FreqMixer: Adaptive Cross-Frequency Learning for Time Series Forecasting

Yi Zhou*, Yaojun Liang*†, Chong Chen, Tao Wang, Lianglun Cheng
Guangdong Provincial Key Laboratory of Cyber-Physical Systems
Guangdong University of Technology, Guangzhou, China
1411775132@qq.com, 2112404347@mail2.gdut.edu.cn, chenc2021@gdut.edu.cn,
wangtao_cps@gdut.edu.cn, LLCheng@gdut.edu.cn

***Abstract*—Real-world time series often exhibit complex and non-stationary temporal dynamics, posing substantial challenges for accurate forecasting. While existing models predominantly rely on simplified decomposition for trends and seasonality, they often struggle to disentangle the multifaceted frequency components inherent in real-world data, thereby limiting their representational power. To address this, we propose FreqMixer, a novel MLP-based framework designed to capture rich temporal patterns from a frequency-aware perspective. The architecture begins by leveraging a Stationary Wavelet Transform (SWT) to decompose series into multi-frequency coefficients, enabling simultaneous modeling in both time and frequency domains. Building on this decomposition, a frequency-adaptive patching strategy is introduced to handle the heterogeneous characteristics of different bands, effectively differentiating long-term trends from short-term fluctuations. This hierarchical representation is further refined by a dual-level cross-frequency attention mechanism, which integrates local multi-scale information through MLP-based aggregation while capturing global dependencies across wavelet coefficients. Extensive experiments on diverse real-world datasets demonstrate that FreqMixer consistently achieves state-of-the-art performance, validating its effectiveness and robustness in complex time series analysis.**

***Index Terms*—Time series forecasting, Stationary Wavelet Transform, Frequency-adaptive patching, Cross-frequency attention**

I. Introduction

Multivariate time series (MTS) forecasting plays a central role in critical application domains such as weather forecasting [1], traffic analysis [2], and financial modeling [3]. In real-world scenarios, however, time series data are frequently affected by non-stationarity, noise, and regime shifts, giving rise to complex temporal patterns that challenge representation learning and long-horizon generalization [4]–[6].

The development of MTS forecasting methods has evolved from classical statistical models, such as ARIMA [7], to modern deep learning architectures. While RNNs and Transformers demonstrate strong capabilities in modeling complex temporal dependencies, they face an inherent tension between contextual coverage and computational efficiency. Short input sequences restrict access to long-term historical information, whereas excessively long sequences incur prohibitive computational costs and amplify noise, ultimately degrading learning effectiveness [8], [9]. In addition, large pre-trained models often struggle with domain-specific adaptation and exhibit biases inherited from their pre-training corpora [10]–[12]. Notably, recent studies reveal that comparatively simple linear or MLP-based models can outperform sophisticated Transformer architectures on standard forecasting benchmarks [13], [14].

Despite their favorable efficiency and generalization properties, existing MLP-based forecasting models often face an inherent trade-off between expressive complexity and computational overhead, frequently forcing a compromise between the high capacity of Transformers and the efficiency of linear architectures [15]. A primary bottleneck in these efficient models is their reliance on an implicit and restrictive assumption: a uniform temporal semantic granularity. In practice, they adopt fixed patching strategies that mechanically divide sequences into equal-length segments, regardless of the underlying temporal dynamics. Such rigid partitioning inevitably truncates meaningful semantic units and discards critical boundary information [16], leading to a fundamental mismatch between model architecture and the intrinsic multi-scale nature of real-world time series. Specifically, temporal signals are typically composed of stable low-frequency trends superimposed with volatile high-frequency fluctuations, whose semantic structures cannot be adequately captured under a single, fixed granularity.

To address this granularity mismatch, we propose **FreqMixer**, a novel MLP-based forecasting framework that enforces frequency-consistent patch semantics. FreqMixer first employs the Stationary Wavelet Transform (SWT) to explicitly decompose the input series into multiple frequency components. Building on this decomposition, we introduce a frequency-adaptive patching strategy in which coarse-grained patches are assigned to low-frequency components to capture long-term trends, while fine-grained patches are used for high-frequency components to preserve short-term dynamics. To effectively integrate information across resolutions, we further design a dual-layer cross-frequency attention mechanism, enabling principled interaction between different patch granularities and frequency scales. As a result, Mixer operations in FreqMixer go beyond simple token mixing and facilitate structured information fusion across multiple temporal resolutions. Our main contributions are summarized as follows:

*These authors contributed equally to this work.
†Corresponding author.

