ADA-MOF: ADAPTIVE MIXTURE OF FREQUENCY-DOMAIN EXPERTS FOR TIME SERIES FORECASTING

Anonymous authors
Paper under double-blind review

ABSTRACT

Multivariate time series forecasts are widely used, such as meteorology, transportation and financial forecasts. However, the dominant frequencies in time series may shift with the evolving spectral distribution of the data. Traditional Mixture of Experts (MoE) models, which employ a fixed number of experts, struggle to adapt to these changes, resulting in frequency coverage imbalance issue. Specifically, too few experts can lead to the overlooking of critical information, while too many can introduce noise. To this end, we propose Ada-MoF, an adaptive Mixture of Frequency-domain Experts model. Ada-MoF integrates univariate spectral intensity and cross-variable frequency response features to adaptively determine the number of experts, ensuring alignment with the input data's frequency distribution. This approach prevents both information loss due to an insufficient number of experts and noise contamination from an excess of experts. The experimental results show that our model achieves state-of-the-art performance on six public benchmarks with only 0.2 million parameters.

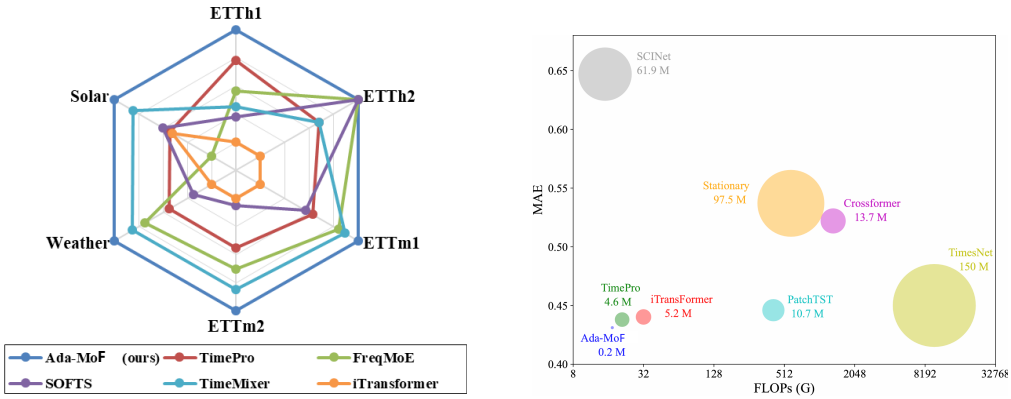
1 INTRODUCTION

Time series forecasting models hold immense application value and are widely utilized in fields such as meteorology, finance, and industrial manufacturing. Historically, time series forecasting models have primarily been categorized into four architectural families: RNN Hochreiter & Schmidhuber (1997), MLP Zeng et al. (2022), Transformer Vaswani et al. (2023), and Mamba Gu & Dao (2024). In recent years, Mixture-of-Experts (MoE) models have gained popularity in large language models due to their diverse feature representations and accelerated inference capabilities. The inherent characteristic of mixture of experts models, where different experts handle different features, naturally lends itself to processing features of varying frequencies in time series forecasting tasks. Frequency domain features represent the periodicity of signals, and capturing complex periodic signals is crucial for time series forecasting tasks.Therefore, the application of mixture of experts in time series forecasting tasks is urgently in need of exploration.

Recently, several hybrid expert-based time series forecasting models have been proposed. Time-MoE Shi et al. (2025) constructes a MoE model with 2.4 billion parameters, replacing the feed-forward layers in the transformer with MoE layers. However, the model primarily focuses on time-domain features while neglecting the importance of frequency-domain features. MoFE-Time Liu et al. (2025) equips every expert with parallel FFT MLP and time domain MLP branches. However, it merely applies the expert to the whole spectrum without decoupling individual frequencies, so dominant harmonics stay mixed with noisy bins and are easily missed. FreqMoE Liu (2025) partitions the spectrum into sub-bands using standard band-pass filters and routes each sub-band to specialized expert modules for computation. However, the use of standard band-pass filters inevitably causes abrupt band truncation, which introduces edge oscillations due to the Gibbs phenomenon when the signal is transformed back into the time domain. These oscillations inject spurious artifacts, and such distortions ultimately constrain the performance of the model.

Furthermore, employing the Hard MoE method that retains the top-K experts in the frequency domain also presents inherent limitations. Since the number of experts in a Hard MoE model is fixed, the allocation of spectral coverage can easily become imbalanced. Formally, let the effective frequency range of the signal be denoted as $\Delta f = f_{\max} - f_{\min}$, and let w represent the characteristic bandwidth that a single expert can reliably model. To achieve complete coverage of the dominant

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.



(a) The Performance of Ada-MoF

(b) Comparison of Parameters and FLOPs

Figure 1: Performance comparison of Ada-MoF with other state-of-the-art Models. Figure (a) shows a radar map based on MSE which shows that Ada-MoF has achieved advanced performance on six public benchmarks. Figure (b) shows the parameters and FLOPs of Ada-MoF versus other state-of-the-art models. The parameter of Ada-MoF is only 0.2M, and the FLOPs are significantly less than those of the existing models. And the MAE on ETTh1 of our model is significantly lower than that of other models.

frequency domain, the number of experts should satisfy $K \approx \Delta f / w$. However, Δf varies across different datasets and even across different time intervals within the same dataset. If $K < \Delta f / w$, then certain dominant frequencies will be omitted, leading to under-coverage and loss of critical periodic components. Conversely, if $K > \Delta f / w$, experts will overlap excessively, introducing redundant coverage and allowing noise bands to dominate. This imbalance can be quantified by defining the coverage ratio $\rho = K \cdot w / \Delta f$, where $\rho < 1$ indicates insufficient coverage and $\rho > 1$ indicates redundant coverage. Both cases degrade the quality of the learned representation, leading to a lower effective prediction SNR and higher forecast error. The above theoretical analysis indicates the necessity of adopting an adaptive expert allocation mechanism that can dynamically adjust K according to the spectral characteristics of the data.

To address the frequency coverage imbalance issue, we propose an adaptive Mixture of Frequency-domain Experts model, named Ada-MoF, which can adaptively select the number of experts based on the input data. It simultaneously computes univariate spectral intensity and cross-variable frequency response features for fusion, and uses the fused features to adaptively determine the number of activated experts. The joint representation of frequency and variable dimensions enables the model to learn high-energy regions representing consensus spectra of dominant frequencies and high-energy channels representing sensitive variables, thereby activating only the experts that process dominant frequencies as much as possible. Experts with higher noise levels are explicitly turned off to reduce noise interference. This approach effectively resolves the frequency coverage imbalance caused by processing different data. The experimental results in Figure 1 demonstrate that our model has achieved state-of-the-art performance on six benchmark datasets, with only 0.2M parameters and significantly lower FLOPs compared to existing methods.

