electronics

MDPI

Article

FreqMoE: Robust Time Series Forecasting via Frequency-Domain Mixture of Experts for Out-of-Distribution Scenarios

Aodong Shen [1,2,*], Zesheng Lai [2] and Tianwei Wang [3]

[1] School of Big Data and Information Engineering, Xinjiang University of Technology, Hotan 848000, China
[2] School of Computer Science and Engineering, Southeast University, Nanjing 211102, China; 220232285@seu.edu.cn
[3] McKelvey School of Engineering, Washington University in St. Louis, St. Louis, MO 63130, USA; tianwei@wustl.edu
* Correspondence: shen.list@seu.edu.cn

Abstract

Time series forecasting plays a fundamental role across diverse domains including energy systems, transportation, healthcare, etc. Despite significant advancements in forecasting models, their practical application often encounters challenges due to distribution shifts, where the statistical properties of testing data deviate from those of training data. Such shifts can severely degrade model performance, necessitating frequent retraining with the latest data to maintain performance. To address this limitation, we introduce FreqMoE, a framework designed for out-of-distribution (OOD) scenarios. Our proposed method applies random masking as a regularization strategy and utilizes a Mixture of Experts (MoE) network to encourage robust frequency-domain representations. Each expert focuses on modeling specific frequency characteristics and the MoE dynamically selects experts based on spectrum embedding, capturing frequency-domain patterns by modeling sequences in the frequency domain. Experimental results on diverse real-world datasets demonstrate that FreqMoE achieves competitive or leading performance compared to existing approaches in both prediction accuracy and robustness under OOD scenarios, demonstrating improved robustness under naturalistic temporal distribution shifts.

Keywords: deep learning; time series forecasting; out of distribution; mixture of experts

1. Introduction

Time series forecasting plays a crucial role across numerous domains, including financial markets [1,2], transportation systems [3,4], energy management [5], and so on. With the rapid advancement of deep learning techniques in recent years, many innovative deep learning models have been proposed [6,7], demonstrating remarkable performance in this field. These models leverage complex nonlinear transformations and automated feature extraction mechanisms to effectively capture intricate temporal patterns and underlying dependencies inherent in time series data. However, real-world time series data often exhibit non-stationarity [8], wherein the statistical properties of the data evolve over time, reflecting dynamically changing patterns. Such non-stationarity presents formidable challenges for accurate forecasting. Existing models experience substantial performance deterioration when applied to data that deviates from their training distribution, a condition known as out of distribution (OOD). This performance decline primarily stems from the models' strong reliance on the training data distribution, limiting their ability to adapt to distributional changes or domain shifts in test sets. As shown in Figure 1,

Check for updates

Academic Editor: Sonia Leva

Received: 16 May 2026
Revised: 17 June 2026
Accepted: 24 June 2026
Published: 1 July 2026

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

the left part is regular periodic fluctuations, such as daily electricity consumption cycles driven by predictable human activity. In contrast, the sequences in the right part display different behaviors, such as sudden shifts in electricity demand caused by unexpected weather events or system anomalies, leading to noticeable shifts in the data distribution. The challenges of model generalization and resilience become particularly evident when confronted with significant distributional changes caused by unexpected events, seasonal variations, or regulatory modifications.

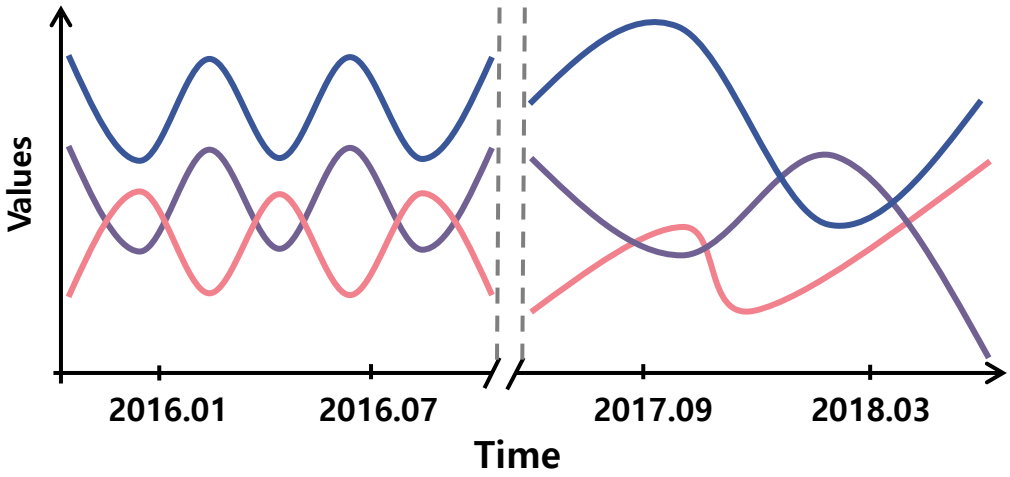


Figure 1. OOD scenarios in time series.

A commonly adopted approach to addressing the OOD problem described above is to regularize the input data. These methods [8,9] often employ statistical techniques or learnable modules to normalize input subsequences, transforming them into a stable distribution. This process mitigates non-stationary effects, enabling the model to more effectively capture underlying patterns and trends within the data. After completing the prediction process, the outputs are subsequently inverse normalized to restore them to the original scale and distribution, ensuring consistency with the input data.

Although normalization techniques are effective in alleviating the OOD problem, they exhibit certain limitations in capturing evolving patterns. Specifically, the uniform scaling of input series may overlook the variations present in different time periods. Moreover, prior studies [10] have highlighted that time series data can be viewed as a composition of multiple sequences with distinct periodicities. Each periodic component reflects unique frequency characteristics, and their superposition along the time axis creates complex patterns of changes. Beneath the complex patterns of time series lie inherent features that are intrinsically tied to the core patterns of the data and remain relatively stable over time. These stable features often encapsulate the underlying structure or fundamental patterns governing the series. Their existence enables the time series to retain a degree of internal consistency despite fluctuations, forming the foundation upon which forecasting models can capture long-term trends and patterns effectively. Modeling time series solely in the time domain can lead to difficulties in disentangling frequency components and extracting these stable features embedded in the data, which affects the forecasting performance and model generalization capability.

To address the above challenges, we introduce **FreqMoE**, an approach that analyzes time series in the frequency domain to effectively decompose and identify distinct frequency components, enabling better capture of inherent patterns. Our method first implements a low-pass filter to eliminate high-frequency noise while preserving essential low-frequency characteristics of the time series. Subsequently, we generate multiple variant sequences by

applying random masking perturbations to the original sequence. This masking strategy encourages the model to learn representations less sensitive to temporal distribution shifts by selectively obscuring temporal information, effectively simulating data incompleteness and variations. These masked variants are then processed through a Mixture of Experts (MoE) network, where each masked variant sequence is exclusively handled by a single expert in the frequency domain, and the final prediction is computed through a weighted combination of expert outputs, with weights determined by a dynamic routing mechanism. Through this architectural design, FreqMoE encourages the learning of features that are more robust to temporal distribution shifts, improving model robustness and generalization capabilities in OOD scenarios. In summary, our contributions are as follows:

- We present FreqMoE, a framework that integrates frequency-domain modeling, random masking, and Mixture of Experts routing to address OOD challenges in time series forecasting. While each component has been studied individually, their specific combination with spectrum-embedding-based dynamic routing constitutes the core contribution.
- We introduce a masking mechanism within FreqMoE to encourage robust frequency-domain representations. Meanwhile, the spectrum embedding of the input sequences enables adaptive weight assignment to the expert networks, helping the model capture stable frequency-domain patterns.
- We systematically evaluate FreqMoE on seven real-world time series datasets spanning multiple domains. Experimental results demonstrate that FreqMoE achieves competitive or leading performance compared to existing models in OOD scenarios, with the lowest performance degradation under distribution shifts.

Table 1 summarizes the positioning of FreqMoE relative to existing approaches, highlighting the integration of frequency-domain modeling, OOD robustness, and MoE architecture.

Table 1. Positioning of FreqMoE relative to existing approaches. ✓ indicates the method possesses the corresponding capability; × indicates it does not.

Method	Freq. Domain	OOD Robustness	MoE
FEDformer [11]	✓	×	×
FiLM [12]	✓	×	×
FreTS [13]	✓	×	×
Koopa [14]	✓	×	×
RevIN [9]	×	✓	×
Stationary [8]	×	✓	×
Diversify [15]	×	✓	×
Time-MoE [16]	×	×	✓
Moirai-MoE [17]	×	×	✓
FreqMoE (Ours)	✓	✓	✓

2. Related Work

2.1. Frequency-Domain Analysis Methods

Recent advancements in time series forecasting have highlighted the growing significance of frequency-domain analysis methods [13,18]. FEDformer [11] integrates seasonal-trend decomposition with Fourier transformation, implementing transformer architecture in the frequency domain to capture comprehensive temporal patterns. FiLM [12] enhances long-term forecasting accuracy by combining Legendre polynomial projections for historical information representation with Fourier projections for noise reduction. Fredformer [19] leverages discrete Fourier transformation for time series decomposition and reconstruction, incorporating local frequency-independent modeling and global semantic frequency

summarization to enhance prediction precision. Koopa [14] employs Fourier filtering to distinguish between time-varying and time-invariant components, implementing a hierarchical Koopman predictor module to characterize temporal dynamics. While these approaches have contributed to improvements in time series prediction, they still face challenges in adapting to distribution shifts and handling complex noise effectively.

2.2. Out-of-Distribution Analysis Methods

The out-of-distribution (OOD) phenomenon has been observed across various time series applications [20], gaining increased attention with the advancement in deep learning methodologies. Revin [9] and Stationary [8] focus on mitigating non-stationarity through normalization techniques and their inverse operations. AdaRNN [21] enhances model performance on non-stationary time series by maximizing distributional divergence across temporal intervals while maintaining consistency in hidden state distributions. Stone [22] improves generalization in spatio-temporal graph tasks by extracting invariant node representations. Diversify [15] employs an iterative approach to identify underlying invariant representations within time series data. However, these approaches predominantly rely on time-domain modeling, limiting their ability to effectively distinguish and capture characteristics of distinct frequency components within the data.

2.3. Mixture of Experts Models

The effectiveness of Mixture of Experts (MoE) architecture has been demonstrated across various domains, including time series forecasting. Time-MoE [16] implements a sparse MoE design that maintains high model capacity while reducing computational costs by selectively activating network components during prediction. Moirai-MoE [17] incorporates specialized expert networks to handle diverse temporal patterns and implements an cluster-based gating mechanism. MoLE [23] enhances existing linear-focused Long-Term Series Forecasting (LTSF) models by integrating multiple linear expert models with a routing mechanism. TimeMixer [6] proposes decomposable multiscale mixing for time series forecasting by leveraging temporal decomposition at different scales. Additionally, Pan et al. [24] propose a hybrid framework for traffic flow estimation combining Markovian models with deep learning. However, these MoE-based approaches often overlook the characteristics of different frequency components within time series data, resulting in reduced generalization capabilities when encountering OOD problems in real-world applications.

