}}_{c,:}(x) = \sum_{e=1}^{E} g_{c,e}(x)\, \hat{\mathbf{y}}_{c,:}^{(e)}(x). \tag{9}$$

E. Loss Function

To mitigate expert collapse and encourage balanced expert usage, we introduce a load-balancing regularizer. For the current mini-batch $\mathcal{S}$, the expert importance for channel c is defined as the accumulated expert weights:

$$\begin{gathered} I_{c,e} = \sum_{x \in \mathcal{S}} g_{c,e}(x), \\ \mathbf{I}_c = (I_{c,1}, \ldots, I_{c,E}) \in \mathbb{R}^{E}. \end{gathered} \tag{10}$$

TABLE II
ABLATION RESULTS ON DIFFERENT DATASETS.

Methods		w/o MSR		Fixed-Patch		w/o HMH		FMoE	
Metric		MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE
ETTh1	96	0.382	0.396	0.384	0.399	0.394	0.413	**0.376**	**0.390**
	192	0.432	0.423	0.436	0.425	0.447	0.443	**0.426**	**0.419**
	336	0.476	0.444	0.478	0.447	0.483	0.462	**0.470**	**0.440**
	720	0.480	0.466	0.493	0.470	0.487	0.482	**0.474**	**0.463**
ETTm1	96	0.382	0.390	0.326	0.360	0.326	0.366	**0.316**	**0.351**
	192	0.421	0.409	0.376	0.387	0.368	0.391	**0.359**	**0.380**
	336	0.450	0.429	0.406	0.404	0.407	0.412	**0.394**	**0.400**
	720	0.521	0.465	0.467	0.439	0.466	0.448	**0.461**	**0.438**
Exchange	96	0.114	0.240	0.092	0.207	0.090	0.210	**0.085**	**0.204**
	192	0.213	0.330	0.183	0.302	0.178	0.300	**0.175**	**0.296**
	336	0.371	0.443	0.328	0.416	0.328	0.420	**0.325**	**0.414**
	720	0.910	0.730	0.817	0.684	0.880	0.717	**0.810**	**0.680**

The load-balancing term is computed as the squared coefficient of variation of $\mathbf{I}_c$ and summed across variables:

$$\mathcal{L}_{\text{bal}} = \sum_{c=1}^{C} \frac{\operatorname{Var}(\mathbf{I}_c)}{\operatorname{Mean}(\mathbf{I}_c)^2 + \varepsilon}, \tag{11}$$

where ε ensures numerical stability. The overall training objective minimizes the forecasting Mean Squared Error ($\mathcal{L}_{\text{mse}}$) combined with this regularizer:

$$\mathcal{L} = \mathcal{L}_{\text{mse}} + \lambda \, \mathcal{L}_{\text{bal}}, \tag{12}$$

where λ controls the strength of the regularization term.

IV. EXPERIMENTS

A. Experimental Settings

1) Datasets: We evaluate FMoE on six public long-term forecasting benchmarks across energy, meteorology, and finance: Weather, Exchange Rate, and four ETT datasets. We split the ETT datasets into training, validation, and test sets with a 6:2:2 ratio, and the others with a 7:1:2 ratio.

2) Baselines and setup: FMoE is compared against a set of representative forecasting models, including xPatch [25], PatchMLP [26], TimeMixer [31], PatchTST [19], iTransformer [20], TimesNet [21], and DLinear [32]. All methods share the same protocol, utilizing the Adam optimizer, a look-back window $T = 96$, and prediction lengths $H \in \{96, 192, 336, 720\}$. Performance metrics are MSE and MAE. All experiments utilize a 32GB NVIDIA RTX 5090 GPU.

B. Main Results

Table I reports the results of different methods. Overall, our model outperforms competing baselines on most datasets. The gains are particularly clear on the ETT datasets, where multiple periodic components are strongly superimposed. In these cases, MSR captures dominant multi-period patterns from coarse-resolution representations, while TCR models inter-period and inter-variable dependencies, enabling consistent improvements over conventional patch-based architectures. On the Weather dataset, whose signals are driven more by irregular variations than clear periodic patterns, our model is slightly below the strongest baseline on a few metrics. Even so, it remains highly competitive, achieving 36 first-place and 8 second-place results across 48 evaluation settings.