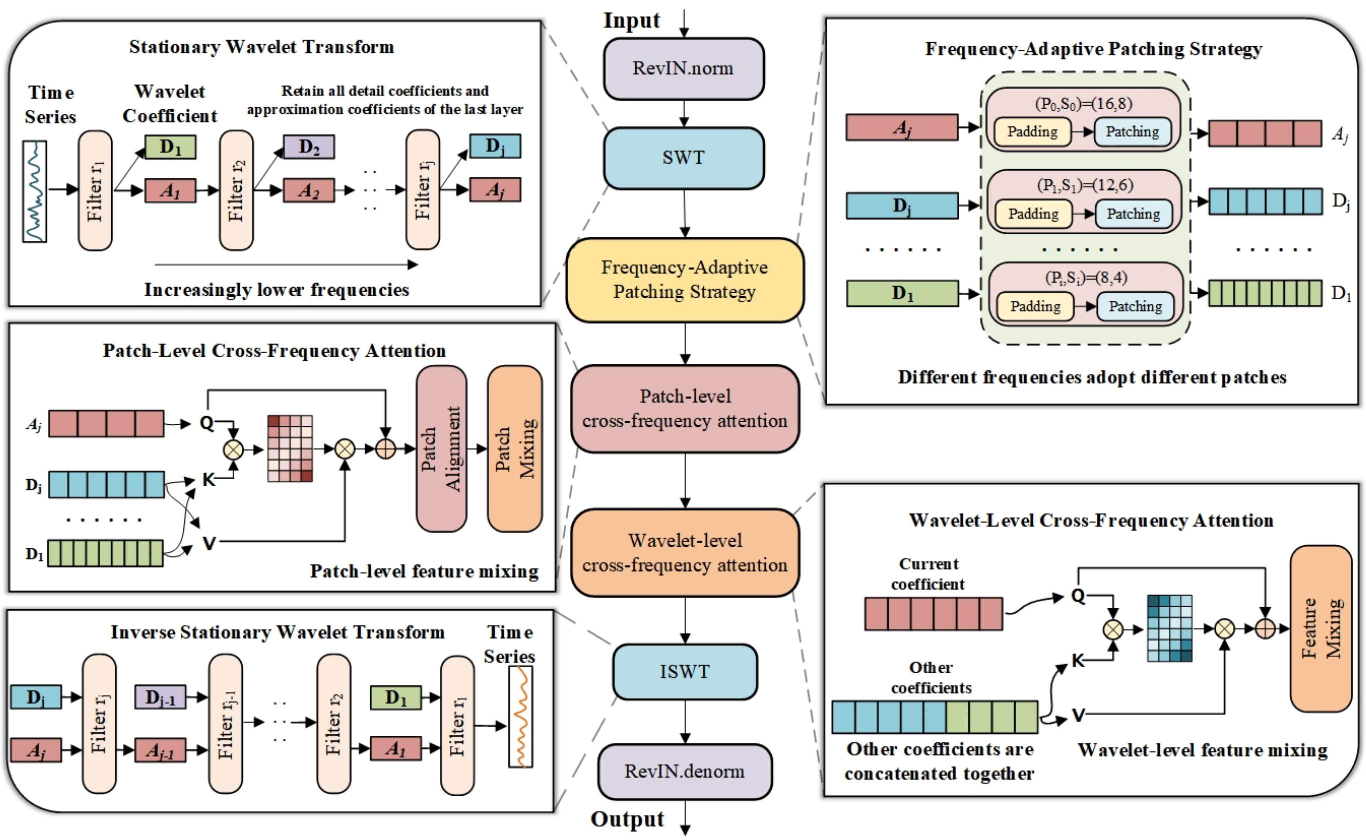

Fig. 1. The FreqMixer framework. SWT is used to decompose the time series into j levels, where A_i represents the approximation coefficients, r_i is the dilation rate that determines the sampling interval of the filter on the signal. P_i and S_i represent the patch size and stride, respectively.

- We identify the granularity mismatch inherent in existing patch-based forecasting models and propose a frequency-adaptive patching strategy that aligns patch semantics with frequency characteristics, yielding more faithful multi-scale representations.
- We introduce a dual-layer cross-frequency attention mechanism that explicitly models dependencies between coarse- and fine-grained temporal features, enabling effective fusion across frequency scales.
- Extensive experiments on multiple benchmark datasets demonstrate that FreqMixer achieves state-of-the-art forecasting performance while maintaining high computational efficiency.