Table 2 summarizes the relative merits and limitations of the related methods.

Table 2. Comparison of related methods: merits and limitations.

Category	Key Merit	Key Limitation
Freq. Domain (FEDformer, FiLM, FreTS, Koopa)	Captures periodicity; reduces noise via spectral filtering	Limited robustness under distribution shifts
OOD Methods (RevIN, Stationary, AdaRNN, Diversify)	Normalization or invariant representation learning for OOD	Predominantly time domain; may miss spectral features
MoE Models (Time-MoE, Moirai-MoE, MoLE)	Adaptive capacity; sparse activation for efficiency	Overlook frequency component characteristics
FreqMoE	Integrates freq. domain, OOD robustness, MoE routing	No theoretical invariance guarantees

3. Preliminary Section

The Fourier transform (FT) is a fundamental mathematical tool for converting signals from the time domain to the frequency domain, enabling comprehensive analysis of signal frequency components. This transformation decomposes time series data into constituent sinusoidal waves of varying frequencies and amplitudes, thereby revealing essential characteristics including periodicity, underlying trends, and noise patterns within the sequence. Given a discrete time series $x(t) = [x_1, x_2, \cdots, x_T] \in \mathbb{R}^{1\times T}$, the discrete Fourier transform (DFT) is mathematically expressed as

$$X(f) = \sum_{t=0}^{N-1} x(t) e^{\frac{-j2\pi ft}{N}} \tag{1}$$

where $X(f)$ denotes the complex spectrum in the frequency domain, f denotes the frequency index, N is the total number of sampling points, and j is an imaginary unit satisfying $j^2 = -1$. The exponential term $e^{\frac{-j2\pi ft}{N}}$ in Equation (1) can be expanded into a combination of cosine and sine by Euler's formula:

$$e^{\frac{-j2\pi ft}{N}} = \cos(\frac{2\pi ft}{N}) - j\sin(\frac{2\pi ft}{N}) \tag{2}$$

After substituting Equation (2) into the Fourier transform equation, $X(f)$ can be decomposed into real and imaginary expressions:

$$X(f) = \mathrm{Re}[X(f)] + j \cdot \mathrm{Im}[X(f)] \tag{3}$$

where $\mathrm{Re}[X(f)]$ denotes the cosine component of the frequency f, reflecting the fundamental amplitude contribution of the signal at that frequency, and $\mathrm{Im}[X(f)]$ represents the sinusoidal component of the frequency f and captures the phase shift information of the signal. The complex spectrum $X(f)$ obtained by Fourier transform not only provides information about the frequency distribution but also allows further calculation of amplitude and phase, which can be used to analyze the periodicity and amplitude characteristics of the time series:

$$A(f) = \sqrt{(\mathrm{Re}[X(f)])^2 + (\mathrm{Im}[X(f)])^2} \tag{4}$$

$$\phi(f) = \arctan\left(\frac{\mathrm{Im}[X(f)]}{\mathrm{Re}[X(f)]}\right) \tag{5}$$

where $A(f)$ is the amplitude spectrum, which measures the energy distribution of the signal at different frequencies, and $\phi(f)$ is the phase spectrum, which describes the phase shifts of the frequency components.

4. Methodology

The proposed FreqMoE framework comprises three components: Temporal Smoothing Module, Masking Mechanism, and Frequency-Domain Mixture of Experts network. These integrated components work synergistically to enhance modeling and prediction capabilities for complex time series in OOD scenarios. The comprehensive architecture of FreqMoE is depicted in Figure 2.

4.1. Temporal Smoothing Module

Time series collected from real-world environments are often affected by noise disturbances and unexpected events, which typically appear as outliers that deviate markedly from the overall trend. Such outliers can severely distort the model's ability to predict future

patterns, thereby undermining its robustness and generalization performance. To mitigate the influence of noise and outliers, we firstly smooth the input time sequences:

$$\bar{x}_t = \frac{1}{w} \sum_{i=-w/2}^{w/2} x_{t-i} \tag{6}$$

where w denotes the size of the moving window, and x_t represents the input sequence at time t. The moving average computes the local mean within a sliding window, effectively reducing sharp local fluctuations and high-frequency noise while preserving the primary trend and low-frequency components of the series. This smoothing operation can be viewed as a low-pass filter (LPF), equivalent to applying a window filter defined as $h_t = \frac{1}{w}$ to the input sequence, where $-\frac{w}{2} \leq t \leq \frac{w}{2}$.

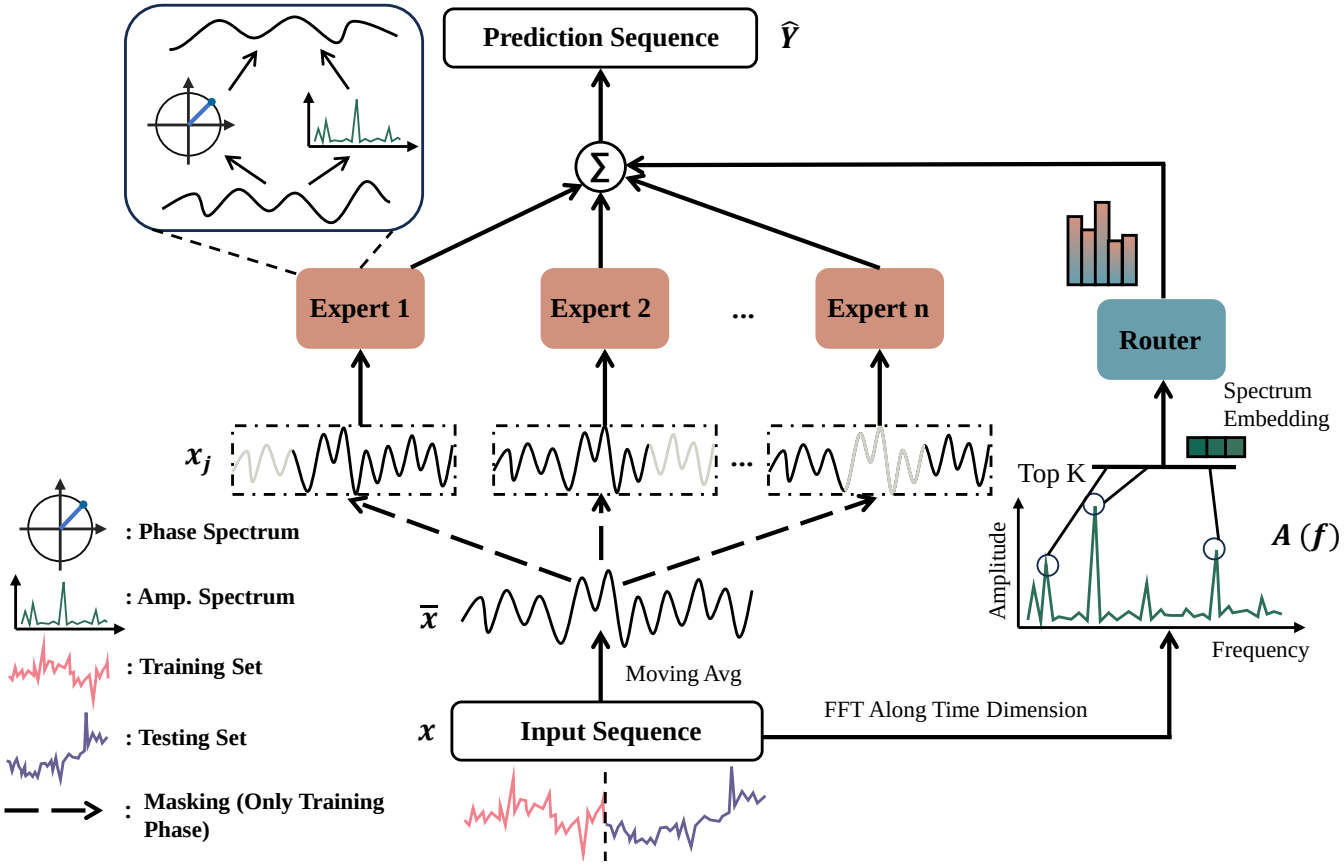


Figure 2. Framework of FreqMoE. The input sequence is filtered for noise via a moving average, followed by masking to encourage robust frequency-domain representations. Finally, the sequences are modeled by frequency-domain MoE to learn robust frequency-domain representations for prediction in OOD scenarios. Masking (dashed border) is applied only during training and disabled during inference. Solid arrows indicate data flow during both training and inference; dashed borders indicate operations applied only during training.

4.2. Masking Mechanism

After obtaining the smoothed series $\bar{x} \in \mathbb{R}^{H \times C}$ with H length of historical sequence and C feature variables, we introduce a masking mechanism to simulate data incompleteness. Random masking is applied independently to each channel at the point level: each time point in each channel is independently masked with probability r. This approach generates a set of masked variants, expressed as

$$\{x_j\}_{j=1}^{N} = \text{Mask}_r(\bar{x}) \tag{7}$$

where r represents the masking ratio, controlled within $[0,1]$, $x_j \in \mathbb{R}^{H\times C}$ denotes the j-th masked time series, and N is the number of masked variants produced. Random masking operations can mimic outlier gaps caused by missing sensor readings, sampling errors, or abrupt events. Incorporating these perturbations during training encourages the model to focus on underlying patterns and key features in the sequence, thereby enhancing its ability to identify stable spectral patterns and improving robustness against incomplete data.

4.3. Frequency-Domain MoE

After generating the masked variant sequences x_j, each sequence is independently processed by an expert. To further enhance FreqMoE's ability to capture robust frequency-domain features, we use the spectrum embedding of the input sequences to dynamically assign weights to each expert. A weighted summation of the outputs from all experts is performed to generate the final prediction results.