2 Related Work

2.1 Time Series Forecasting Method

Current advanced time series forecasting primarily includes machine learning models, linear models, Transformer, Mamba, and other architectures. Firstly, machine learning models like ARIMA and SARIMA provide a foundational theoretical framework Reddy et al. (2025). And tree-based ensemble methods such as Random Forest and XGBoost excel at capturing nonlinear relationships Paliari et al. (2021). In addition, for improved handling of challenges such as multi-frequency data fusion, enhanced LSTM architectures are developed by integrating techniques like wavelet decomposition Ajder et al. (2025). Secondly, linear models represented by DLinear Zeng et al. (2022), and

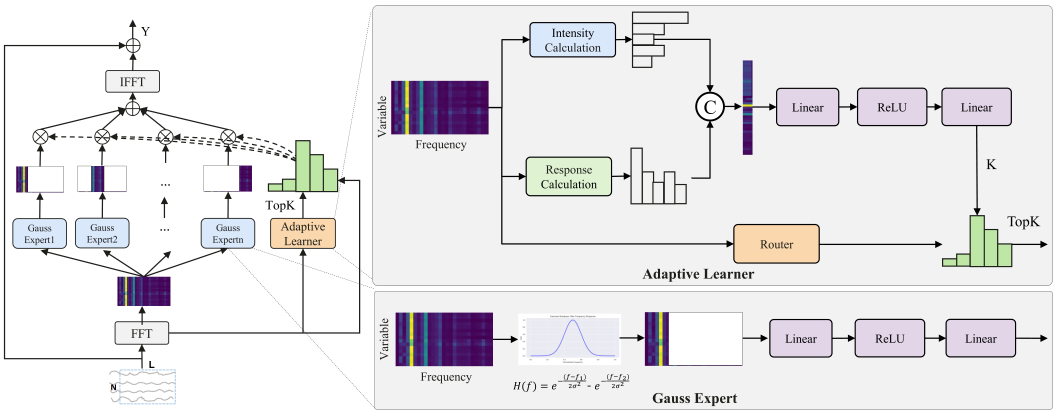


Figure 2: Overview of Ada-MoF. Gaussian band-pass filters decouple the spectrum into expert-specific bands. The adaptive learner fuses frequency- and variable-wise descriptors to predict a sample-specific expert budget and routing scores. A soft–hard router uses a differentiable soft mask for optimization and an exact hard Top-K mask for sparse execution.

RLinear Kim et al. (2023) directly perform regression on historical sequences using a single layer or very shallow MLP. They have achieved advanced performance on long sequence benchmarks due to their small parameter size, fast training, and stability over long windows. Thirdly, Transformers capture dependencies of arbitrary distances through self-attention or sparse attention. Models like PatchTST Nie et al. (2023), FEDformer Zhou et al. (2022), and iTransformer Liu et al. (2024), continuously reconstruct normalization, patch division, and frequency domain attention, maintaining leading accuracy in multivariate and multi-step prediction tasks. However, their quadratic complexity and memory consumption remain bottlenecks for practical deployment. In recent years, mamba-based methods represented by S-Mamba Wang et al. (2025) and TimePro Ma et al. (2025) have begun to emerge, reducing complexity to O(L) while retaining global receptive fields, providing new scalable solutions for long-range time series modeling. But, existing methods generally rely on end-to-end training for feature extraction, failing to explicitly decouple dominant frequency components, which results in critical spectral information being overwhelmed by redundant features, thus becoming a bottleneck for further performance improvement.

2.2 Mixture of Experts in Time Series Forecasting

In recent years, the Mixture-of-Experts (MoE) paradigm has begun to migrate from the fields of NLP and CV to the domain of time series prediction, aiming to expand model capacity while maintaining low inference costs. Time-MoE Shi et al. (2025) is the first to replace the expert layer in the Transformer's Feed-Forward layer, allowing different experts to capture distinct features. However, it does not practically assign different inputs to each expert. MoFE-Time Liu et al. (2025) equips every expert with parallel FFT MLP and time domain MLP branches, using sparse MoE routing to assign frequency components to the most appropriate expert. However, it merely applies the MLP to the whole spectrum without decoupling individual frequencies, so dominant harmonics stay mixed with noisy bins and are easily missed. FreqMoE Liu (2025) takes a step further by splitting the spectrum into sub-bands and routing each to a dedicated expert before SoftMoE fusion, yet the abrupt band-wise truncation introduces Gibbs-like edge oscillations when the signal is converted back to the time domain, inadvertently re-introducing interference. In summary, existing MoE models are not well-equipped to effectively address the dominant frequency suppression issue.

3 Method

3.1 Preliminaries

In time series forecasting, the dominant frequency may vary according to changes in the frequency distribution of the data. Using a fixed number of experts often leads to an imbalance in frequency domain coverage. We provide a detailed analysis of this issue.

A time–domain signal $x(t)$ can be characterized in the frequency domain through its energy spectral density $E(f)$. According to Parseval's relation, the total energy of the signal is preserved under the Fourier transform and satisfies

$$\int_{-\infty}^{\infty} E(f)\, df = E_{\text{total}}, \tag{1}$$

where E_{total} denotes the overall energy of the sequence. For time series whose spectral content evolves over time, the effective frequency span can be expressed as

$$\Delta f(t) = f_{\max}(t) - f_{\min}(t). \tag{2}$$

In a conventional Mixture-of-Experts (MoE) architecture with a fixed number of experts K, the model implicitly partitions the frequency axis into K subregions, assigning roughly $\Delta f(t)/K$ of the spectrum to each expert. Let w represent the characteristic bandwidth that a single expert can reliably model. To cover the entire spectral range, the required number of experts is therefore approximately

$$K \approx \frac{\Delta f(t)}{w}. \tag{3}$$

When the underlying signal exhibits frequency drift, the value of $\Delta f(t)$ may expand or contract over time. Because K is fixed, the ratio $\Delta f(t)/K$ becomes misaligned with the actual spectral structure: an enlarged $\Delta f(t)$ forces each expert to span a wider region, leading to insufficient spectral resolution, whereas a reduced $\Delta f(t)$ causes multiple experts to overlap on the narrow band, creating redundancy and unstable routing. This mismatch naturally results in uneven spectral coverage and unreliable gating behavior, revealing a fundamental limitation of fixed-expert MoE models in nonstationary frequency environments.

To address this issue, we propose an adaptive Mixture of Frequency-domain Experts model called Ada-MoF. It integrates univariate spectral intensity and multi-variable spectral average to adaptively determine the number of experts, ensuring that the number of experts matches the frequency distribution of the input data. This approach avoids information loss due to too few experts and noise introduction due to too many experts.

3.2 Adaptive Expert Selection

To adaptively select the expert that dominates the frequency and suppresses noise, we propose an adaptive expert selection mechanism. It captures both the dominant frequency and key features by simultaneously capturing the univariate spectral intensity and the cross-variable average frequency response, achieving a dual-drive frequency-variable adaptive expert selection.

First, the Fast Fourier Transform (FFT) is employed to convert time-domain features into the frequency domain. Subsequently, to identify the dominant frequencies, we first perform cross-variable averaging at each frequency. Specifically, we sum the magnitudes of the Fourier coefficients across all V channels at the same frequency f, and then divide by V to obtain the cross-variable averaged frequency response $\mu(f)$. This $\mu(f)$ reflects the overall response intensity of the system at each frequency point, aiding in the identification of the system's dominant frequencies. When $\mu(f)$ at a certain frequency is significantly higher than its neighboring frequencies, it indicates that the frequency component is highly reproducible across different variables, likely corresponding to the system's inherent period or external forcing signal. Conversely, if $\mu(f)$ is at a low level without significant peaks, it suggests that the frequency is dispersed across channels, with the energy source primarily being random noise or measurement error, thus having limited value for subsequent predictions. Therefore, $\mu(f)$ provides a global clue as to which frequencies are truly important. The formula for calculating the cross-variable averaged frequency response is as follows:

$$\mu(f) = \frac{1}{V} \sum_{v=1}^{V} |X_v(f)|, \tag{4}$$

where V denotes the number of variables and $X_v(f)$ the complex FFT result of variable v at frequency f. The vector $\boldsymbol{\mu} \in \mathbb{R}^F$ will be referred to as the frequency response vector.

Furthermore, to identify key variables, the average spectral intensity $E(v)$ of each variable is computed. Specifically, we sum the amplitudes of all frequency components of variable v and divide

by the total number of frequencies F. The value $E(v)$ reflects the average spectral intensity of the variable across the entire frequency domain and serves as a quick indicator of its activity level. A larger $E(v)$ typically suggests the presence of strong seasonal components or high-frequency variations, whereas a smaller $E(v)$ indicates that the sequence tends to be stationary or strongly damped. The spectral intensity is calculated as:

$$E_v = \frac{1}{F} \sum_{f=0}^{F-1} |X_v(f)|, \tag{5}$$

where F is the number of frequency bins and $X_v(f)$ denotes the complex FFT coefficient of variable v at frequency f. The vector $\mathbf{E}$ is referred to as the spectral intensity vector.