Reproducibility and Availability: We provide our source code and pre-trained model checkpoints to support transparency and reproducibility; all project resources are hosted on https://github.com/Morii-s/FreqMixer.

II. Related Work

The advancement of deep learning models has significantly propelled research in multivariate time series forecasting, showcasing their remarkable potential in this domain.

Transformer-based models have demonstrated strong performance in time series forecasting. Informer [17] reduces computational complexity and effectively models long-term dependencies by introducing a sparse attention mechanism. Autoformer [18] enhances performance by adaptively decomposing time series into trend and seasonal components for independent modeling. FEDformer [19] combines the Fast Fourier Transform with decomposition to better capture frequency-domain features. PatchTST [20] enables efficient learning of both local and global patterns by partitioning the time series into patches. iTransformer [21] focuses on optimizing the attention mechanism to improve information extraction capabilities in long-sequence forecasting. However, these Transformer-based models often face high computational complexity and limited local modeling capabilities.

Recent MLP-based models have emerged as efficient alternatives to Transformer architectures for time series forecasting. TSMixer [22] employs a simple yet effective MLP architecture that alternates between mixing along temporal and feature dimensions, thereby capturing dependencies without relying on attention mechanisms. TimeMixer [13] introduces a time-channel mixing mechanism, explicitly incorporating time as a dimension in MLP operations to enhance temporal modeling. WPMixer [23] integrates wavelet-based decomposition into the mixing process, facilitating more precise modeling of multi-scale temporal patterns. CycleNet [24] captures the intrinsic cyclic structures of time series through a dual-branch design, separating periodic trends and residual components, thereby enhancing interpretability and accuracy in long-term forecasting. These models collectively demonstrate that well-designed MLP-based and hybrid architectures can achieve competitive performance while minimizing overhead.

Although these models have made significant progress in efficiency and performance, they still exhibit deficiencies in extracting features from the various frequencies of time series, which limits a deeper exploration of their intrinsic patterns. Furthermore, when dealing with high-dimensional multivariate data, they face a common challenge: their performance is highly dependent on the model's dimension, requiring a large number of parameters to overcome architectural limitations such as representation bottlenecks or information isolation issues. To address this challenge, we propose FreqMixer, a collaborative information processing architecture. Through cross-scale information interaction, FreqMixer can achieve better performance even with a smaller model dimension, which demonstrates its advantage in parameter efficiency when handling high-dimensional data.

III. Methodology

A. Model Architecture

FreqMixer is a modular time series forecasting framework designed to explicitly model cross-frequency dependencies. As illustrated in Fig. 1, it consists of three core components: the SWT, a frequency-adaptive patching strategy, and a dual-level cross-frequency attention mechanism.

Given an input sequence, we first apply reversible instance normalization (RevIN) [25] to mitigate distributional shift across samples. The normalized sequence is then decomposed by SWT into a set of approximation and detail coefficients corresponding to different frequency bands. Since different frequency components exhibit distinct temporal characteristics and information densities, we employ a frequency-adaptive patching strategy in which the patch scale is inversely related to the frequency level. Low-frequency components are partitioned using coarser patches to capture long-term trends, while high-frequency components are processed with finer patches to preserve short-term fluctuations.

The resulting multi-scale patch representations are subsequently fed into a dual-level cross-frequency attention framework. At the patch level, local interactions across different frequency bands are enabled through cross-attention, followed by a learnable alignment operation to unify patch dimensions across scales. Following intra-level temporal mixing via patch-wise MLPs, we apply wavelet-level cross-frequency attention, augmented by learnable level embeddings, to facilitate global feature fusion across all frequency components. Finally, the fused features are projected to the prediction horizon, reconstructed through inverse SWT, and transformed back to the original data space via inverse RevIN.

B. Stationary Wavelet Transform

To capture multi-scale temporal patterns while avoiding the time-shift problem inherent in standard discrete wavelet transforms, FreqMixer adopts the SWT [26]. SWT decomposes the input signal into a final-level approximation and detail coefficients via low-pass and high-pass filters without downsampling, thereby preserving temporal alignment.