4.3.1. Expert Network

The structure of the expert network is shown in Figure 3. Given an input sequence x_j (in the training phase, x_j denotes the masked variant sequence, while in the testing phase, x_j denotes the smoothed sequence $\bar{x}$ and no masking operation is performed), we first transform the sequence from the time domain to the frequency domain using the FFT to obtain its frequency representation $X_j(f)$. The amplitude $A_j(f)$ and phase $\phi_j(f)$ of the sequence in the frequency domain are then calculated based on Equations (4) and (5), respectively. To enable effective feature extraction and accurate modeling of future sequences, two learnable feed-forward networks (FFNs) are introduced to independently process the amplitude and phase components:

$$\hat{A}_j(f), \hat{\phi}_j(f) = \mathrm{FFN}_A(A_j(f)), \mathrm{FFN}_\phi(\phi_j(f)) \tag{8}$$

where $\hat{A}_j(f)$ and $\hat{\phi}_j(f)$ denote the predicted amplitude and phase components for the future sequences, respectively. Leveraging Euler's formula, we integrate the transformed amplitude and phase components to reconstruct the predicted representation in the frequency domain. The reconstructed frequency-domain signal is then converted back to the time domain using the inverse fast Fourier transform (iFFT), yielding the final output sequence. This process can be formulated as follows:

$$\begin{aligned} \hat{X}_j(f) &= \hat{A}_j(f) e^{i\hat{\phi}_j(f)} \\ \hat{Y}_j &= \mathcal{F}^{-1}(\hat{X}_j(f)) \end{aligned} \tag{9}$$

where $\mathcal{F}^{-1}$ denotes iFFT operation, $\hat{X}_j(f)$ and $\hat{Y}_j$ represent the predicted frequency-domain sequence and output sequence, respectively. We clarify the handling of different input and output lengths in the frequency domain. The input sequence of length $H = 96$ yields a frequency representation of size $\lfloor H/2 \rfloor + 1 = 49$ after FFT. The expert FFNs map from this 49-dimensional input to a target output frequency dimension of size $\lfloor T_{out}/2 \rfloor + 1$, where $T_{out} \in \{96, 192, 336, 720\}$ is the prediction length. The iFFT is then applied with the target length T_{out} to produce the time-domain prediction. Phase wrapping is naturally handled by the arctan function in the phase computation. The process enables the model to capture meaningful patterns within the sequence from a frequency-domain perspective, facilitating the generation of accurate and robust predictions in OOD scenarios.

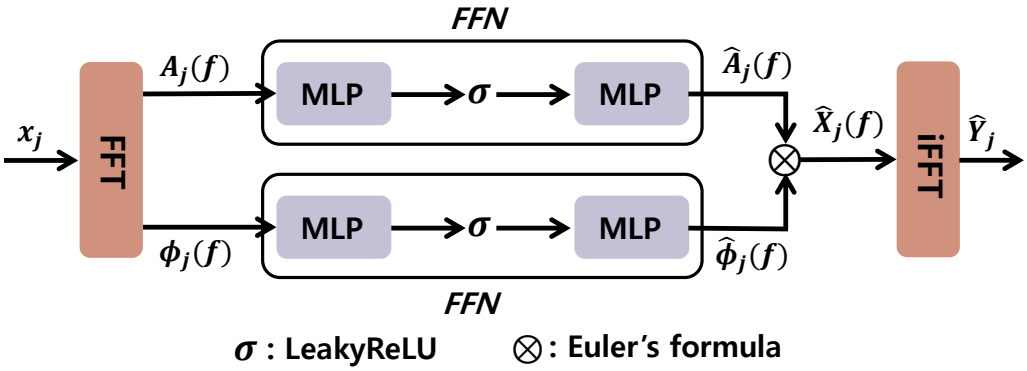


Figure 3. Expert network. Independent FFNs are designed to model the amplitude and phase components of the input sequence separately.

4.3.2. Router

To enhance FreqMoE's ability to capture robust frequency-domain features, we leverage the spectrum embedding of the input sequence to dynamically assign weights to each expert. Specifically, the input sequence x is first transformed into the frequency domain using the Fourier transform to extract its spectrum $X(f)$. The top k frequency components with the highest amplitudes are selected and ranked, forming the set of dominant frequency components:

$$\{f_1, \cdots, f_k\} = \underset{f_* \in \{1, \cdots, [\frac{T}{2}]\}}{\text{argTopk}} \ (A(f)) \tag{10}$$

where T is the length of the input sequence and $A(f)$ denotes the amplitude of the sequence at each frequency. Using the selected frequency components, we construct the vector $\mathbf{v}$ to serve as a spectrum embedding:

$$\mathbf{v} = [A(f_1), A(f_2), \ldots, A(f_k)] \tag{11}$$

where $A(f_i)$ denotes the amplitude spectrum of the input sequence at frequency f_i. We note that the top-k amplitude selection inherently encodes frequency index information, since the selected components correspond to specific frequency positions. To maintain a consistent scale for the feature vectors, we apply the softmax function to $\mathbf{v} \in \mathbb{R}^k$, resulting in the normalized vector $\hat{\mathbf{v}} \in \mathbb{R}^k$. The vector $\hat{\mathbf{v}}$ is then projected by the router weight matrix $W_r \in \mathbb{R}^{N \times k}$ and noise weight matrix $W_n \in \mathbb{R}^{N \times k}$, where N is the number of experts. The routing score vector $\mathbf{s} \in \mathbb{R}^N$ is computed as follows, where a noise term is introduced to promote balanced utilization across the experts:

$$\mathbf{s} = W_r \hat{\mathbf{v}} + \epsilon \cdot \text{Softplus}(W_n \hat{\mathbf{v}}), \quad \epsilon \sim \mathcal{N}(0, 1) \tag{12}$$

where W_n represents the learnable noise weight matrix, initialized from a positive distribution. Each element s_i of $\mathbf{s}$ denotes the routing score for the i-th expert. The noise term $\epsilon \cdot \text{Softplus}(W_n \hat{\mathbf{v}})$ is active only during trainingto encourage exploration and balanced expert utilization, and is disabled (set to zero) during inference. The routing score of the i-th expert is processed through a softmax function to derive the corresponding routing probability p_i. Finally, the prediction outputs from all experts are weighted and aggregated based on p_i to produce the final prediction results of FreqMoE:

$$p_i = \frac{\exp(s_i + b_i)}{\sum_{j=1}^{N} \exp(s_j + b_j)}$$
$$\hat{Y} = \sum_{i=1}^{N} p_i \cdot E_i(x) \tag{13}$$

where b_i and b_j are learnable biases, $E_i(x)$ is the prediction result of the i-th expert, N is the number of experts, and $\hat{Y}$ is the final prediction result. To mitigate the load balancing problem in MoE [25], an additional regularization term is incorporated into the loss function to promote balanced utilization of all experts. Specifically, we define the importance of expert i as the batch-wise sum of its routing probabilities:

$$\text{Importance}_i(x) = \sum_{b=1}^{B} p_i^{(b)} \tag{14}$$

where B is the batch size and $p_i^{(b)}$ is the routing probability for expert i on sample b. The auxiliary loss is then formulated as

$$\mathcal{L}_{\text{importance}}(x) = \lambda \cdot CV(\text{Importance}(x))^2 \tag{15}$$

where CV denotes the coefficient of variation across experts and λ is a scaling factor for the importance loss. The final loss function consists of two parts and can be expressed as the following equation:

$$\mathcal{L} = \mathcal{L}_{mse} + \mathcal{L}_{\text{importance}} \tag{16}$$

where $\mathcal{L}_{mse}$ denotes the MSE loss between the prediction and the ground truth and $\mathcal{L}_{\text{importance}}$ is used to balance the expert selection process.

4.3.3. Implementation Details

We provide the following implementation details for clarity. All expert networks share the same architecture (separate FFNs for amplitude and phase) but do not share parameters—each expert has independent learnable weights. The top-k frequency component selection is fixed at $k = 4$ for all experiments (not adaptive), selected based on the hyperparameter sensitivity analysis (Section 5.8). The relationship between the noise term and the MoE balancing loss is complementary: the noise term encourages exploration during routing in training, while the importance loss penalizes unbalanced expert utilization at the batch level.

To justify the choice of amplitude-only spectrum embedding for the router, we conduct an ablation comparing three embedding designs on the weather dataset. As shown in Table 3, amplitude-only routing achieves the best performance. Adding explicit frequency indices does not improve results, as the top-k selection already encodes frequency position information. Including phase information marginally changes the result but doubles the embedding dimension, adding complexity without a commensurate benefit. We therefore retain amplitude-only as the default.

Table 3. Ablation of router embedding design on the weather dataset (average MSE and MAE across four horizons). Best results are highlighted in **bold**.

Router Embedding	Avg MSE	Avg MAE
Amplitude only (default)	**0.533**	**0.416**
Amplitude + frequency index	0.542	0.422
Amplitude + phase	0.537	0.419

5. Experiments

5.1. Datasets

We evaluate the effectiveness of the proposed FreqMoE model using seven real-world datasets spanning multiple domains, including energy, electricity, weather, and traffic. Specifically, the datasets include ETT (ETTh1, ETTh2, ETTm1, ETTm2), weather, electricity and traffic [26,27]. To thoroughly evaluate the model's generalization performance under OOD scenarios, we design a testing protocol. For dataset partitioning, the initial 50% of the time series data was split into training and validation sets with a 7:3 ratio. The last 25% of the data was designated as the test set. The middle 25% is intentionally excluded to create a temporal gap between the validation and test periods, increasing the likelihood of distributional divergence. This partitioning strategy not only mitigates the risk of data leakage but also introduces a temporal gap between the validation and test sets, creating a more realistic and challenging OOD scenario. We acknowledge that this temporal holdout protocol simulates naturalistic temporal drift rather than controlled synthetic distribution shifts. To rigorously quantify the distributional difference, we report Wasserstein-1 distance and Maximum Mean Discrepancy (MMD) between training and test subsequences in Table 4. The results indicate that all datasets exhibit measurable distributional shifts under this protocol. Further details are provided in Table 5.

Table 4. Distributional distance metrics between training and test periods. W1 denotes Wasserstein-1 distance; MMD denotes Maximum Mean Discrepancy.

Metric	ETTh1	ETTh2	ETTm1	ETTm2	Weather	Electricity	Traffic
W1	1.995	5.892	2.256	6.572	20.121	308.941	201.003
MMD	0.087	0.250	0.108	0.258	0.066	0.045	0.025

Table 5. Dataset partitioning protocol for OOD evaluation.

Dataset	Train Span	Validation Span	Test Span
ETTh1	1 July 2016–12 March 2017	13 March 2017–28 June 2017	27 December 2017–26 June 2018
ETTh2	1 July 2016–12 March 2017	13 March 2017–28 June 2017	27 December 2017–26 June 2018
ETTm1	1 July 2016–12 March 2017	13 March 2017–28 June 2017	27 December 2017–26 June 2018
ETTm2	1 July 2016–12 March 2017	13 March 2017–28 June 2017	27 December 2017–26 June 2018
Weather	1 January 2020–8 May 2020	9 May 2020–2 July 2020	1 October 2020–31 December 2020
Electricity	1 July 2016–19 July 2017	20 July 2017–31 December 2017	1 October 2018–2 July 2019
Traffic	1 July 2016–13 March 2017	14 March 2017–1 July 2017	31 December 2017–2 July 2018

5.2. Data Analysis

We systematically examine the differences between in-distribution and out-of-distribution scenarios by analyzing data characteristics. Specifically, we extract daily subsequences from the training and test sets and evaluate them based on statistical properties and dynamic patterns. To assess statistical properties, we compute the mean difference between the subsequences to quantify overall trend shifts. For dynamic patterns, we employ Dynamic Time Warping (DTW) [28] distance to measure variations in sequence patterns. The DTW distance is calculated using the following formula:

$$DTW(x, y) = \min_{\pi} \sum_{(i,j)\in\pi} \|x_i - y_j\| \tag{17}$$

where x and y represent two distinct time series, while π denotes the alignment path. The term $\|x_i - y_j\|$ refers to the local distance between the corresponding points in the two series along the alignment path.