By concatenating $E(v)$ with $\mu(f)$, the model obtains two complementary views: $\boldsymbol{\mu}$ describes where spectral energy is concentrated, whereas $\mathbf{E}$ indicates which variables exhibit substantial spectral activity. Let M be the number of candidate experts and let $K_{\min}$ and $K_{\max} \leq M$ denote the allowed budget range. For each input sample, the shared adaptive learner first computes

$$\mathbf{z} = \phi(\mathbf{W}_1 \cdot \boldsymbol{\chi}[\boldsymbol{\mu}; \mathbf{E}] + \mathbf{b}_1)\,, \qquad \widetilde{K} = K_{\min} + (K_{\max} - K_{\min}) \operatorname{sigmoid}(\mathbf{w}_K^\top \mathbf{z} + b_K), \tag{6}$$

where $\boldsymbol{\chi}(\boldsymbol{\mu}, \mathbf{E})$ denotes the concatenation of the frequency response vector $\boldsymbol{\mu} \in \mathbb{R}^F$ and the spectral intensity vector $\mathbf{E} \in \mathbb{R}^V$. $\mathbf{W}_1 \in \mathbb{R}^{H \times (F+V)}$ and $\mathbf{w}_K^\top \in \mathbb{R}^{D \times H}$ are the weight matrix of the linear layer. $\mathbf{b}_1 \in \mathbb{R}^H$ and $\mathbf{b}_K \in \mathbb{R}^D$ are bias vectors. $\phi(\cdot)$ is the ReLU activation function. $\widetilde{K}$ is a continuous, sample-specific expert budget. The integer used in the forward pass is

$$K_{\text{hard}} = \operatorname{clip}\Big(\operatorname{round}(\widetilde{K}), K_{\min}, K_{\max}\Big)\,. \tag{7}$$

The learner also produces an importance score for every candidate expert,

$$\mathbf{s} = \mathbf{W}_s \mathbf{z} + \mathbf{b}_s \in \mathbb{R}^M. \tag{8}$$

Directly rounding $\widetilde{K}$ and ranking $\mathbf{s}$ are non-differentiable. We therefore optimize them with a soft–hard straight-through router. During training, the rank of expert i is approximated by the differentiable soft rank

$$\widetilde{r}_i = 1 + \sum_{j \neq i} \operatorname{sigmoid}\left(\frac{s_j - s_i}{\tau_r}\right), \qquad m_i^{\text{soft}} = \operatorname{sigmoid}\left(\frac{\widetilde{K} + \frac{1}{2} - \widetilde{r}_i}{\tau_m}\right), \tag{9}$$

where τ_r and τ_m are positive temperatures. As the temperatures decrease, $\widetilde{r}_i$ approaches the discrete rank and m_i^{soft} approaches a binary Top-K decision. In parallel, exact sorting produces the hard mask

$$m_i^{\text{hard}} = \mathbb{I}[r_i(\mathbf{s}) \leq K_{\text{hard}}]\,. \tag{10}$$

We combine both masks using the straight-through estimator

$$m_i = m_i^{\text{soft}} + \big(m_i^{\text{hard}} - m_i^{\text{soft}}\big)\,.\text{detach}(), \tag{11}$$

where $\text{detach}()$ is stop-gradient. Consequently, $m_i = m_i^{\text{hard}}$ numerically in the forward pass, but its backward derivative is that of m_i^{soft}. This gives the expert scores and continuous budget gradients even when the hard membership changes.

Let h_i be the output of the i-th Gaussian frequency expert. The normalized routing weight and mixture output are

$$g_i = \frac{m_i \exp(s_i)}{\sum_{j=1}^M m_j \exp(s_j) + \epsilon}, \qquad \mathbf{h} = \sum_{i=1}^M g_i h_i. \tag{12}$$

During training, the soft surrogate provides gradients to $\widetilde{K}$ and all routing scores, while the routing value in the forward pass remains the exact hard Top-K mixture. Candidate expert outputs are retained during training to evaluate the soft surrogate gradient; at inference, the soft branch is removed and only the K_{hard} selected experts are executed. Thus, sparse computation is exact at inference. No optimal-K annotation is required: the budget predictor, router, filters, and experts are jointly optimized by the forecasting objective, so an unsuitable budget is penalized through the resulting prediction error.

Table 1: Performance comparison of models before and after integrating the Ada-MoF module.

Models		TimeMixer		TimePro		iTransfomer		PatchTST	
Metric		MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE
ETTh1	Original	0.447	0.440	0.438	0.438	0.454	0.447	0.453	0.446
	+Ada-MoF	**0.432**	**0.431**	**0.436**	**0.433**	**0.453**	**0.447**	**0.438**	**0.434**
ETTh2	Original	0.383	0.407	0.377	0.403	0.383	0.407	0.385	0.410
	+Ada-MoF	**0.373**	**0.400**	**0.374**	**0.401**	**0.378**	**0.403**	**0.371**	**0.399**
ETTm1	Original	0.381	0.395	0.391	0.400	0.407	0.410	0.396	0.406
	+Ada-MoF	**0.377**	**0.395**	**0.386**	**0.396**	**0.406**	**0.408**	**0.384**	**0.398**
ETTm2	Original	0.275	0.323	0.281	0.326	0.288	0.332	0.287	0.330
	+Ada-MoF	**0.272**	**0.321**	**0.276**	**0.321**	**0.286**	**0.330**	**0.281**	**0.327**

3.3 Gaussian Feature Decoupling

To mitigate the noise introduced by the Gibbs phenomenon inherent in standard band-pass filters, we adopt a Gaussian band-pass filter to achieve a smoother decomposition of frequency-domain features. Specifically, based on the frequency domain features obtained from the fast Fourier transform, a Gaussian band-pass filter was established for feature filtering. The Gaussian band-pass filter retains only the energy within the target passband while exponentially suppressing the out-of-band components, ensuring a smooth filtering characteristic. The boundaries of each Gaussian band-pass filter are learned through an end-to-end optimization process. This optimization process automatically reallocates the support regions of the filters. Information-rich frequency bands are directed to experts with higher activation values, while frequency bands dominated by interference or noise are directed to other experts with lower activation values. This learnable frequency partitioning strategy avoids spectral aliasing, providing each expert with a statistically independent input subspace and laying the foundation for the adaptive selection of dominant frequency bands. The equation for Gaussian band-pass filtering is as follows:

$$H(f) = \exp\left(-\frac{(f-f_1)^2}{2\sigma^2}\right) - \exp\left(-\frac{(f-f_2)^2}{2\sigma^2}\right) \tag{13}$$

where $H(f)$ denotes the frequency response of the filter at frequency f. The parameters f_1 and f_2 represent the upper and lower cutoff frequencies of the passband, respectively. The term σ is the standard deviation that controls the bandwidth of each Gaussian component. A larger σ results in a smoother transition and wider frequency coverage. By subtracting two Gaussian functions centered at f_1 and f_2, this formulation creates a bandpass effect that suppresses both low and high frequencies while preserving those within the desired range.