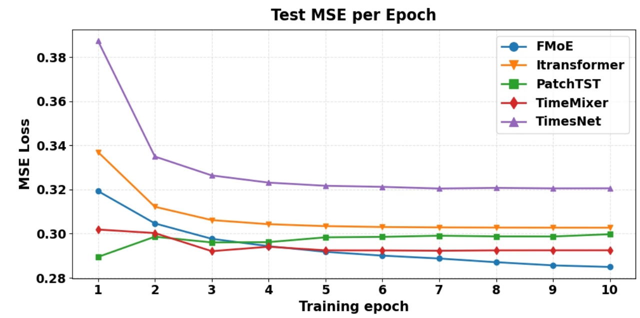

Fig. 3. Evolution of Test Performance: FMoE vs. Baselines on ETTh2.

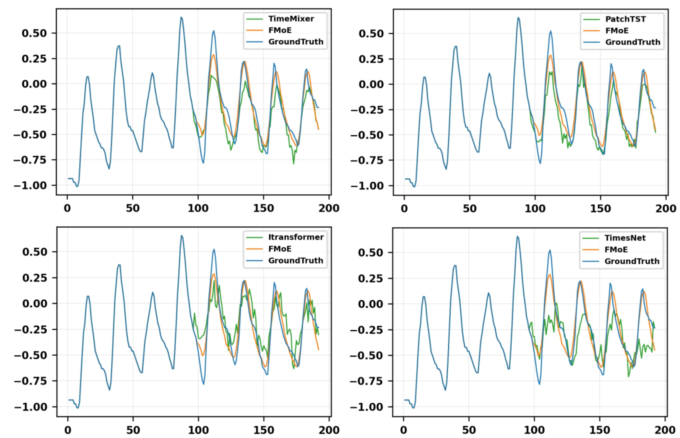

Fig. 4. Comparative Forecasting Performance: FMoE vs. Baseline Models on a Representative Test Sample.

C. Ablation Study

Table II compares full FMoE against three ablation variants, where it consistently achieves the lowest errors. This performance gap highlights three key findings: First, removing the multi-scale mechanism (**w/o MSR**) degrades performance, confirming that MSR is essential for robust period discovery. Second, applying a uniform patch length (**Fixed-Patch**) rather than FFT-driven selection demonstrates that adaptive patching significantly outperforms rigid designs. Third, replacing the MoE head with a static MLP (**w/o HMH**) reveals that dynamic expert assignment enhances predictions for variables with diverse temporal dynamics.

D. Visualization Analysis

To intuitively illustrate the model's behavior, we visualize FMoE on the ETTh2 dataset. We compare its test performance evolution and prediction curves against four representative baselines: PatchTST [19], TimeMixer [31], iTransformer [20], and TimesNet [21].

1) Evolution of Test Performance: Fig. 3 shows the test MSE on ETTh2 across 10 epochs. Baselines such as iTransformer and TimeMixer drop rapidly at early epochs but soon plateau or fluctuate, indicating early saturation. In contrast, FMoE

improves more steadily and continuously. Although its initial decline is less sharp, it avoids early saturation, surpasses all baselines by the 5th epoch, and achieves the lowest final loss. This sustained improvement suggests that FMoE enables more robust optimization and representation learning.

2) Visualization of Forecasting Results: Fig. 4 compares FMoE and four baselines on a representative test sample. The orange FMoE curve closely tracks the blue Ground Truth, capturing global periodic trends and local fluctuations. During rapid changes, iTransformer and TimesNet produce noticeable jagged patterns. This instability likely occurs because iTransformer neglects intra-variate dependencies, while TimesNet's limited CNN receptive field restricts long-term inter-period modeling. Similarly, TimeMixer and PatchTST exhibit phase lags and underestimated amplitudes, missing true peaks due to their struggle with multi-period superposition. By explicitly modeling multiple periodic patterns, FMoE better captures complex temporal dynamics, accurately identifies turning points, and preserves signal magnitudes.