Figures 4 and 5 compare the DTW distance and mean difference distributions for different datasets under in-domain and out-domain scenarios. The results reveal that the electricity-domain dataset exhibits substantial dynamic variations, with a pronounced difference between the DTW distances and mean differences between in-domain and out-domain subsequences. This indicates significant dynamic changes in this domain, potentially driven by the high sensitivity of electricity data to seasonal and demand fluctuations. In contrast, the weather- and traffic-domain datasets display smoother dynamic changes, with DTW distance and mean difference distributions for in-domain and out-domain scenarios showing closer alignment. This stability may be attributed to natural patterns or infrastructure constraints that regulate trends in weather and traffic data over time, resulting in more consistent sequence behaviors.

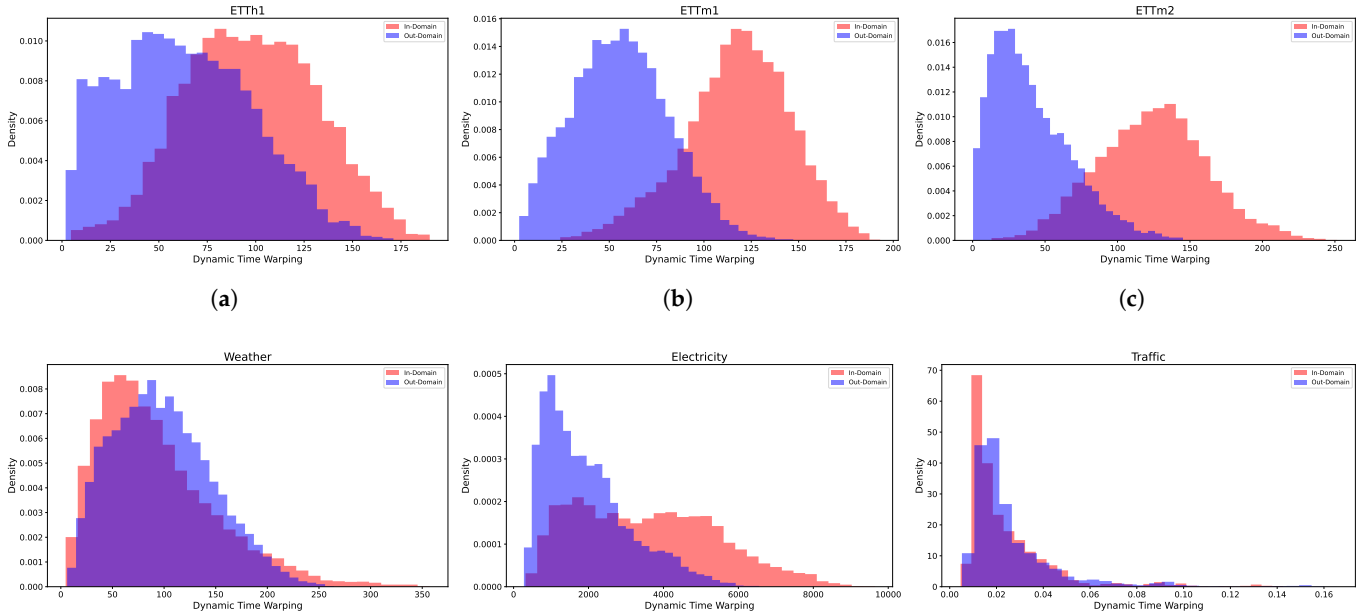


Figure 4. Comparison of DTW distances across in-domain and out-domain scenarios for various datasets. "In-domain" refers to subsequences sampled from the training period; "out-domain" refers to subsequences sampled from the test (OOD) period. (**a**) ETTh1; (**b**) ETTm1; (**c**) ETTm2; (**d**) Weather; (**e**) Electricity; (**f**) Traffic.

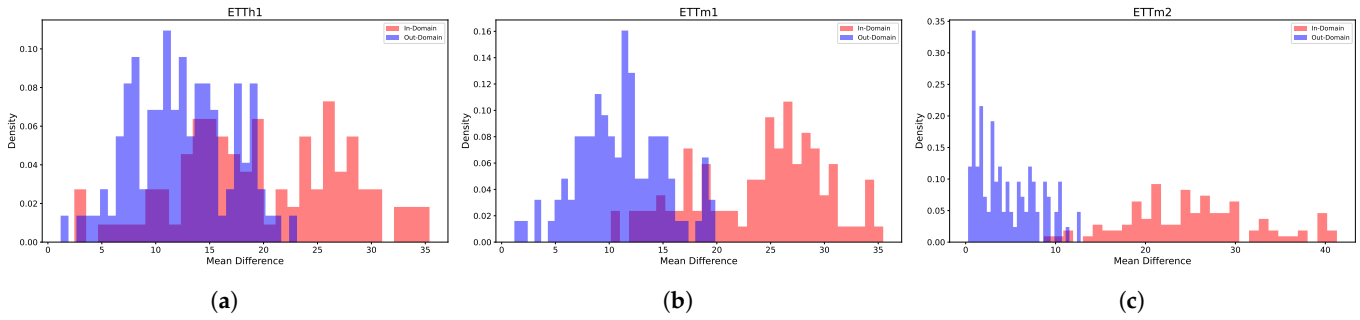
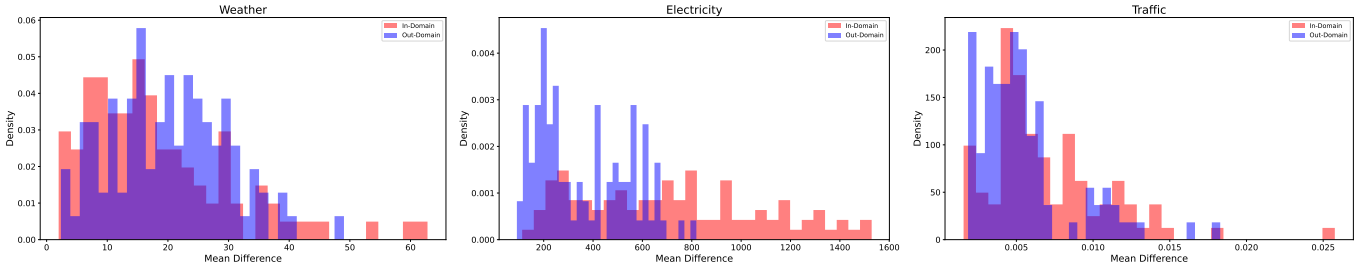


Figure 5. *Cont.*



(**d**) (**e**) (**f**)

Figure 5. Comparison of mean differences across in-domain and out-domain scenarios for various datasets. "In-domain" refers to subsequences sampled from the training period; "out-domain" refers to subsequences sampled from the test (OOD) period. (**a**) ETTh1; (**b**) ETTm1; (**c**) ETTm2; (**d**) Weather; (**e**) Electricity; (**f**) Traffic.

5.3. Baseline and Setting

Our baseline includes several state-of-the-art time series forecasting approaches, covering both general forecasting architectures and methods specifically designed for non-stationarity and OOD scenarios:

- iTransformer [29]: A transformer-based model that embeds time series as variable tokens, employs the attention mechanism to capture multivariate correlations, and significantly improves the efficiency of computation.
- PatchTST [30]: A model that segments the time series into subsequence-level patches to serve as input tokens for the transformer and employs channel-independence techniques, significantly reducing the computational and memory overhead.
- FreTS [13]: It proposed a method using an MLP in the frequency domain by transforming a time series into the frequency domain via the DFT, applying a complex linear transformation and then returning to the time domain via inverse Fourier transform to obtain predictions.
- DLinear [31]: It outperformed the transformer-based model in the benchmark by combining trend decomposition techniques with simple linear models.
- Stationary [8]: It enhances the model's predictive robustness in dynamic environments by smoothing the input series, standardizing its statistical properties, and mitigating the impact of distributional shifts.
- RevIN [9]: A normalization-based method specifically designed for non-stationary time series that applies reversible instance normalization to mitigate distribution shifts.
- Diversify [15]: An OOD-specific method that iteratively discovers invariant representations within time series data for improved generalization.
- TimesNet [32]: A recent frequency-domain forecasting model that transforms 1D temporal variations into 2D frequency-domain representations and applies 2D convolution.

To ensure fairness and consistency in evaluation, we adopt a fixed input length of $H = 96$ and evaluate output lengths $T_{out} \in \{96, 192, 336, 720\}$ across all datasets for all approaches. For long-term forecasting tasks, we employ mean squared error (MSE) and mean absolute error (MAE) as evaluation metrics. All results are reported as the mean over 3 independent runs with different random seeds (42, 2024, 3407).

5.4. Model and Training Configuration

For our experiments, the masking ratio is configured to 0.15, the sliding window length of the moving average is set to 5, the number of experts is set to 4, and the *top-k* frequency components are set to 4 in FreqMoE. Our model is trained with the L2 loss and the Adam optimizer [33]. The maximum number of training epochs is 10 and early stopping with a

patience of 3 epochs on validation MSE is applied. Z-score normalization is applied per channel. All experiments are implemented using PyTorch 2.0.1 [34] on NVIDIA RTX A6000 48 GB GPU. The source code and configuration files for FreqMoE will be publicly released upon acceptance. More detailed modeling and training configurations are presented in Table 6.

Table 6. Detailed description of model hyperparameters and training settings.

Dataset	Model Hyperparameters		Training Settings	
	Expert Num	Hidden Dim	LR	BS
ETTh1	4	256	1×10^{-3}	32
ETTh2	4	128	5×10^{-3}	32
ETTm1	4	256	1×10^{-3}	32
ETTm2	4	256	1×10^{-3}	32
Weather	4	256	1×10^{-3}	32
Electricity	4	256	1×10^{-3}	32
Traffic	4	256	1×10^{-3}	32

5.5. Main Results

The performance comparison between FreqMoE and other models is presented in Table 7. The results demonstrate that FreqMoE achieves competitive or leading performance in both MSE and MAE metrics across most prediction settings. In terms of average performance over all datasets, FreqMoE consistently ranks within the top two, further supporting its stability and effectiveness in OOD scenarios. Notably, FreqMoE achieves higher prediction accuracy with fewer parameters compared to mainstream transformer-based models, achieving better average performance than iTransformer across most evaluated datasets. On ETT datasets, DLinear remains competitive in some settings, while FreqMoE shows particularly strong results on the weather, electricity, and traffic datasets. The performance of FreqMoE can be attributed to its frequency-domain analysis combined with the masking and MoE routing mechanism, enabling the model to capture robust frequency-domain patterns from historical sequences. Additionally, the integration of a dynamic routing mechanism allows the model to adapt flexibly to varying data distributions, improving predictive accuracy and robustness against out-of-distribution data.