Besides, we design a spectrum-driven adaptive standard deviation mechanism. This method automatically determines the standard deviation σ by analyzing the energy distribution of the frequency spectrum. Specifically, we first compute the average spectral intensity and normalize it by the current center frequency. As the center frequency increases, the standard deviation decreases dynamically, achieving adaptive frequency tuning. The low-frequency band often contains the long-term periodic components of a signal, such as seasonal variations or long-term trends. A larger standard deviation can better capture these components, ensuring that the filter does not miss important periodic information. The high-frequency band often contains more noise components. A narrower filter bandwidth can more accurately separate the useful components in the signal, avoiding the misjudgment of noise as signal. Meanwhile, we employ a parameter α to adjust its magnitude, ensuring it remains within a reasonable range. Compared to manually setting σ, this method is more adaptable to different frequency band characteristics.

$$\sigma = \sigma_0 \cdot \frac{\alpha}{D_0} \cdot \frac{1}{N} \sum_{i=1}^{N} |X(f_i)|^2 \tag{14}$$

where the σ denotes the automatically determined standard deviation, which controls the bandwidth of the Gaussian bandpass filter. σ_0 is the initial standard deviation. The term D_0 represents the center frequency of the filter. α refers to the adjustment coefficient.

Table 2: Multivariate long-term forecasting results across different horizons (H ∈ {96, 192, 336, 720}) under a lookback window of L = 96. Per-row best (**red**) and second-best (blue) results are highlighted.

		TimeMixer +Ada-MoF (Ours)		TimePro (ICML'25)		FreqMoE (ArXiv'25)		SOFTS (NeurIPS'24)		TimeMixer (ICLR'24)		iTransformer (ICLR'24)		PatchTST (ICLR'23)		TimesNet (ICLR'23)	
Metric		MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE
ETTh1	96	**0.373**	**0.394**	0.375	0.398	0.382	0.404	0.381	0.399	0.375	0.400	0.386	0.405	0.394	0.406	0.384	0.402
	192	**0.422**	0.426	0.427	0.429	0.433	0.429	0.435	0.431	0.429	**0.421**	0.441	0.436	0.440	0.435	0.436	0.429
	336	**0.462**	**0.443**	0.472	0.450	0.475	0.451	0.480	0.452	0.484	0.458	0.487	0.458	0.491	0.462	0.491	0.469
	720	**0.469**	**0.462**	0.476	0.474	0.485	0.476	0.499	0.488	0.498	0.482	0.503	0.491	0.487	0.479	0.521	0.500
	Avg	**0.432**	**0.431**	0.438	0.438	0.444	0.440	0.449	0.442	0.447	0.440	0.454	0.447	0.453	0.446	0.458	0.450
ETTh2	96	**0.287**	**0.339**	0.293	0.345	0.290	0.340	0.297	0.347	0.294	0.345	0.297	0.349	0.288	0.340	0.340	0.374
	192	**0.367**	**0.390**	**0.367**	0.394	0.369	**0.390**	0.373	0.394	0.376	0.396	0.380	0.400	0.376	0.395	0.402	0.414
	336	0.411	0.429	0.419	0.431	0.411	0.427	**0.410**	**0.426**	0.423	0.436	0.428	0.432	0.440	0.451	0.452	0.452
	720	0.426	0.442	0.427	0.445	0.421	0.443	**0.411**	**0.433**	0.438	0.451	0.427	0.445	0.436	0.453	0.462	0.468
	Avg	**0.373**	**0.400**	0.377	0.403	**0.373**	**0.400**	**0.373**	**0.400**	0.383	0.407	0.383	0.407	0.385	0.410	0.414	0.427
ETTm1	96	**0.313**	**0.352**	0.326	0.364	0.319	0.357	0.325	0.361	0.320	0.357	0.334	0.368	0.329	0.365	0.338	0.375
	192	**0.358**	**0.381**	0.367	0.383	0.363	0.384	0.375	0.389	0.361	**0.381**	0.377	0.391	0.380	0.394	0.374	0.387
	336	**0.387**	**0.403**	0.402	0.409	0.393	0.404	0.405	0.412	0.390	0.404	0.426	0.420	0.400	0.410	0.410	0.411
	720	**0.449**	**0.439**	0.469	0.446	0.457	0.443	0.466	0.447	0.454	0.441	0.491	0.459	0.475	0.453	0.478	0.450
	Avg	**0.377**	**0.394**	0.391	0.400	0.383	0.397	0.393	0.403	0.381	0.395	0.407	0.410	0.396	0.406	0.400	0.406
ETTm2	96	**0.173**	**0.256**	0.178	0.260	0.176	0.259	0.180	0.261	0.175	0.258	0.180	0.264	0.184	0.264	0.187	0.267
	192	**0.235**	**0.297**	0.242	0.303	0.240	0.299	0.246	0.306	0.237	0.299	0.250	0.309	0.246	0.306	0.249	0.309
	336	**0.292**	**0.339**	0.303	0.342	0.299	0.338	0.319	0.352	0.298	0.340	0.311	0.348	0.308	0.346	0.321	0.351
	720	**0.389**	**0.393**	0.400	0.399	0.396	0.394	0.405	0.401	0.391	0.396	0.412	0.407	0.409	0.402	0.408	0.403
	Avg	**0.272**	**0.321**	0.281	0.326	0.278	0.323	0.287	0.330	0.275	0.323	0.288	0.332	0.287	0.330	0.291	0.333
ECL	96	0.153	0.244	**0.139**	0.234	0.152	0.246	0.143	**0.233**	0.153	0.247	0.148	0.240	0.164	0.251	0.168	0.272
	192	0.167	0.256	**0.156**	0.249	0.165	0.255	0.158	**0.248**	0.166	0.256	0.162	0.253	0.173	0.262	0.184	0.289
	336	0.185	0.275	**0.172**	**0.267**	0.181	0.274	0.178	0.269	0.185	0.277	0.178	0.269	0.190	0.279	0.198	0.300
	720	0.224	0.310	**0.209**	**0.299**	0.219	0.307	0.218	0.305	0.225	0.310	0.225	0.317	0.230	0.313	0.220	0.320
	Avg	0.182	0.271	**0.169**	**0.262**	0.179	0.270	0.174	0.264	0.182	0.273	0.178	0.270	0.189	0.276	0.192	0.295
Weather	96	**0.161**	0.209	0.166	**0.207**	0.168	0.215	0.166	0.208	0.162	0.209	0.174	0.214	0.176	0.217	0.172	0.220
	192	**0.206**	**0.250**	0.216	0.254	0.212	0.253	0.217	0.253	0.209	0.251	0.221	0.254	0.221	0.256	0.219	0.261
	336	**0.261**	**0.291**	0.273	0.296	0.268	**0.291**	0.282	0.300	0.265	0.293	0.278	0.296	0.275	0.296	0.280	0.306
	720	**0.341**	**0.344**	0.351	0.346	0.342	0.345	0.356	0.351	0.344	0.346	0.358	0.347	0.352	0.346	0.365	0.359
	Avg	**0.242**	**0.273**	0.251	0.276	0.247	0.276	0.255	0.278	0.245	0.275	0.258	0.278	0.256	0.279	0.259	0.287
SolarEnergy	96	**0.182**	0.258	0.196	0.237	0.221	0.266	0.200	**0.230**	0.189	0.259	0.203	0.237	0.205	0.246	0.250	0.292
	192	**0.208**	0.277	0.231	0.263	0.253	0.287	0.229	**0.253**	0.222	0.283	0.233	0.261	0.237	0.267	0.296	0.318
	336	**0.224**	0.280	0.250	0.281	0.264	0.296	0.243	**0.269**	0.231	0.292	0.248	0.273	0.250	0.276	0.319	0.330
	720	**0.217**	0.273	0.253	0.285	0.263	0.289	0.245	**0.272**	0.223	0.285	0.249	0.275	0.252	0.275	0.338	0.337
	Avg	**0.208**	0.272	0.232	0.266	0.250	0.284	0.229	**0.256**	0.216	0.280	0.233	0.262	0.236	0.266	0.301	0.319
Average		**0.298**	**0.337**	0.306	0.339	0.308	0.341	0.311	0.342	0.304	0.341	0.314	0.344	0.314	0.345	0.325	0.353
1st Count		**51**		10		4		13		2		0		0		0	

4 EXPERIMENT

4.1 MAIN RESULTS

To evaluate the effectiveness of the proposed Ada-MoF module, we integrate it into several existing models. As shown in Table 1, the integration yields consistent performance improvements by a clear reduction in both MSE and MAE metrics in most cases. For instance, on the ETTh2 dataset, PatchTST with Ada-MoF achieves an MSE of 0.371 and MAE of 0.399, outperforming its original scores of 0.385 and 0.410. Similarly, for TimeMixer on the ETTm1 dataset, the MSE decreases from 0.391 to 0.386 after integration. It is noteworthy that in the few scenarios where significant gains are not observed (e.g., iTransformer on ETTh1), the performance remains on par with the original model, indicating that the module introduces no detriment. Among all enhanced models, Ada-MoF empowers TimeMixer to achieve the most competitive performance.