V. Concluding Remarks

In this paper, we propose FMoE, a frequency-guided multi-period MoE modeling approach for multivariate long-term time series forecasting. FMoE extracts dominant periods from a coarse spectral view to build period-specific branches with different patch granularities and adaptively fuses their representations via normalized spectral strengths to mitigate multi-period superposition. Experiments on six benchmarks demonstrate consistent improvements over strong baselines. Despite these results, FMoE introduces higher computational overhead than single-stream methods, and its reliance on spectral information may limit performance on highly chaotic or strongly non-stationary signals without clear periodic structure. Future work will focus on more efficient branching designs and broader applications.

References

[1] B. Lim and S. Zohren, "Time-series forecasting with deep learning: a survey," *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, vol. 379, no. 2194, 2021.

[2] T. Hong and S. Fan, "Probabilistic electric load forecasting: A tutorial review," *International Journal of Forecasting*, vol. 32, no. 3, pp. 914–938, 2016.

[3] D. A. Tedjopurnomo, Z. Bao, B. Zheng, F. M. Choudhury, and A. K. Qin, "A survey on modern deep neural network for traffic prediction: Trends, methods and challenges," *IEEE Transactions on Knowledge and Data Engineering*, vol. 34, no. 4, pp. 1544–1561, 2020.

[4] O. B. Sezer, M. U. Gudelek, and A. M. Ozbayoglu, "Financial time series forecasting with deep learning: A systematic literature review: 2005–2019," *Applied Soft Computing*, vol. 90, p. 106181, 2020.

[5] M. A. Farahani, F. E. Kalach *et al.*, "Time-series forecasting in smart manufacturing systems: An experimental evaluation of the state-of-the-art algorithms," *Robotics and Computer-Integrated Manufacturing*, vol. 95, p. 103010, 2025.

[6] C. Meek, D. M. Chickering, and D. Heckerman, "Autoregressive tree models for time-series analysis," in *Proceedings of the 2002 SIAM International Conference on Data Mining*. SIAM, 2002, pp. 229–244.

[7] S. Hansun, "A new approach of moving average method in time series analysis," in *2013 Conference on New Media Studies (CoNMedia)*. IEEE, 2013, pp. 1–4.

[8] G. E. Box, G. M. Jenkins, G. C. Reinsel, and G. M. Ljung, *Time series analysis: forecasting and control*. John Wiley & Sons, 2015.

[9] J. L. Elman, "Finding structure in time," *Cognitive Science*, vol. 14, no. 2, pp. 179–211, 1990.

[10] A. Graves, "Long short-term memory," *Supervised sequence labelling with recurrent neural networks*, pp. 37–45, 2012.

[11] D. Salinas, V. Flunkert, J. Gasthaus, and T. Januschowski, "DeepAR: Probabilistic forecasting with autoregressive recurrent networks," *International Journal of Forecasting*, vol. 36, no. 3, pp. 1181–1191, 2020.

[12] S. Bai, "An empirical evaluation of generic convolutional and recurrent networks for sequence modeling," *arXiv preprint arXiv:1803.01271*, 2018.

[13] B. N. Oreshkin, D. Carpov, N. Chapados, and Y. Bengio, "N-BEATS: Neural basis expansion analysis for interpretable time series forecasting," *arXiv preprint arXiv:1905.10437*, 2019.

[14] C. Challu, K. G. Olivares, B. N. Oreshkin, F. G. Ramirez, M. M. Canseco, and A. Dubrawski, "NHITS: Neural Hierarchical Interpolation for Time Series Forecasting," in *Proc. AAAI*, vol. 37, no. 6, 2023, pp. 6989–6997.

[15] Q. Wen, T. Zhou, C. Zhang, W. Chen, Z. Ma, J. Yan, and L. Sun, "Transformers in time series: A survey," *arXiv preprint arXiv:2202.07125*, 2022.

[16] H. Zhou, S. Zhang, J. Peng, S. Zhang, J. Li, H. Xiong, and W. Zhang, "Informer: Beyond efficient transformer for long sequence time-series forecasting," in *Proc. AAAI*, vol. 35, no. 12, 2021, pp. 11 106–11 115.