Statistical Significance. To assess the statistical significance of FreqMoE's performance advantages, we conducted one-sided paired t-tests using the following protocol. The metric is MSE averaged across the four prediction horizons ($T_{out} \in \{96, 192, 336, 720\}$) per dataset. Each horizon-level MSE value, computed over the full test set and averaged over 3 independent random seeds (42, 2024, 3407), serves as a paired observation. FreqMoE is compared against each baseline across the seven datasets. The win/tie/loss summary at $p < 0.05$ is presented in Table 8.

FreqMoE achieves significantly lower MSE ($p < 0.05$) than iTransformer, FreTS, Stationary, and Diversify on all seven datasets and significantly outperforms PatchTST, RevIN, and TimesNet on the majority of datasets. On ETTm1, DLinear remains competitive, consistent with the per-dataset results in Table 7.

Table 7. Full results of multivariate time series forecasting with prediction length $T_{out} \in \{96, 192, 336, 720\}$. Lower MSE and MAE represent better performance. The best results are **bolded** and the second best results are underlined.

Methods		FreqMoE (Ours)		iTrans-former		Patch-TST		FreTS		DLinear		Station-ary		RevIN		Diver-sify		Times-Net	
Metric		MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE
ETTh1	96	0.790	0.544	0.813	0.552	0.806	0.557	0.803	0.555	**0.778**	**0.539**	1.185	0.696	0.802	0.553	0.815	0.558	0.808	0.555
	192	0.880	0.588	0.892	0.600	0.904	0.604	0.905	0.593	**0.860**	**0.578**	1.349	0.750	0.891	0.598	0.908	0.607	0.897	0.601
	336	0.954	0.628	0.981	0.647	0.997	0.653	0.976	0.628	**0.925**	**0.611**	1.575	0.824	0.963	0.636	0.981	0.645	0.972	0.639
	720	**1.109**	**0.719**	1.250	0.771	1.279	0.779	1.138	0.720	1.114	0.721	1.897	0.935	1.128	0.732	1.142	0.738	1.121	0.728
	Avg	0.933	0.620	0.984	0.643	0.997	0.648	0.970	0.626	**0.919**	**0.612**	1.502	0.801	0.946	0.630	0.962	0.637	0.950	0.631
ETTh2	96	0.787	0.601	0.805	0.611	**0.770**	**0.592**	1.545	0.838	1.199	0.758	1.358	0.831	0.823	0.619	0.841	0.627	0.812	0.615
	192	**1.018**	**0.674**	1.043	0.688	1.028	0.683	2.534	1.059	1.817	0.928	1.854	0.981	1.051	0.695	1.078	0.703	1.049	0.691
	336	**1.224**	**0.737**	1.264	0.755	1.238	0.747	3.622	1.263	2.623	1.098	2.178	1.063	1.268	0.753	1.305	0.762	1.271	0.749
	720	**1.595**	**0.857**	1.616	0.864	1.626	0.856	6.490	1.710	4.776	1.459	2.501	1.158	1.641	0.871	1.692	0.884	1.648	0.867
	Avg	**1.156**	**0.717**	1.182	0.730	1.166	0.720	3.548	1.218	2.604	1.061	1.973	1.008	1.196	0.735	1.229	0.744	1.195	0.731
ETTm1	96	0.656	0.487	0.698	0.507	0.653	0.491	0.676	0.504	**0.659**	**0.483**	0.920	0.585	0.672	0.498	0.681	0.505	0.669	0.496
	192	0.752	0.522	0.793	0.537	0.763	0.528	0.767	0.530	**0.749**	**0.508**	1.034	0.622	0.768	0.533	0.779	0.541	0.764	0.530
	336	0.874	0.567	0.883	0.564	0.862	0.543	0.880	0.571	**0.854**	**0.545**	1.260	0.693	0.891	0.575	0.903	0.584	0.886	0.572
	720	0.985	0.621	1.033	0.631	0.991	**0.617**	**0.980**	**0.617**	0.985	0.623	1.550	0.807	1.003	0.632	1.018	0.641	0.998	0.629
	Avg	0.817	0.549	0.852	0.560	0.817	0.545	0.826	0.556	**0.812**	**0.540**	1.191	0.677	0.834	0.560	0.845	0.568	0.829	0.557
ETTm2	96	**0.487**	**0.485**	0.551	0.514	0.497	0.487	1.004	0.682	0.594	0.541	0.645	0.562	0.502	0.493	0.511	0.501	0.498	0.490
	192	**0.609**	**0.535**	0.677	**0.570**	0.620	0.538	2.054	0.914	0.855	0.647	1.187	0.775	0.625	0.544	0.638	0.552	0.620	0.540
	336	**0.766**	**0.593**	0.800	0.609	0.781	0.596	2.422	1.020	1.303	0.796	0.997	0.678	0.785	0.603	0.801	0.612	0.778	0.598
	720	**1.060**	**0.688**	1.121	0.718	1.073	0.695	5.454	1.513	2.362	1.052	1.189	0.768	1.089	0.698	1.107	0.709	1.078	0.694
	Avg	**0.731**	**0.575**	0.787	0.603	0.743	0.579	2.734	1.032	1.279	0.759	1.005	0.696	0.750	0.585	0.764	0.594	0.744	0.581
Weather	96	**0.339**	**0.318**	0.345	0.323	0.371	0.342	0.393	0.377	0.376	0.368	0.401	0.375	0.342	0.321	0.351	0.329	0.346	0.325
	192	**0.448**	**0.379**	0.456	0.384	0.473	0.395	0.465	0.410	0.471	0.420	0.487	0.426	0.452	0.386	0.463	0.393	0.458	0.389
	336	**0.575**	**0.441**	0.580	0.445	0.592	0.451	0.592	0.482	0.585	0.481	0.626	0.497	0.581	0.448	0.594	0.456	0.588	0.452
	720	**0.771**	**0.525**	0.786	0.530	0.790	0.532	0.779	0.572	0.737	0.554	0.826	0.582	0.783	0.534	0.798	0.543	0.791	0.539
	Avg	**0.533**	**0.416**	0.542	0.421	0.557	0.430	0.557	0.460	0.542	0.456	0.585	0.470	0.540	0.422	0.552	0.430	0.546	0.426
Elec-tricity	96	**0.210**	**0.293**	0.219	0.299	0.244	0.319	0.228	0.322	0.225	0.312	0.255	0.345	0.218	0.301	0.226	0.308	0.222	0.305
	192	**0.224**	**0.305**	0.231	0.326	0.241	0.320	0.230	0.313	0.227	0.317	0.288	0.369	0.231	0.314	0.239	0.322	0.235	0.318
	336	**0.243**	**0.320**	0.253	0.341	0.256	0.335	0.249	0.333	0.245	0.334	0.321	0.391	0.251	0.332	0.259	0.339	0.255	0.336
	720	**0.294**	0.369	0.303	0.370	0.307	0.373	0.297	**0.357**	**0.294**	0.370	0.346	0.399	0.303	0.378	0.311	0.386	0.307	0.383
	Avg	**0.243**	**0.322**	0.252	0.334	0.262	0.337	0.251	0.331	0.248	0.333	0.303	0.376	0.251	0.331	0.259	0.339	0.255	0.336
Traffic	96	**0.736**	**0.416**	0.755	0.428	0.746	0.420	0.772	0.413	0.874	0.463	0.909	0.444	0.812	0.438	0.835	0.446	0.798	0.431
	192	**0.703**	**0.404**	0.772	0.437	0.754	0.410	0.787	0.417	0.804	0.437	0.945	0.464	0.783	0.425	0.808	0.433	0.771	0.418
	336	**0.712**	**0.405**	0.821	0.457	0.767	0.413	0.796	**0.405**	0.814	0.439	0.981	0.479	0.791	0.428	0.815	0.436	0.778	0.421
	720	**0.757**	**0.423**	0.904	0.488	0.827	0.437	0.886	0.434	0.872	0.462	1.034	0.505	0.845	0.451	0.868	0.459	0.831	0.445
	Avg	**0.727**	**0.412**	0.813	0.453	0.774	0.420	0.810	0.417	0.841	0.450	0.967	0.473	0.808	0.436	0.832	0.444	0.795	0.429
1st count		**25**		0		1		3		9		0		0		0		0	

Table 8. Statistical significance summary: number of datasets (out of 7) where FreqMoE significantly outperforms each baseline (one-sided paired t-test, $p < 0.05$).

Baseline	Win	Tie	Loss
iTransformer	7	0	0
PatchTST	5	2	0
FreTS	7	0	0
DLinear	4	1	2
Stationary	7	0	0
RevIN	6	1	0
Diversify	7	0	0
TimesNet	6	1	0

5.6. Ablation Experiments

Table 9 presents the results of ablation experiments conducted on the components of FreqMoE. It can be observed that the complete FreqMoE model achieves the best performance. Removing any component leads to a decrease in performance, with the removal of the frequency-domain MoE causing the most significant impact. Specifically, the MAE increases by 5.5% on weather and 6.2% on electricity, highlighting the critical role of the frequency-domain MoE. The removal of the Masking Mechanism also results in performance degradation, underscoring the importance of this regularization strategy in enhancing the model's robustness. Additionally, the Temporal Smoothing Module filters out noise caused by unexpected events, enhancing FreqMoE's capability to capture the intrinsic characteristics of the sequence.

We further include three additional ablation variants. **Random Routing** replaces the spectrum-embedding-based routing with uniform weights, resulting in noticeable performance degradation, which demonstrates the benefit of spectrum-based expert selection. **Time-domain MoE** applies the MoE architecture in the time domain instead of the frequency domain, showing larger degradation that demonstrates the importance of frequency-domain modeling. **w/o Mask + w/o MA** removes both the masking and moving average components simultaneously, resulting in the largest degradation, indicating that these components provide complementary benefits.