The decomposition is performed in a layer-wise manner, where the approximation coefficients from the previous level are further decomposed at the current level. To expand the receptive field without introducing additional parameters, we follow a dilated convolution strategy [27], in which the dilation rate increases exponentially with the decomposition level. Formally, the decomposition is defined as:

$$A_j[k] = \sum_n h[n] \cdot A_{j-1}[k - r \cdot n], \tag{1}$$

$$D_j[k] = \sum_n g[n] \cdot A_{j-1}[k - r \cdot n], \tag{2}$$

where $h[n]$ and $g[n]$ denote the low-pass and high-pass filters, respectively, and r is the dilation rate at level j. To reduce redundancy while preserving essential information, we retain only the final-level approximation coefficient together with all detail coefficients, forming a compact multi-frequency representation of the input sequence.

C. Frequency-Adaptive Patching Strategy

Existing patch-based time series models typically rely on fixed patch sizes and strides, which are insufficient to accommodate the heterogeneous temporal characteristics of different frequency components. To address this limitation, we propose a frequency-adaptive patching strategy in which the patch size and stride are determined by the frequency level of wavelet coefficients. Instead of using absolute patch configurations, our design follows a scale-consistent principle that maintains a stable ratio between patch granularity and the characteristic temporal scale of each frequency band.

Specifically, low-frequency components associated with long-term trends are partitioned using larger patches to maximize temporal coverage, whereas high-frequency components with rapid fluctuations are processed using smaller patches to preserve fine-grained local information. As illustrated in Fig. 2, this hierarchical patching strategy aligns the granularity of patch representations with the information density of each wavelet level, thereby enhancing multi-scale modeling.

Since frequency-dependent patching leads to different numbers of patches across wavelet levels, direct tensor-wise operations are infeasible. Let N_m denote the number of patches at frequency level m. To enable subsequent cross-frequency interaction, all patch sequences are aligned to a unified target length $N_{\text{target}} = \min_m N_m$ through linear interpolation along the patch dimension. This interpolation establishes a continuous mapping between the original and target patch indices, ensuring temporal smoothness while avoiding information loss caused by truncation or zero-padding.

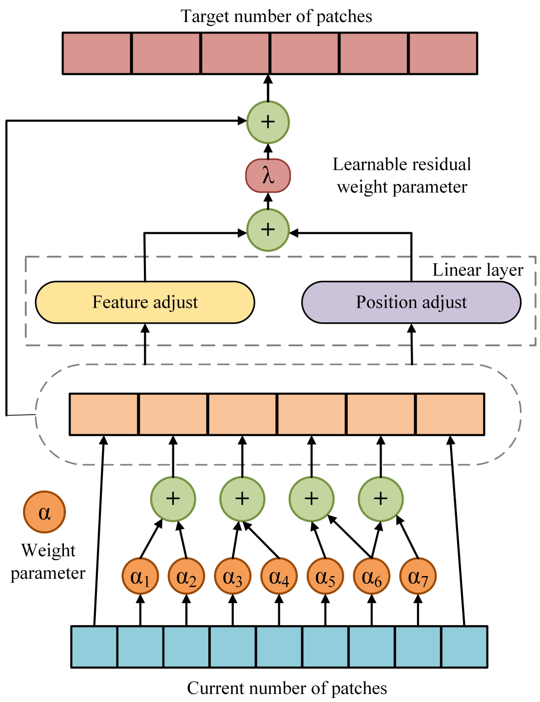

Fig. 2. Patch alignment. Temporal continuity is maintained through linear interpolation, and the interpolation results are further optimized using a learnable feature adjustment layer and residual connections.

Although linear interpolation preserves temporal continuity, it lacks adaptability to frequency-specific feature distributions. To enhance alignment quality, we introduce a learnable alignment refinement mechanism that augments the interpolated features with frequency-aware adjustments:

$$\mathbf{F}_m^{\text{aligned}} = \mathbf{F}'_m + \lambda \cdot \left(\phi(\mathbf{F}'_m) + \psi(\mathbf{F}'_m)\right), \tag{3}$$

where $\mathbf{F}'_m$ denotes the interpolated patch features at level m. The function $\phi(\cdot)$ is implemented as a two-layer MLP that adaptively refines feature representations according to frequency characteristics, while $\psi(\cdot)$ is a position-aware linear transformation designed to compensate for potential positional information loss introduced by interpolation. The residual weight λ is learnable and initialized with a small value, allowing the model to shift from interpolation-dominated alignment to frequency-adaptive refinement during training.