Table 2 presents a performance comparison of various state-of-the-art time series forecasting models, including Ada-MoF (our proposed model), TimePro Ma et al. (2025), FreqMoE Liu (2025), SOFTS Han et al. (2024), TimeMixer Wang et al. (2024), iTransformer Liu et al. (2024), PatchTST Nie et al. (2023), and TimesNet Wu et al. (2023), across several datasets, with different forecasting horizons (96, 192, 336, and 720) and a fix lookback window 96. Ada-MoF consistently achieves the

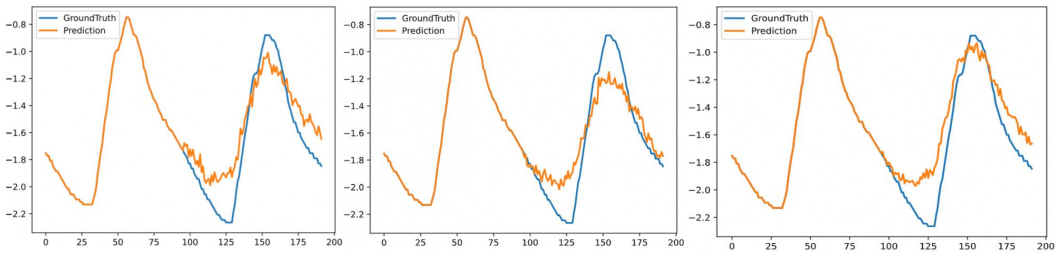


Figure 3: Comparison of 96-step Forecasts by FreqMOE, TimeMixer, and Ada-MoF on the ETTm2 Dataset. GroundTruth (blue) versus forecasts (orange).

Table 3: Contribution of individual components in Ada-MoF to forecasting performance.

Adaptive Learner	Gaussian Experts	ETTh1 MSE	ETTh1 MAE	Solar MSE	Solar MAE
×	×	0.447	0.44	0.216	0.280
×	✓	0.441 ↓**1.3%**	0.438 ↓**0.5%**	0.210 ↓**2.9%**	0.275 ↓**1.8%**
✓	✓	0.432 ↓**3.1%**	0.431 ↓**2.1%**	0.208 ↓**3.7%**	0.272 ↓**2.9%**

best performance in terms of both MSE and MAE across multiple datasets and time horizons. For example, in the ETTh1 dataset, Ada-MoF delivers the best MAE of 0.388 at the 96-step horizon, outperforming TimePro (0.394) and FreqMoE (0.399). This trend of outperforming other models is also observed across other datasets like ETTh2, ETTm1, and Weather, where Ada-MoF maintains a consistent edge in both error metrics. Notably, Ada-MoF's average MSE and MAE values at different time horizons are lower than those of the competing models. On the ETTh1 dataset, for instance, Ada-MoF achieves an average MSE of 0.432, outperforming TimePro (0.438) and FreqMoE (0.440). This superior performance remains evident at longer forecasting horizons, such as 720 steps, where Ada-MoF continues to yield more accurate predictions than models like iTransformer and PatchTST. Overall, Ada-MoF achievs 51 first-place rankings, significantly outperforming all other models for multivariate long-term time series forecasting.

To provide a more intuitive demonstration of Ada-MoF's forecasting performance, Fig. 3 presents a comparison of the 96-step forecasts from Ada-MoF, iTransformer, and TimeMixer on the ETTm2 dataset. The GroundTruth (blue) is plotted alongside the predictions (orange). While all models are able to capture the overall trend of the data, the forecasted curves of Ada-MoF are noticeably closer to the actual data, particularly at key points such as the peaks and valleys. In comparison, the predictions from FreqMOE and TimeMixer show more noticeable deviations from the GroundTruth, especially during the inflection points, where Ada-MoF maintains a more accurate fit.

4.2 Model Analysis

To evaluate the impact of different hyperparameters and the effectiveness of the structure, we carry out a comprehensive set of experiments. More extensive experiments on parameter settings, comparisons, component ablations, and plug-and-play robustness are provided in the supplementary material.

Ablation Study To verify the effectiveness of the adaptive learner and the Gaussian expert, related ablation experiments are performed. As shown in Table 3, starting from the baseline without either module, adding Gaussian experts alone reduces errors, decreasing by 1.3% on ETTh1 and 2.9% on Solar. This confirms the benefit of Gaussian feature decoupling. After FFT, learnable Gaussian bandpass filters split the spectrum into compact subbands, providing each expert with a clean frequency range and reducing aliasing, especially effective for highly seasonal data like Solar. Enabling the adaptive learner on top of Gaussian experts brings the largest improvements, down 3.1% and 14.2% from baseline. The gain stems from dual feature gating to activate the Top-K most relevant experts while suppressing noise-dominated bands, with the spectral decoupling of Gaussian experts.

Table 4: Comparison of different expert number selection methods.

Method	Exchange		Traffic	
	MSE	MAE	MSE	MAE
MLP	0.383	0.414	0.493	0.312
SE	0.385	0.415	0.496	0.321
$\boldsymbol{\mu}(f)$ + MLP	0.382	0.413	0.489	0.310
$\mathbf{E}_v$ + MLP	0.373	0.411	0.486	0.305
Dual Feature	0.363	0.402	0.482	0.297

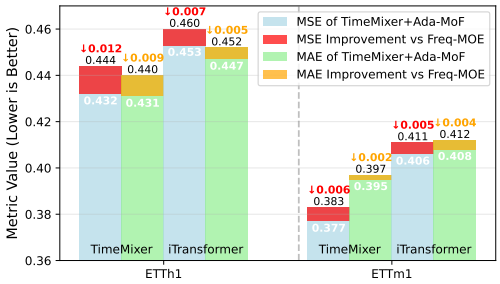


Figure 4: Performance comparison of Ada-MoF versus Freq-MOE modules.

Analysis of different expert number selection methods As shown in Table 4, it isolates the contribution of the dual-feature expert-budget predictor. Let the input x have shape (F, V), where F is the number of frequency bins and V is the number of variables. In the MLP and SE rows, $F \times V$ is compressed to 1 using an MLP or squeeze-and-excitation module, respectively. For the $\boldsymbol{\mu}(f)$+MLP row, x is averaged over variables to obtain a feature of shape $(F, 1)$, which is then compressed to 1. For the $\mathbf{E}_v$+MLP row, x is averaged over frequencies to obtain a feature of shape $(V, 1)$. For Dual Feature, the two averages are concatenated into a feature of shape $(F + V, 1)$ and compressed to 1 through an MLP. We report Exchange and Traffic results. Dual Feature is consistently optimal.