5.7. Analysis of Model Degradation

Table 10 evaluates the performance of various models across both in-distribution and out-of-distribution scenarios, along with the corresponding performance decay. For the degradation analysis, both in-distribution (IN) and out-of-distribution (OOD) results for all models were obtained under our unified experimental framework. The IN results use a standard chronological split (first 70% for training, next 20% for validation, last 10% for testing, following the protocol from [29]), while the OOD results use our temporal-gap protocol (Table 5). All models were run with the same implementation framework, optimization settings, and random seeds to ensure fair comparison. In the in-distribution prediction tasks, all models demonstrate relatively stable performance. However, under out-of-distribution scenarios, the baseline models experience significant performance degradation. On the large-scale traffic dataset, the MSE of iTransformer increases from 0.428 to 0.813, reflecting a decay of 89.9%. In contrast, performance decay of our proposed FreqMoE is only 28.2%, which is lower than that of the other models. Notably, the OOD-specific baselines RevIN, Diversify, and TimesNet also show significant degradation on traffic (59.6–66.0%), while FreqMoE achieves the lowest average degradation across the evaluated datasets, though the advantage is not uniform across all datasets and horizons. For example, on ETTh2, FreqMoE still exhibits substantial absolute degradation (OOD MSE of 1.156, a 206.6% increase over the IN setting), and on ETTh1, FreTS achieves a comparable

degradation rate (98.9% vs. FreqMoE's 98.0%). The primary advantage of FreqMoE is most pronounced on electricity (17.3%) and traffic (28.2%), where it substantially outperforms all baselines. These experimental results demonstrate the effectiveness of FreqMoE in addressing OOD challenges on average, though we note that the robustness advantage is dataset dependent rather than universal.

Table 9. Ablation experiments of FreqMoE. w/o Mask, w/o MA, and w/o MoE represent the removal of Masking Mechanism, Temporal Smoothing Module, and frequency-domain MoE, respectively. Random Routing replaces spectrum-based routing with uniform weights. Time-domain MoE applies the MoE in the time domain instead of the frequency domain. The best results are **bolded**.

		FreqMoE		**w/o Mask**		**w/o MA**		**w/o MoE**		**Random Routing**		**Time-Domain MoE**		**w/o Mask + w/o MA**	
	T_{out}	**MSE**	**MAE**	**MSE**	**MAE**	**MSE**	**MAE**	**MSE**	**MAE**	**MSE**	**MAE**	**MSE**	**MAE**	**MSE**	**MAE**
ETTh1	96	**0.790**	**0.544**	0.815	0.562	0.809	0.552	0.840	0.579	0.821	0.561	0.831	0.570	0.878	0.595
	192	**0.880**	**0.588**	0.911	0.603	0.901	0.599	0.937	0.619	0.914	0.610	0.934	0.621	0.986	0.653
	336	**0.954**	**0.628**	0.978	0.651	0.976	0.643	1.009	0.670	0.991	0.654	1.021	0.666	1.071	0.696
	720	**1.109**	**0.719**	1.146	0.741	1.127	0.734	1.175	0.761	1.166	0.759	1.201	0.769	1.268	0.806
	Avg	**0.933**	**0.620**	0.963	0.639	0.953	0.632	0.990	0.657	0.973	0.646	0.997	0.657	1.051	0.688
ETTh2	96	**0.787**	**0.601**	0.813	0.617	0.801	0.610	0.837	0.641	0.815	0.617	0.847	0.633	0.875	0.657
	192	**1.018**	**0.674**	1.045	0.693	1.037	0.684	1.078	0.710	1.051	0.701	1.082	0.717	1.148	0.742
	336	**1.224**	**0.737**	1.263	0.759	1.253	0.749	1.300	0.777	1.276	0.771	1.307	0.783	1.383	0.817
	720	**1.595**	**0.857**	1.643	0.885	1.623	0.873	1.686	0.909	1.672	0.894	1.720	0.912	1.820	0.960
	Avg	**1.156**	**0.717**	1.225	0.759	1.178	0.729	1.191	0.739	1.204	0.746	1.239	0.761	1.307	0.794
Weather	96	**0.339**	**0.318**	0.346	0.331	0.346	0.325	0.357	0.335	0.348	0.328	0.362	0.337	0.382	0.350
	192	**0.448**	**0.379**	0.459	0.389	0.452	0.385	0.475	0.399	0.466	0.388	0.483	0.403	0.497	0.412
	336	**0.575**	**0.441**	0.594	0.453	0.586	0.449	0.614	0.466	0.598	0.463	0.619	0.472	0.650	0.488
	720	**0.771**	**0.525**	0.797	0.545	0.787	0.536	0.814	0.557	0.813	0.554	0.831	0.561	0.875	0.589
	Avg	**0.533**	**0.416**	0.549	0.429	0.543	0.424	0.565	0.439	0.556	0.433	0.574	0.443	0.601	0.460
Elec.	96	**0.210**	**0.293**	0.213	0.300	0.211	0.300	0.221	0.312	0.215	0.304	0.222	0.309	0.241	0.315
	192	**0.224**	**0.305**	0.233	0.316	0.230	0.315	0.234	0.321	0.230	0.322	0.239	0.321	0.251	0.329
	336	**0.243**	**0.320**	0.253	0.333	0.246	0.323	0.258	0.341	0.255	0.332	0.266	0.339	0.272	0.357
	720	**0.294**	**0.369**	0.299	0.377	0.302	0.376	0.310	0.396	0.310	0.383	0.312	0.397	0.345	0.412
	Avg	**0.243**	**0.322**	0.249	0.332	0.247	0.328	0.255	0.342	0.252	0.335	0.260	0.342	0.277	0.353

Table 10. Average MSE for in-distribution (IN) and out-of-distribution (OOD) scenarios with percentage decrease (Dec. %). The best results are **bolded** and the second best results are underlined.

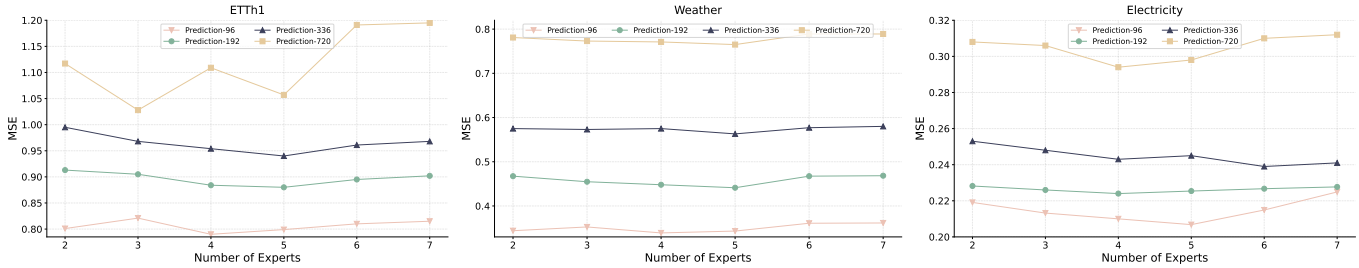
Methods	**ETTh1**			**ETTh2**			**Electricity**			**Traffic**		
	IN	**OOD**	**Dec. (%)**	**IN**	**OOD**	**Dec. (%)**	**IN**	**OOD**	**Dec. (%)**	**IN**	**OOD**	**Dec. (%)**
Stationary	0.570	1.502	163.5%	0.526	1.973	275.0%	0.193	0.303	56.9%	0.624	0.967	54.9%
DLinear	0.456	0.919	101.5%	0.559	2.604	365.8%	0.209	0.248	18.6%	0.625	0.841	34.5%
FreTS	0.488	0.970	98.9%	0.550	3.548	545.0%	0.209	0.251	20.0%	0.552	0.810	31.8%
PatchTST	0.469	0.997	112.5%	0.378	1.166	208.4%	0.205	0.262	27.8%	0.481	0.774	60.9%
iTransformer	0.454	0.984	116.7%	0.383	1.182	208.6%	0.178	0.252	41.5%	0.428	0.813	89.9%
RevIN	0.461	0.946	105.2%	0.372	1.196	221.5%	0.181	0.218	20.4%	0.489	0.812	66.0%
Diversify	0.478	0.962	101.3%	0.385	1.229	219.2%	0.198	0.259	30.8%	0.515	0.835	62.1%
TimesNet	0.465	0.950	104.3%	0.371	1.195	222.1%	0.184	0.222	20.7%	0.498	0.795	59.6%
FreqMoE	0.471	0.933	**98.0%**	0.377	1.156	**206.6%**	0.207	0.243	**17.3%**	0.567	0.727	**28.2%**

5.8. Hyperparameter Sensitivity

To analyze the sensitivity of FreqMoE to key parameters, we conducted a systematic study of the number of experts, masking ratio, moving average window size, and top-k frequency components. As illustrated in Figure 6, the model's prediction performance

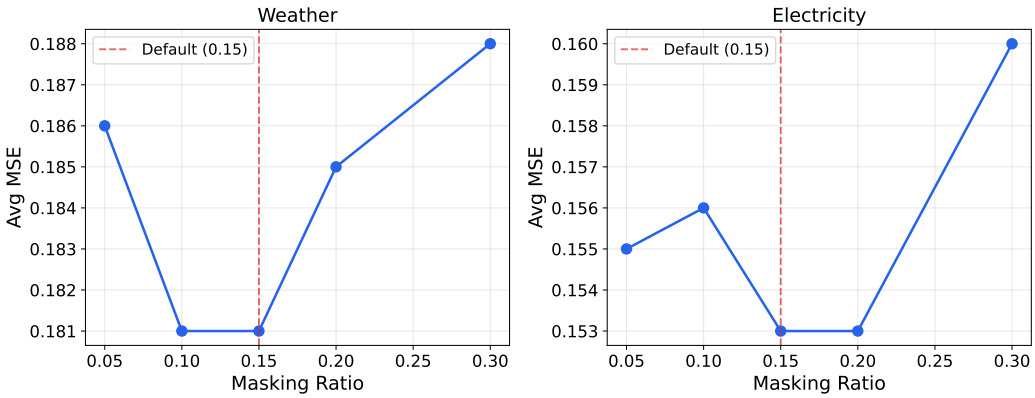
demonstrates a distinct nonlinear relationship with the number of experts. Specifically, when the number of experts is small, increasing their count significantly reduces prediction error. However, once the number of experts surpasses a certain threshold, the model's performance plateaus and eventually declines, suggesting that while additional experts increase the model's capacity, they also elevate the risk of overfitting.

Figure 7 presents the sensitivity analysis for the masking ratio. The results show that $r = 0.15$ provides a good balance: too small a ratio provides insufficient regularization, while too large a ratio removes excessive information. Figure 8 shows the sensitivity to the moving average window size, with $w = 5$ yielding the best overall performance. Figure 9 presents the sensitivity to the top-k frequency components, indicating that $k = 4$ provides a good trade-off between information retention and routing dimensionality.



(**a**) (**b**) (**c**)

Figure 6. Impact of the number of experts on prediction performance across different datasets. (**a**) ETTh1; (**b**) Weather; (**c**) Electricity.