This alignment strategy retains the temporal continuity advantages of linear interpolation while introducing learnable, frequency-aware corrections, providing a unified and high-quality patch representation for subsequent cross-frequency attention and feature fusion.

D. Dual-Level Cross-Frequency Attention

The core of FreqMixer lies in its ability to explicitly model interactions among heterogeneous frequency components. Rather than treating each frequency band independently, we introduce a dual-level cross-frequency attention mechanism that enables structured information exchange at both local (patch-wise) and global (wavelet-wise) scales.

The first stage operates at the patch level and serves as the foundational phase of cross-frequency interaction. After frequency-adaptive patching and alignment, patch sequences from different wavelet levels are allowed to interact through cross-frequency attention. For a given frequency level m, its patch representations are used as queries, while the concatenated patches from all other frequency levels act as keys and values:

$$\text{Attn}(Q_m, K_{\neq m}, V_{\neq m}) = \text{Softmax}\left(\frac{Q_m K_{\neq m}^T}{\sqrt{d}}\right) V_{\neq m}. \tag{4}$$

This design enables large-scale, low-frequency patches to incorporate fine-grained information from high-frequency components, while high-frequency patches gain access to broader contextual cues from low-frequency trends. The attended features are fused with the original patch representations via residual connections and further processed by a patch-wise MLP, which performs intra-level temporal mixing along the patch dimension. This stage ensures that each patch representation integrates both cross-frequency information and same-frequency temporal context.

The second stage operates at the wavelet level and focuses on global cross-frequency fusion. After patch-level aggregation, each wavelet level is treated as a single semantic token augmented with a learnable level encoding. Multi-head cross-attention is then applied across frequency tokens to capture complementary relationships between approximation and detail coefficients in the feature dimension. Through this mechanism, low-frequency components receive corrective signals for local variations from high-frequency components, while high-frequency components are guided by long-term trend information encoded in low-frequency representations. The resulting features are further refined through a channel-wise MLP and combined with the input via residual connections to enhance representational stability.

Finally, the fused features are projected to the prediction horizon and reconstructed through the inverse Stationary Wavelet Transform:

$$\begin{aligned} A_{j-1}[k] = \frac{1}{2}\Bigg(&\sum_n \tilde{h}[n] \cdot A_j[k - r \cdot n] \\ &+ \sum_n \tilde{g}[n] \cdot D_j[k - r \cdot n]\Bigg), \end{aligned} \tag{5}$$

followed by inverse RevIN to obtain the final prediction.

IV. Experiments

This section will present our experimental results, including performance on long-term forecasting tasks and studies from ablation experiments, to validate the effectiveness of the proposed model and its related modules.

A. Long-term forecasting results

Setups. To comprehensively evaluate the effectiveness of our model in long-term forecasting, we conducted experiments on

TABLE I

THE RESULTS OF LONG-TERM FORECASTING ARE PRESENTED WITH A LOOK-BACK LENGTH OF 96 FOR ALL TASKS. THE BEST RESULTS ARE HIGHLIGHTED IN BOLD, WHILE THE SECOND-BEST RESULTS ARE UNDERLINED.