[17] H. Wu, J. Xu, J. Wang, and M. Long, "Autoformer: Decomposition transformers with auto-correlation for long-term series forecasting," *Proc. NeurIPS*, vol. 34, pp. 22 419–22 430, 2021.

[18] T. Zhou, Z. Ma, Q. Wen, X. Wang, L. Sun, and R. Jin, "FEDformer: Frequency Enhanced Decomposed Transformer for Long-Term Series Forecasting," in *International Conference on Machine Learning*. PMLR, 2022, pp. 27 268–27 286.

[19] Y. Nie, "A time series is worth 64 words: Long-term forecasting with transformers," *arXiv preprint arXiv:2211.14730*, 2022.

[20] Y. Liu, T. Hu, H. Zhang, H. Wu, S. Wang, L. Ma, and M. Long, "iTransformer: Inverted transformers are effective for time series forecasting," *arXiv preprint arXiv:2310.06625*, 2023.

[21] H. Wu, T. Hu, Y. Liu, H. Zhou, J. Wang, and M. Long, "TimesNet: Temporal 2d-variation modeling for general time series analysis," *arXiv preprint arXiv:2210.02186*, 2022.

[22] Z. Zhang, Y. Zhang *et al.*, "Learning hierarchical representations in temporal and frequency domains for time series forecasting," in *Pattern Recognition and Computer Vision: 6th Chinese Conference, PRCV 2023, Proceedings, Part IX*, 2023, pp. 91–103.

[23] Z. Tan, X. Luo, Y. Liu, and Y. Zhang, "Mask the redundancy: Evolving masking representation learning for multivariate time-series clustering," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 40, 2026, pp. 25 787–25 795.

[24] K. Yang, X. Wang, Z. Wang, C. Chi, C. Deng, and J. Feng, "STNet: Spatial-Temporal Transformers are Effective for Multivariate Time Series Forecasting," in *Proc. IJCNN*, 2025.

[25] A. Stitsyuk and J. Choi, "xPatch: Dual-stream time series forecasting with exponential seasonal-trend decomposition," in *Proc. AAAI*, vol. 39, no. 19, 2025, pp. 20 601–20 609.

[26] P. Tang and W. Zhang, "Unlocking the Power of Patch: Patch-Based MLP for Long-Term Time Series Forecasting," in *Proc. AAAI*, vol. 39, no. 12, 2025, pp. 12 640–12 648.

[27] R. A. Jacobs, M. I. Jordan, S. J. Nowlan, and G. E. Hinton, "Adaptive mixtures of local experts," *Neural Computation*, vol. 3, no. 1, pp. 79–87, 1991.

[28] D. Lepikhin, H. Lee, Y. Xu, D. Chen, O. Firat, Y. Huang, M. Krikun, N. Shazeer, and Z. Chen, "Gshard: Scaling giant models with conditional computation and automatic sharding," *arXiv preprint arXiv:2006.16668*, 2020.

[29] T. Zhou, Z. Ma, Q. Wen, L. Sun, T. Yao, W. Yin, R. Jin *et al.*, "FiLM: Frequency improved Legendre Memory Model for Long-term Time Series Forecasting," *Proc. NeurIPS*, vol. 35, pp. 12 677–12 690, 2022.

[30] X. Shi, S. Wang, Y. Nie, D. Li, Z. Ye, Q. Wen, and M. Jin, "Time-MoE: Billion-scale time series foundation models with mixture of experts," *arXiv preprint arXiv:2409.16040*, 2024.

[31] S. Wang, H. Wu, X. Shi, T. Hu, H. Luo, L. Ma, J. Y. Zhang, and J. Zhou, "TimeMixer: Decomposable multiscale mixing for time series forecasting," *arXiv preprint arXiv:2405.14616*, 2024.

[32] A. Zeng, M. Chen, L. Zhang, and Q. Xu, "Are transformers effective for time series forecasting?" in *Proc. AAAI*, vol. 37, no. 9, 2023, pp. 11 121–11 128.