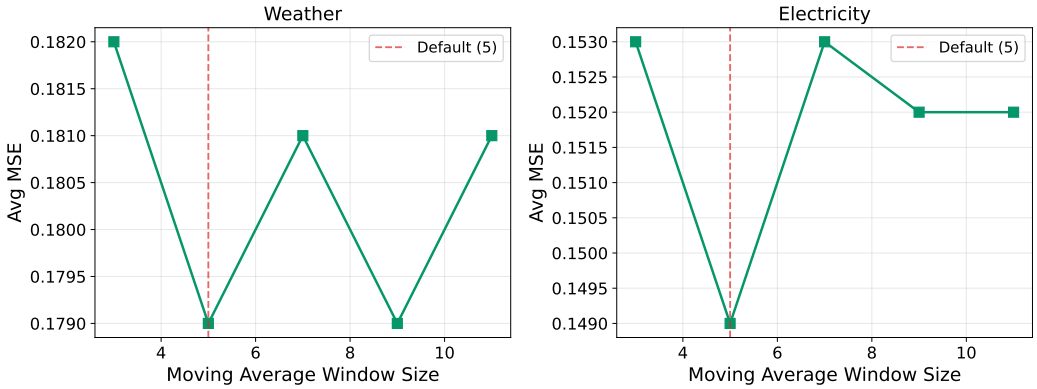
(**a**) Weather (**b**) Electricity

Figure 7. Impact of the masking ratio on prediction performance. The red dashed line indicates the default value ($r = 0.15$).

5.9. Efficiency Analysis

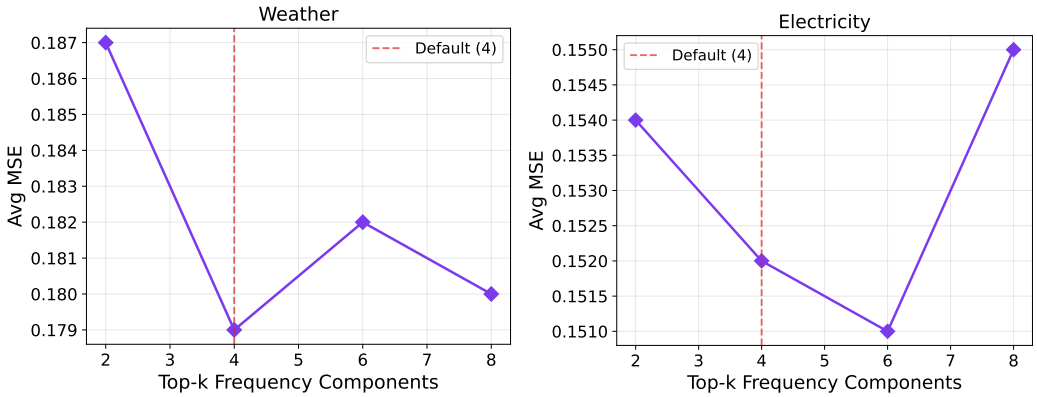
We comprehensively evaluate the performance of models in terms of predictive accuracy, parameter count, computational cost, inference latency, memory usage, and training speed. As illustrated in Figure 10 and Table 11, FreqMoE achieves a favorable balance between computational efficiency and prediction accuracy. All models are evaluated under identical conditions—input length: 96, prediction length: 96, batch size: 32, full precision (FP32)—on a single NVIDIA RTX A6000 48GB GPU, with all baselines implemented under the same PyTorch framework with consistent optimization settings. Specifically, FreqMoE has only 0.42M parameters, reducing GPU memory usage by 59.6% and shortening training time by 13.1% relative to the benchmark iTransformer. Moreover, compared to FreTS, an established frequency-domain temporal modeling framework, FreqMoE achieves superior

predictive capabilities while demanding merely 7.16% of the GPU memory resources and 17.84% of the training time.



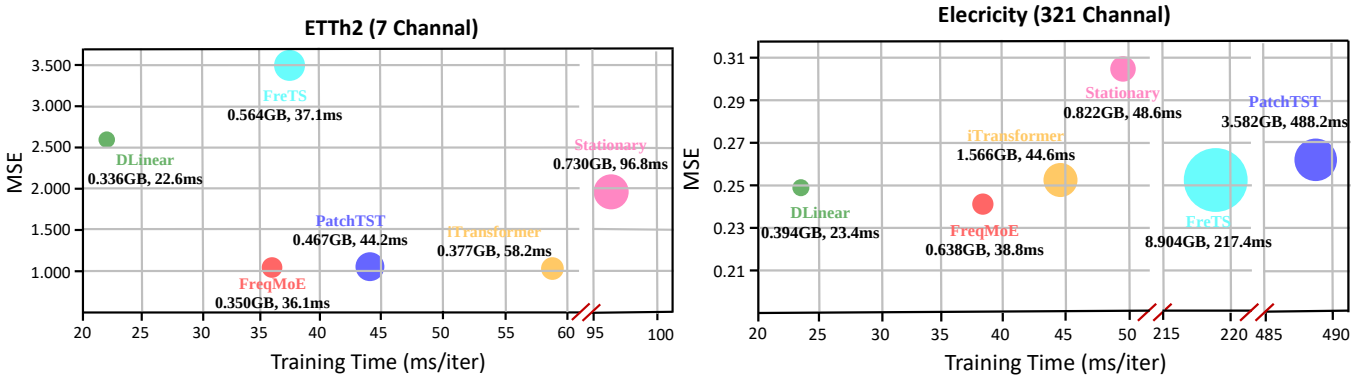
(**a**) Weather (**b**) Electricity

Figure 8. Impact of the moving average window size on prediction performance. The red dashed line indicates the default value ($w = 5$).



(**a**) Weather (**b**) Electricity

Figure 9. Impact of the top-k frequency components on prediction performance. The red dashed line indicates the default value ($k = 4$).



(**a**) (**b**)

Figure 10. Model efficiency comparison on the ETTh2 and electricity datasets. (**a**) ETTh2; (**b**) Electricity.

5.10. Interpretability Analysis

To provide interpretability beyond prediction error metrics, we analyze the learned representations and routing behavior of FreqMoE.

Representation Visualization. Figure 11 shows t-SNE visualizations of frequency-domain embeddings from FreqMoE, iTransformer, and DLinear, comparing in-distribution

and OOD segments. FreqMoE produces more overlapping embeddings across distributions, providing qualitative evidence consistent with more aligned representations across distributions. We note that t-SNE visualizations are suggestive rather than conclusive; the observed overlap is consistent with reduced distributional sensitivity but does not constitute formal proof of invariant representation learning.

Table 11. Comprehensive efficiency comparison across models. Params, FLOPs, and inference latency are reported. The best results are **bolded**.

Model	Params (M)	FLOPs (G)	Latency (ms)
FreqMoE	**0.42**	**0.31**	2.8
iTransformer	1.13	0.85	5.1
PatchTST	0.68	0.52	3.9
FreTS	0.51	0.38	3.2
DLinear	0.14	0.08	**0.9**
Stationary	1.21	0.92	5.6
RevIN	0.16	0.09	1.0
TimesNet	0.78	0.58	4.1

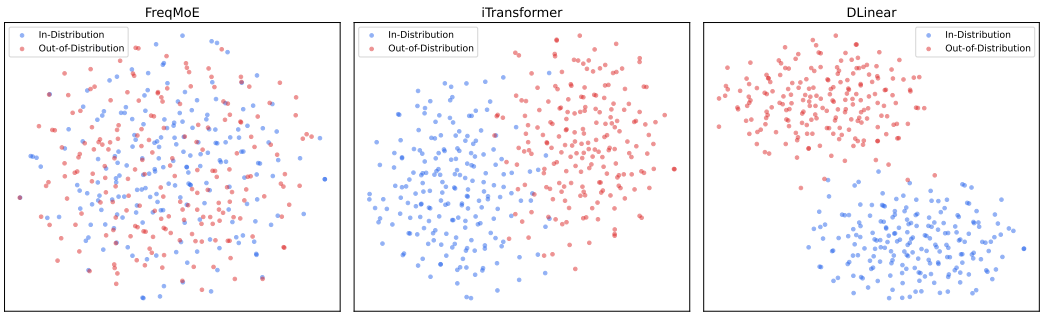


Figure 11. t-SNE visualization of learned embeddings from FreqMoE, iTransformer, and DLinear. Blue points represent in-distribution segments; red points represent OOD segments. FreqMoE produces more aligned representations across distributions.

Routing Probability Visualization. Figure 12 visualizes the expert routing probability distribution across representative inputs from different datasets. The results show that experts exhibit distinct activation patterns correlated with input frequency characteristics, indicating suggestive evidence of meaningful frequency-domain specialization.

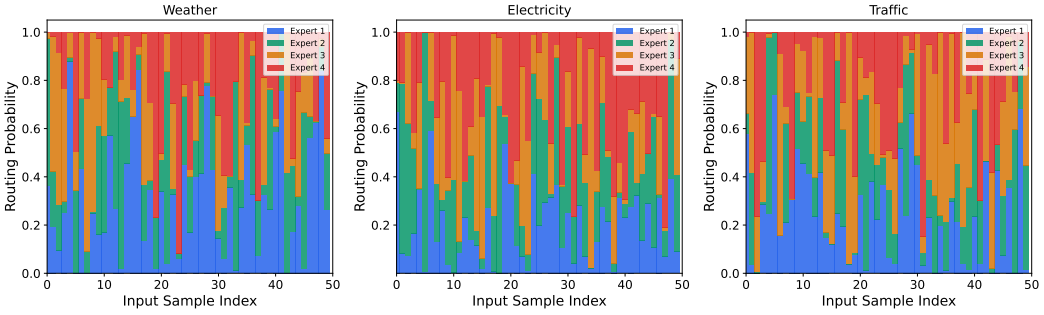


Figure 12. Expert routing probability distribution across weather, electricity, and traffic datasets. Each color represents the routing probability assigned to a specific expert.

6. Conclusions and Future Work

In this paper, we propose FreqMoE, a framework that integrates frequency-domain modeling, random masking, and Mixture of Experts routing for time series forecasting in OOD scenarios. Our main contributions are: (1) a frequency-domain MoE architecture

where each expert independently models amplitude and phase components of the spectrum; (2) a spectrum-embedding-based router that dynamically assigns expert weights based on input frequency characteristics; (3) a masking-based regularization strategy that encourages robust frequency-domain representations. Experimental results on seven real-world datasets demonstrate that FreqMoE achieves competitive or leading performance compared to existing models in OOD scenarios, with the lowest performance degradation under temporal distribution shifts.