Since $E(v)$ averages spectral magnitude over all frequencies, it cannot by itself distinguish a strongly periodic variable from broadband noise with the same average magnitude. But $E(v)$ is not used alone; it is jointly processed with $\boldsymbol{\mu}(f)$. Periodic and noisy variables may share similar $E(v)$ values, yet their frequency-response patterns differ. A periodic signal yields a localized peak at one or a few frequencies, whereas broadband noise yields a dispersed, relatively flat response. The adaptive learner therefore uses $\mathbf{X} = [\boldsymbol{\mu}; \mathbf{E}]$ through a nonlinear MLP, where $E(v)$ indicates which variables contain substantial spectral activity, and $\boldsymbol{\mu}(f)$ indicates whether that activity is concentrated or dispersed.

Similarly, a very weak frequency appearing in only one or a few channels may be attenuated by averaging $\mu(f)$ over all variables. Our method prioritizes frequency components with meaningful sample-level predictive relevance, rather than arbitrarily weak isolated components; a sufficiently strong component still yields a detectable local peak after cross-variable averaging. Moreover, $\boldsymbol{\mu}(f)$ is not used alone. The variable-wise feature $\mathbf{E}$ preserves the sparse activity pattern of affected variables. When a dominant frequency occurs in only a few channels, those variables can still appear as high-activity entries in $\mathbf{E}$, while the frequency location remains represented by a local peak in $\boldsymbol{\mu}$. The adaptive learner jointly processes these complementary patterns through $\mathbf{X} = [\boldsymbol{\mu}; \mathbf{E}]$.

Comparison of between Ada-MoF and FreqMOE The comparative results on the ETTh1 and ETTm1 datasets clearly demonstrate the advantage of integrating our proposed Ada-MoF module over the Freq-MOE baseline. As shown in Fig. 4, Ada-MoF consistently achieves superior performance across both TimeMixer and iTransformer models. On the ETTm1 dataset, Ada-MoF yields a lower MSE for TimeMixer (0.377 vs. 0.383). The improvement is more pronounced on the ETTh1 dataset, where Ada-MoF attains a lower MSE in TimeMixer (0.432 vs. 0.444). These consistent gains on key benchmarks validate that Ada-MoF is a more effective enhancement for capturing temporal dependencies than the Freq-MOE module.

5 Conclusion

In this paper, we propose Ada-MoF, an adaptive expert-mixing framework, which first decomposes the full spectrum into non-overlapping sub-bands using a learnable Gaussian bandpass filter, so that each expert receives only one clean frequency slice, where the dominant band and the noise band can be decomposed. Despite this decomposition, the noise sub-band may still be activated and the prediction quality may be reduced. To suppress them, we introduce an adaptive expert selection module that fuses spectral intensity and cross-variable frequency response to learn the number of valuable experts. Across public long-term forecasting benchmarks, Ada-MoF provides competitive or leading results, achieving the best overall average MSE and MAE among the evaluated methods.

6 POLICY ON LLM USE FOR RESEARCH AND WRITING PAPERS

We adopt LLM to aid and polish the writing of this manuscript. Specifically, LLM is used to improve grammar, wording, and clarity. However, the logic and main content of the manuscript are completed by all authors.

REFERENCES

Ali Ajder, Hisham A. A. Hamza, and Ramazan Ayaz. Wavelet-enhanced hybrid lstm-xgboost model for predicting time series containing unpredictable events. *IEEE Access*, 13:58671–58679, 2025. doi: 10.1109/ACCESS.2025.3556540.

Rafal A Angryk, Petrus C Martens, Berkay Aydin, Dustin Kempton, Sushant S Mahajan, Sunitha Basodi, Azim Ahmadzadeh, Xumin Cai, Soukaina Filali Boubrahimi, Shah Muhammad Hamdi, et al. Multivariate time series dataset for space weather data analytics. *Scientific data*, 7(1):227, 2020.

Albert Gu and Tri Dao. Mamba: Linear-time sequence modeling with selective state spaces, 2024. URL https://arxiv.org/abs/2312.00752.

Lu Han, Xu-Yang Chen, Han-Jia Ye, and De-Chuan Zhan. SOFTS: Efficient multivariate time series forecasting with series-core fusion. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024. URL https://openreview.net/forum?id=89AUi5L1uA.

Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8): 1735–1780, 1997.

Jie Hu, Li Shen, and Gang Sun. Squeeze-and-excitation networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 7132–7141, 2018.

Taesung Kim, Jinhee Kim, Yun Tae, Chang Kim, Jang Park, and Jaegul Choo. Reversible instance normalization for accurate time-series forecasting against distribution shift. In *International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=rLinear2023.

Guokun Lai, Wei-Cheng Chang, Yiming Yang, and Hanxiao Liu. Modeling long- and short-term temporal patterns with deep neural networks. SIGIR '18, pp. 95–104, New York, NY, USA, 2018. Association for Computing Machinery. ISBN 9781450356572. doi: 10.1145/3209978.3210006. URL https://doi.org/10.1145/3209978.3210006.

Yiwen Liu, Chenyu Zhang, Junjie Song, Siqi Chen, Sun Yin, Zihan Wang, Lingming Zeng, Yuji Cao, and Junming Jiao. Mofe-time: mixture of frequency domain experts for time-series forecasting models. *arXiv preprint arXiv:2507.06502*, 2025.

Yong Liu, Tengge Hu, Haoran Zhang, Haixu Wu, Shiyu Wang, Lintao Ma, and Mingsheng Long. itransformer: Inverted transformers are effective for time series forecasting. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=JePfAI8fah.

Ziqi Liu. Freqmoe: Enhancing time series forecasting through frequency decomposition mixture of experts. *arXiv preprint arXiv:2501.15125*, 2025.

Xiaowen Ma, Zhenliang Ni, Shuai Xiao, and Xinghao Chen. Timepro: Efficient multivariate long-term time series forecasting with variable-and time-aware hyper-state. *arXiv preprint arXiv:2505.20774*, 2025.

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

Iliana Paliari, Aikaterini Karanikola, and Sotiris Kotsiantis. A comparison of the optimized lstm, xgboost and arima in time series forecasting. In *2021 12th International Conference on Information, Intelligence, Systems & Applications (IISA)*, pp. 1–7, 2021. doi: 10.1109/IISA52424.2021.9555520.

B. Dilip Kumar Reddy, J. Swami Naik, S. Vinay Kumar, Suresh Kumar, G. Haritha, and M. Raghavendra Reddy. A methodological review on time series forecasting by using arima. In *Proceedings of the International Conference on Advanced Materials, Manufacturing and Sustainable Development (ICAMMSD 2024)*, pp. 709–719. Atlantis Press, 2025. ISBN 978-94-6463-662-8. doi: 10.2991/978-94-6463-662-8_55. URL https://doi.org/10.2991/978-94-6463-662-8_55.

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 *International conference on learning representations*, volume 2025, pp. 34635–34667, 2025.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need, 2023. URL https://arxiv.org/abs/1706.03762.

Shiyu Wang, Haixu Wu, Xiaoming Shi, Tengge Hu, Huakun Luo, Lintao Ma, James Zhang, and Jun Zhou. Timemixer: Decomposable multiscale mixing for time series forecasting. In *International conference on learning representations*, volume 2024, pp. 38626–38652, 2024.

Zihan Wang, Fanheng Kong, Shi Feng, Ming Wang, Xiaocui Yang, Han Zhao, Daling Wang, and Yifei Zhang. Is mamba effective for time series forecasting? *Neurocomputing*, 619:129178, 2025.

Haixu Wu, Jiehui Xu, Jianmin Wang, and Mingsheng Long. Autoformer: Decomposition transformers with auto-correlation for long-term series forecasting. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan (eds.), *Advances in Neural Information Processing Systems*, volume 34, pp. 22419–22430. Curran Associates, Inc., 2021.

Haixu Wu, Tengge Hu, Yong Liu, Hang Zhou, Jianmin Wang, and Mingsheng Long. Timesnet: Temporal 2d-variation modeling for general time series analysis. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=ju_Uqw384Oq.