Limitations and Future Work. This study has several limitations. First, our OOD evaluation relies on temporal holdout, which represents naturalistic but not controlled distribution shifts; future work should evaluate them under synthetic shift scenarios and on datasets with documented domain shifts. Second, we do not provide formal theoretical guarantees for invariant feature learning; such analysis through invariant risk minimization frameworks would strengthen the theoretical foundation. Third, the evaluation is limited to benchmark datasets; real-world deployment scenarios would provide stronger validation. Fourth, the masking strategy simulates point-level data incompleteness but may not fully capture complex shift types such as seasonal regime changes or structural breaks. Future work will explore adaptive masking strategies, extend to multivariate frequency routing, and investigate theoretical connections to invariant risk minimization.

Author Contributions: Conceptualization, A.S. and Z.L.; methodology, A.S.; validation, A.S. and Z.L.; formal analysis, T.W.; investigation, Z.L. and T.W.; data curation, Z.L. and T.W.; writing—original draft preparation, Z.L.; writing—review and editing, A.S. and Z.L.; visualization, T.W.; supervision, A.S.; project administration, A.S. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The ETT dataset can be acquired at https://github.com/zhouhaoyi/ETDataset (accessed on 9 May 2026), the Weather dataset can be acquired at https://www.bgc-jena.mpg.de/wetter/ (accessed on 9 May 2026), the Electricity dataset can be acquired at https://archive.ics.uci.edu/dataset/321/electricityloaddiagrams/ (accessed on 9 May 2026), and the Traffic dataset can be acquired at https://pems.dot.ca.gov/ (accessed on 9 May 2026). The source code, configuration files, and experiment scripts for FreqMoE are being organized and will be released as a public GitHub repository.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Li, Z.; Han, J.; Song, Y. On the forecasting of high-frequency financial time series based on ARIMA model improved by deep learning. *J. Forecast.* **2020**, *39*, 1081–1097.
2. Yang, M.; Wang, J. Adaptability of financial time series prediction based on BiLSTM. *Procedia Comput. Sci.* **2022**, *199*, 18–25. [CrossRef]
3. Ma, C.; Dai, G.; Zhou, J. Short-term traffic flow prediction for urban road sections based on time series analysis and LSTM_BILSTM method. *IEEE Trans. Intell. Transp. Syst.* **2021**, *23*, 5615–5624. [CrossRef]
4. Zhang, J.; Chen, F.; Cui, Z.; Guo, Y.; Zhu, Y. Deep learning architecture for short-term passenger flow forecasting in urban rail transit. *IEEE Trans. Intell. Transp. Syst.* **2020**, *22*, 7004–7014. [CrossRef]
5. Deb, C.; Zhang, F.; Yang, J.; Lee, S.E.; Shah, K.W. A review on time series forecasting techniques for building energy consumption. *Renew. Sustain. Energy Rev.* **2017**, *74*, 902–924. [CrossRef]
6. Wang, S.; Wu, H.; Shi, X.; Hu, T.; Luo, H.; Ma, L.; Zhang, J.Y.; Zhou, J. TimeMixer: Decomposable Multiscale Mixing for Time Series Forecasting. In Proceedings of the Twelfth International Conference on Learning Representations, Vienna, Austria, 7–11 May 2024.

7. Wu, H.; Xu, J.; Wang, J.; Long, M. Autoformer: Decomposition transformers with auto-correlation for long-term series forecasting. *Adv. Neural Inf. Process. Syst.* **2021**, *34*, 22419–22430. [CrossRef]
8. Liu, Y.; Wu, H.; Wang, J.; Long, M. Non-stationary transformers: Exploring the stationarity in time series forecasting. *Adv. Neural Inf. Process. Syst.* **2022**, *35*, 9881–9893. [CrossRef]
9. Kim, T.; Kim, J.; Tae, Y.; Park, C.; Choi, J.H.; Choo, J. Reversible instance normalization for accurate time-series forecasting against distribution shift. In Proceedings of the Tenth International Conference on Learning Representations, Virtual, 25–29 April 2022.
10. Proakis, J.G.; Manolakis, D.G. *Digital Signal Processing: Principles Algorithms and Applications*; Pearson India Education: Tamil Nadu, India, 2007.
11. Zhou, T.; Ma, Z.; Wen, Q.; Wang, X.; Sun, L.; Jin, R. Fedformer: Frequency enhanced decomposed transformer for long-term series forecasting. In *Proceedings of the 39th International Conference on Machine Learning*; PMLR: Baltimore, MD, USA, 2022; pp. 27268–27286. [CrossRef]
12. Zhou, T.; Ma, Z.; Wen, Q.; Sun, L.; Yao, T.; Yin, W.; Jin, R. Film: Frequency improved legendre memory model for long-term time series forecasting. *Adv. Neural Inf. Process. Syst.* **2022**, *35*, 12677–12690. [CrossRef]
13. Yi, K.; Zhang, Q.; Fan, W.; Wang, S.; Wang, P.; He, H.; An, N.; Lian, D.; Cao, L.; Niu, Z. Frequency-domain MLPs are more effective learners in time series forecasting. *Adv. Neural Inf. Process. Syst.* **2024**, *36*, 76656–76679. [CrossRef]
14. Liu, Y.; Li, C.; Wang, J.; Long, M. Koopa: Learning non-stationary time series dynamics with koopman predictors. *Adv. Neural Inf. Process. Syst.* **2023**, *36*, 12271–12290. [CrossRef]
15. Lu, W.; Wang, J.; Sun, X.; Chen, Y.; Ji, X.; Yang, Q.; Xie, X. Diversify: A general framework for time series out-of-distribution detection and generalization. *IEEE Trans. Pattern Anal. Mach. Intell.* **2024**, *46*, 4534–4550. [CrossRef]
16. Shi, X.; Wang, S.; Nie, Y.; Li, D.; Ye, Z.; Wen, Q.; Jin, M. Time-MoE: Billion-Scale Time Series Foundation Models with Mixture of Experts. *arXiv* **2024**, arXiv:2409.16040.
17. Liu, X.; Liu, J.; Woo, G.; Aksu, T.; Liang, Y.; Zimmermann, R.; Liu, C.; Li, J.; Savarese, S.; Xiong, C.; et al. Moirai-MoE: Empowering Time Series Foundation Models with Sparse Mixture of Experts. *arXiv* **2025**, arXiv:2410.10469.
18. Yi, K.; Fei, J.; Zhang, Q.; He, H.; Hao, S.; Lian, D.; Fan, W. FilterNet: Harnessing frequency filters for time series forecasting. *arXiv* **2024**, arXiv:2411.01623.
19. Piao, X.; Chen, Z.; Murayama, T.; Matsubara, Y.; Sakurai, Y. Fredformer: Frequency Debiased Transformer for Time Series Forecasting. *arXiv* **2024**, arXiv:2406.09009.
20. Wang, H.; Chen, J.; Pan, T.; Dong, Z.; Jiang, R.; Song, X. Evaluating the Generalization Ability of Spatiotemporal Model in Urban Scenario. *IEEE Trans. Mob. Comput.* **2025**, *24*, 13535–13548. [CrossRef]
21. Du, Y.; Wang, J.; Feng, W.; Pan, S.; Qin, T.; Xu, R.; Wang, C. Adarnn: Adaptive learning and forecasting of time series. In Proceedings of the 30th ACM International Conference on Information & Knowledge Management, Queensland, Australia, 1–5 November 2021; pp. 402–411.
22. Wang, B.; Ma, J.; Wang, P.; Wang, X.; Zhang, Y.; Zhou, Z.; Wang, Y. Stone: A spatio-temporal ood learning framework kills both spatial and temporal shifts. In Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, Barcelona, Spain, 25–29 August 2024; pp. 2948–2959.
23. Ni, R.; Lin, Z.; Wang, S.; Fanti, G. Mixture-of-Linear-Experts for Long-term Time Series Forecasting. In *Proceedings of the 27th International Conference on Artificial Intelligence and Statistics*; PMLR: Valencia, Spain, 2024; pp. 4672–4680.
24. Pan, Y.A.; Guo, J.; Chen, Y.; Cheng, Q.; Li, W.; Liu, Y. A fundamental diagram based hybrid framework for traffic flow estimation and prediction by combining a Markovian model with deep learning. *Expert Syst. Appl.* **2024**, *238*, 122219. [CrossRef]
25. Shazeer, N.; Mirhoseini, A.; Maziarz, K.; Davis, A.; Le, Q.; Hinton, G.; Dean, J. Outrageously Large Neural Networks: The Sparsely-Gated Mixture-of-Experts Layer. *arXiv* **2017**, arXiv:1701.06538.
26. Zhou, H.; Zhang, S.; Peng, J.; Zhang, S.; Li, J.; Xiong, H.; Zhang, W. Informer: Beyond efficient transformer for long sequence time-series forecasting. In Proceedings of the AAAI Conference on Artificial Intelligence, Virtual, 19–21 May 2021; Volume 35, pp. 11106–11115.
27. Lai, G.; Chang, W.C.; Yang, Y.; Liu, H. Modeling Long- and Short-Term Temporal Patterns with Deep Neural Networks. In Proceedings of the 41st International ACM SIGIR Conference on Research & Development in Information Retrieval, New York, NY, USA, 8–12 July 2018; SIGIR '18; pp. 95–104. [CrossRef]
28. Müller, M. Dynamic time warping. In *Information Retrieval for Music and Motion*; Springer: Berlin/Heidelberg, Germany, 2007; pp. 69–84.
29. Liu, Y.; Hu, T.; Zhang, H.; Wu, H.; Wang, S.; Ma, L.; Long, M. itransformer: Inverted transformers are effective for time series forecasting. *arXiv* **2024**, arXiv:2310.06625.
30. Nie, Y.; Nguyen, N.H.; Sinthong, P.; Kalagnanam, J. A Time Series is Worth 64 Words: Long-term Forecasting with Transformers. *arXiv* **2022**, arXiv:2211.14730
31. Zeng, A.; Chen, M.; Zhang, L.; Xu, Q. Are transformers effective for time series forecasting? In Proceedings of the AAAI Conference on Artificial Intelligence, Washington, DC, USA, 7–14 February 2023; Volume 37, pp. 11121–11128.

32. Wu, H.; Hu, T.; Liu, Y.; Zhou, H.; Wang, J.; Long, M. Timesnet: Temporal 2d-variation modeling for general time series analysis. In Proceedings of the Eleventh International Conference on Learning Representations, Kigali, Rwanda, 1–5 May 2023.
33. Kingma, D.P.; Ba, J. Adam: A method for stochastic optimization. *arXiv* **2014**, arXiv:1412.6980.
34. Paszke, A.; Gross, S.; Massa, F.; Lerer, A.; Bradbury, J.; Chanan, G.; Killeen, T.; Lin, Z.; Gimelshein, N.; Antiga, L.; et al. Pytorch: An imperative style, high-performance deep learning library. *Adv. Neural Inf. Process. Syst.* **2019**, *32*, 8026–8037. [CrossRef]

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