Ailing Zeng, Muxi Chen, Lei Zhang, and Qiang Xu. Are transformers effective for time series forecasting? In *Advances in Neural Information Processing Systems*, volume 35, pp. 1–12, 2022.

Haoyi Zhou, Shanghang Zhang, Jieqi Peng, Shuai Zhang, Jianxin Li, Hui Xiong, and Wancai Zhang. Informer: Beyond efficient transformer for long sequence time-series forecasting. *Proceedings of the AAAI Conference on Artificial Intelligence*, 35(12):11106–11115, May 2021. doi: 10.1609/aaai.v35i12.17325. URL https://ojs.aaai.org/index.php/AAAI/article/view/17325.

Tian Zhou, Ziqing Ma, Qingsong Wen, Xue Wang, Liang Sun, and Rong Jin. Fedformer: Frequency enhanced decomposed transformer for time-series forecasting. In *International Conference on Machine Learning*, pp. 27265–27277. PMLR, 2022.

A Settings

A.1 Datasets

We evaluate our method on widely-used benchmarks for long-term multivariate time-series forecasting, covering electricity load, renewable energy, and meteorology. ETTh1/ETTh2 Zhou et al. (2021) are two hourly datasets originate from transformer load and oil temperature monitoring. ETTm1/ETTm2 Zhou et al. (2021) are the minute-level counterparts of ETTh1/ETTh2, sampled every 15 minutes. ECL (Electricity Consuming Load) Wu et al. (2021) is a time series data set of power loads, which records hourly consumption from 321 clients. The Weather Angryk et al. (2020) is a real-world weather dataset. Solar-Energy Lai et al. (2018) is collected from 137 solar power plants, which provides 10-minute resolution energy production data. Unless specified otherwise, all datasets follow the standard train/validation/test splits with prediction horizons of $\{96, 192, 336, 720\}$.

ETTh1/ETTh2 Zhou et al. (2021) These two hourly datasets originate from transformer load and oil temperature monitoring. Both contain 7 variables, with moderate size and relatively weak periodic patterns. The difference between ETTh1 and ETTh2 lies in their collection points, while their statistical characteristics remain comparable. The dataset is split into 8545/2881/2881 samples for training, validation, and testing, respectively.

ETTm1/ETTm2 Zhou et al. (2021) These are the minute-level counterparts of ETTh1/ETTh2, sampled every 15 minutes. They preserve the same 7 channels but contain longer sequences, resulting in a larger training set of 34465 samples (with 11521 each for validation and testing). ETTm1 and ETTm2 come from different devices or time spans but reflect similar industrial electricity monitoring environments, with richer short-term fluctuations.

Weather This dataset consists of 21 meteorological variables (e.g., air temperature, humidity), sampled every 10 minutes across Germany in 2020. The high dimensionality and strong seasonal/daily cycles make it a challenging benchmark for capturing multi-scale temporal dependencies and cross-variable correlations. It is divided into 36792/5271/10540 samples for training, validation, and testing.

ECL (Electricity Consuming Load) Wu et al. (2021) This dataset records hourly consumption from 321 clients. With both large volume and high dimensionality, it is commonly used to test scalability and efficiency of forecasting models under high-dimensional multivariate settings. The split is 18317/2633/5261 for training, validation, and testing.

Traffic Zhou et al. (2021) This dataset describes road occupancy rates collected by sensors on San Francisco freeways, recorded hourly from 2015 to 2016. It contains 862 time series, each representing the occupancy rate of a sensor within the range [0, 1] . The data exhibits strong daily and weekly periodic patterns, making it a standard benchmark for evaluating a model's ability to capture complex spatiotemporal dependencies in high-dimensional traffic flow forecasting.

Exchange Wu et al. (2021) This dataset contains daily exchange rates of eight countries from 1990 to 2016. It is characterized by weak seasonality and high stochasticity, presenting a distinct challenge compared to other benchmarks . Effective forecasting on this dataset requires models to capture subtle mean-reverting trends rather than relying on strong periodic patterns, testing the model's robustness to non-stationary financial time series.

Solar-Energy Lai et al. (2018) Collected from 137 solar power plants, this dataset provides 10-minute resolution energy production data. It shows clear daily periodicity but is also strongly affected by irregular factors such as weather conditions, making it suitable to evaluate robustness under periodic yet volatile patterns. The dataset split is 36601/5161/10417 for training, validation, and testing.

A.2 Implementation Details

Improving TimeMixer with Ada-MoF Ada-MoF enhances TimeMixer's frequency decomposition MoE in two ways. First, we replace hard frequency masks with Gaussian bandpass filters to smoothly separate frequency bands and reduce spectral leakage. Second, we introduce a dynamic expert selection mechanism that uses a channel attention module to adaptively choose how many frequency experts to activate per input, enabling sparse and efficient inference. Together, these improvements allow more flexible and efficient multi-scale frequency processing.

Model and Hardware Configuration Ada-MoF is integrated at the end of the TimeMixer embedding stage, enabling an additional round of adaptive feature refinement prior to the Past Decomposable Mixing, which enhances the overall representational capacity of the model. We conduct a grid search over the following hyperparameters: the encoder depth $\in \{1, 2, 3, 4\}$, and the feature dimension $\in \{8, 16, 32\}$. All experiments were run on eight NVIDIA Tesla V100 GPUs. The eight V100 GPUs constituted the total experimental resource pool and were used to run independent dataset, forecasting-horizon, and hyperparameter configurations in parallel. A single Ada-MoF training run uses only one V100 GPU and does not require multi-GPU training. Under the reported configuration,

Table 5: Dataset-specific hyperparameter configurations.

Dataset	Encoder depth	Feature dimension
ETTh1	1	16
ETTh2	1	16
ETTm1	1	16
ETTm2	1	16
ECL	2	32
Weather	2	16
Solar	2	16

Table 6: Parameter and FLOP accounting before and after adding Ada-MoF.

Configuration	96		192		336		720	
	FLOPs (M)	Params. (K)	FLOPs (M)	Params. (K)	FLOPs (M)	Params. (K)	FLOPs (M)	Params. (K)
TimeMixer	397.38	79.70	469.74	97.36	578.26	123.86	867.67	194.51
TimeMixer+Ada-MoF	399.60	85.30	471.95	102.96	580.48	129.46	869.89	200.12
PatchTST	9.70	20.40	14.41	38.93	21.49	66.72	40.37	140.83
PatchTST+Ada-MoF	9.96	20.95	14.68	39.48	21.75	67.27	40.63	141.38
iTransformer	1.70	4.80	2.29	6.43	3.18	8.88	5.54	15.41
iTransformer+Ada-MoF	1.74	5.35	2.33	6.98	3.21	9.43	5.57	15.96

one run requires approximately 1.5 GB of peak GPU memory. Therefore, the eight GPUs reduced the wall-clock time of the complete experimental campaign but were not necessary for fitting or training the model.

Optimization and Metrics The models are trained with mean squared error (MSE) as the objective. During evaluation, both MSE and mean absolute error (MAE) are reported to reflect variance- and bias-related performance. We adopt the Adam optimizer combined with cosine annealing for gradual learning rate decay.

A.3 HYPERPARAMETERS

In our implementation, α is a fixed scaling coefficient. We use $\alpha = 0.25$ as a moderate empirical value to prevent the Gaussian windows from becoming excessively narrow or broad. Other hyperparameters, including feature dimension and encoder depth, are tuned on a per-dataset basis using a grid search. The selected configurations are listed in Table 5.

A.4 PARAMETER AND FLOP ACCOUNTING BEFORE AND AFTER ADDING ADA-MOF

Ada-MoF is presented both as a forecasting architecture and as a module integrated into existing backbones. We therefore separately report

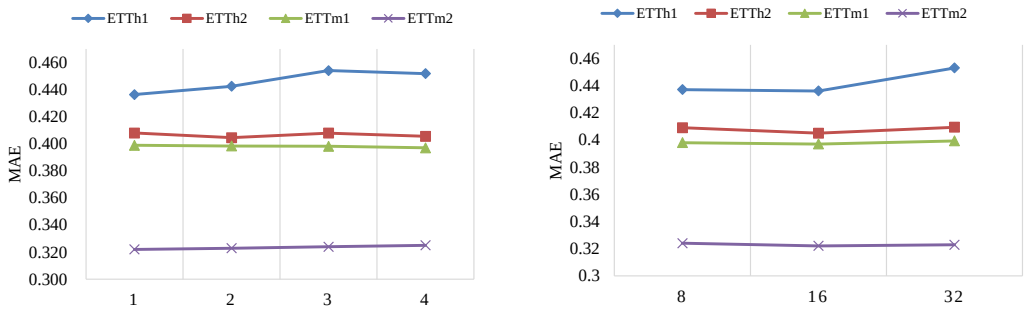
$$P_{\text{total}} = P_{\text{baseline}} + P_{\text{Ada-MoF}}, \tag{15}$$

where P_{baseline} is the number of trainable parameters in the original forecasting model, $P_{\text{Ada-MoF}}$ is the number of newly introduced trainable parameters, and P_{total} is the parameter count of the complete augmented model.

With feature dimension 16 and two encoder layers, the computational overhead is summarized in Table 6. The Ada-MoF module adds approximately 2 MFlops and 6 K parameters to TimeMixer, approximately 0.2 MFlops and 0.5 K parameters to PatchTST, and approximately 0.04 MFlops and 0.5 K parameters to iTransformer.

Table 7: Additional comparisons on the Exchange and Traffic datasets.

Dataset	TimeMixer+Ada-MoF		TimeMixer		PatchTST+Ada-MoF		PatchTST		Time-MoE		MoFE-Time	
	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE
Exchange	0.363	0.402	0.391	0.453	0.367	0.404	0.403	0.404	0.453	0.412	0.477	0.449
Traffic	0.482	0.297	0.484	0.297	0.476	0.302	0.481	0.304	0.517	0.384	0.502	0.370



(a) Layers of Encoder

(b) Dimension of Features

Figure 5: Hyperparameter sensitivity analysis of Ada-MoF on ETT datasets.

B More Comparative Results

B.1 Performance comparison of models before and after integrating the Ada-MoF module.

we conduct experiments on the Exchange and Traffic datasets and added Time-MoE and MoFE-Time as baselines. Ada-MoF is designed as a plug-and-play enhancement component that can be integrated into existing models. We applied it to TimeMixer and PatchTST and observed consistent gains on both datasets. The official implementation of TimeMixer++ was not publicly available at the time of the experiments, so we do not include it at this stage. The additional results are shown in Table 7.

C More ablation study

C.1 Analysis of Layers of Encoder

As shown in Fig. 5, depth brings limited gains because the frequency-domain decoupling already concentrates predictive energy into clean, narrow bands handled per expert. ETTh1 is best at 1 layer (0.436) and degrades when stacked deeper (0.454 at 3 layers, 0.452 at 4 layers). ETTh2 favors 2 layers (0.404), with deeper settings offering no improvement. ETTm1 changes marginally and is best at 4 layers (0.397), while ETTm2 is best at 1 layer (0.322) and worsens slightly as depth increases. These results suggest that 1–2 layers are generally sufficient: adding depth can re-mix already purified sub-band features, causing over-smoothing or optimization noise with little benefit.

C.2 Analysis of Dimension of Features

As shown in Fig. 5, a feature dimension of 16 yields optimal performance, with metrics plateauing or degrading at lower (8) or higher (32) values. Specifically, this setting yields MAEs of 0.436 on ETTh1, 0.405 on ETTh2, 0.397 on ETTm1, and 0.322 on ETTm2. A moderate dimension of 16 is sufficient to encode $\mu(f)$ (Eq. 4) and $E(v)$ (Eq. 5), whereas an 8-dimensional space is too limited, causing underfitting, and a 32-dimensional space introduces redundant parameters and estimation noise that undermine gating confidence. Therefore, 16 dimensions achieves the optimal balance between parameter efficiency and model generalization.

Table 8: Sensitivity to the maximum number of experts (MAE).

Dataset	5	6	7	8	9	10
ETTh1	0.446	0.439	0.436	0.450	0.443	0.445
ETTh2	0.412	0.410	0.405	0.409	0.404	0.419
ETTm1	0.399	0.400	0.401	0.399	0.397	0.397
ETTm2	0.323	0.323	0.323	0.322	0.323	0.322

Table 9: Ablation of expert-budget prediction features.

	ETTh1		Solar	
Method	MSE	MAE	MSE	MAE
Baseline	0.447	0.440	0.242	0.296
MLP	0.438	0.432	0.234	0.289
SE Hu et al. (2018)	0.442	0.436	0.229	0.282
Dual Feature(Ours)	**0.432**	**0.431**	**0.208**	**0.272**

C.3 ABLATION ON THE NUMBER OF EXPERTS

We analyze the sensitivity analysis on the maximum number of experts. Increasing the expert count from 5 to 10 exhibits a moderate-is-best pattern, while the overall performance remains largely stable with only modest variation. ETTh1 achieves its lowest MAE (0.436) with 7 experts, ETTh2 with 9 experts (0.404), ETTm1 stabilizes at 0.397 for 9–10 experts, and ETTm2 shows little variation (MAE 0.322–0.323). These results are consistent with the principle of Ada-MoF: Gaussian band-pass filtering allocates spectrally compact sub-bands to independent experts, while dual-dimensional adaptive gating activates only the most informative ones. Too few experts fail to cover the dominant frequency range; too many experts introduce noise-dominated sub-bands and intensify gating competition. Thus, 7–9 experts provide a good balance between frequency coverage and noise suppression. The detailed results are shown in Table 8.

C.4 ABLATION OF EXPERT-BUDGET PREDICTION FEATURES

As shown in Table 9, with the rest of the pipeline fixed, we compare different expert-number prediction features under the same soft–hard routing rule. A simple MLP gate provides modest gains over the baseline (ETTh1: 0.438 MSE, 0.432 MAE; Solar: 0.234 MSE, 0.289 MAE), though it lacks explicit spectral guidance. Channel Attention improves variable weighting and performs better on Solar (ETTh1: 0.442 MSE, 0.436 MAE; Solar: 0.229 MSE, 0.282 MAE), yet it cannot identify dominant frequency bands. Joint frequency-aware and variable-aware gating yields the best results on both datasets (ETTh1: 0.432 MSE, 0.431 MAE; Solar: 0.208 MSE, 0.272 MAE), demonstrating that allocating expert capacity according to both predictive subbands and active variables is essential.

C.5 EXPERIENCES ACROSS RANDOM SEEDS

We repeat the experiments using five independent random seeds. For each seed, the data split, model-selection protocol, and hyperparameters were kept unchanged. We report the mean and standard

Table 10: Five-seed robustness results for TimeMixer+Ada-MoF.

Dataset	MSE	MAE
ETTh1	0.434 ± 0.003	0.432 ± 0.003
ETTh2	0.374 ± 0.002	0.400 ± 0.001
ETTm1	0.378 ± 0.003	0.396 ± 0.003
ETTm2	0.275 ± 0.004	0.325 ± 0.005
ECL	0.185 ± 0.005	0.274 ± 0.006
Weather	0.243 ± 0.004	0.275 ± 0.003
Solar	0.211 ± 0.002	0.275 ± 0.004

deviation of test MSE and MAE across the five runs. The fluctuations are within an acceptable range and remain relatively stable, as shown in Table 10.