Models		FreqMixer (Ours)		WPMixer (2025)		SimpleTM (2025)		TimeKAN (2025)		TimeMixer (2024)		iTransformer (2024)		TSMixer (2023)		TimesNet (2023)	
Metric		MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE
ETTm1	96	**0.310**	**0.352**	0.323	0.358	0.325	0.362	0.322	0.361	0.326	0.363	0.344	0.368	0.331	0.378	0.338	0.375
	192	**0.352**	**0.375**	0.362	0.377	0.361	0.381	0.357	0.383	0.369	0.389	0.377	0.391	0.386	0.399	0.374	0.387
	336	**0.378**	**0.397**	0.388	**0.397**	0.392	0.404	0.392	0.406	0.396	0.407	0.426	0.420	0.426	0.421	0.410	0.411
	720	**0.440**	**0.433**	0.449	0.437	0.455	0.438	0.451	0.437	0.454	0.441	0.491	0.459	0.489	0.465	0.478	0.450
ETTm2	96	**0.172**	**0.252**	**0.172**	0.253	0.176	0.259	0.174	0.255	0.175	0.258	0.180	0.264	0.179	0.282	0.187	0.267
	192	**0.236**	**0.296**	0.237	0.297	0.238	0.299	0.240	0.300	0.238	0.301	0.250	0.309	0.244	0.305	0.249	0.309
	336	**0.292**	**0.332**	**0.292**	0.335	0.295	0.338	0.304	0.342	0.298	0.340	0.311	0.348	0.320	0.357	0.321	0.351
	720	**0.391**	**0.390**	0.392	0.394	0.393	0.395	0.396	0.398	0.395	0.397	0.412	0.407	0.419	0.432	0.408	0.403
ETTh1	96	**0.364**	**0.391**	0.371	0.395	0.366	0.393	0.367	0.395	0.387	0.400	0.386	0.405	0.387	0.411	0.384	0.402
	192	**0.414**	0.419	0.427	**0.418**	0.422	0.421	**0.414**	0.420	0.441	0.430	0.441	0.436	0.441	0.437	0.436	0.429
	336	0.455	0.439	0.469	0.437	**0.446**	0.442	0.452	**0.435**	0.518	0.474	0.487	0.458	0.507	0.467	0.638	0.469
	720	**0.447**	**0.454**	0.464	0.455	0.475	0.470	0.476	0.465	0.490	0.474	0.503	0.491	0.527	0.548	0.521	0.500
ETTh2	96	**0.283**	**0.335**	0.284	0.338	0.285	0.340	0.290	0.340	0.290	0.343	0.297	0.349	0.308	0.355	0.340	0.374
	192	**0.358**	**0.386**	0.366	0.388	0.360	0.388	0.375	0.392	0.383	0.401	0.380	0.400	0.393	0.405	0.402	0.414
	336	0.406	0.421	0.379	0.408	**0.367**	**0.401**	0.423	0.435	0.417	0.431	0.428	0.432	0.428	0.434	0.452	0.452
	720	**0.418**	**0.438**	0.423	0.440	**0.418**	**0.438**	0.462	0.462	0.430	0.447	0.427	0.445	0.443	0.451	0.462	0.468
Weather	96	**0.161**	0.207	0.165	**0.206**	0.162	0.207	0.162	0.208	0.166	0.212	0.174	0.214	0.175	0.247	0.172	0.220
	192	**0.206**	**0.248**	0.210	**0.248**	0.208	0.249	0.208	0.252	0.209	0.250	0.221	0.254	0.224	0.294	0.219	0.261
	336	**0.263**	0.290	0.265	**0.289**	0.265	0.290	0.264	0.291	0.264	0.293	0.278	0.296	0.262	0.326	0.280	0.306
	720	0.343	0.340	0.340	**0.338**	0.341	0.342	**0.338**	0.341	0.349	0.346	0.358	0.347	0.349	0.348	0.365	0.359
Electricity	96	0.150	0.241	0.151	0.242	**0.148**	0.242	0.175	0.258	0.153	0.247	**0.148**	**0.240**	0.190	0.299	0.168	0.272
	192	0.164	0.255	0.164	0.254	**0.151**	**0.248**	0.182	0.273	0.166	0.256	0.162	0.253	0.216	0.323	0.184	0.322
	336	0.181	0.272	0.180	0.271	**0.173**	**0.267**	0.197	0.286	0.185	0.277	0.178	0.269	0.226	0.334	0.198	0.300
	720	0.219	0.307	0.220	0.306	**0.200**	**0.294**	0.236	0.320	0.225	0.310	0.225	0.317	0.250	0.353	0.220	0.320

six widely used real-world datasets, including ETTh1, ETTh2, ETTm1, ETTm2, Weather, and Electricity.

Baselines and Evaluation. We compared FreqMixer with seven recent time series forecasting methods, namely WPMixer [23], SimpleTM [27], TimeKAN [28], TimeMixer [13], iTransformer [21], TSMixer [22], and TimesNet [29]. Following the practices of TSMixer and TimeMixer, all datasets were normalized to have a zero mean and unit standard deviation. The normalized datasets served as the foundation for ground truth in our evaluations. For long-term forecasting, prediction lengths were set to 96, 192, 336, and 720, consistent with previous studies. During the training phase, mean squared error (MSE) and mean absolute error (MAE) were used for performance evaluation.

Results. The prediction results are shown in Table I. Lower MSE and MAE values indicate higher prediction accuracy. The experimental results demonstrate that FreqMixer performs exceptionally well on low-dimensional datasets (e.g., ETTh1, ETTh2, ETTm1, ETTm2, and Weather). On the ETTm1 dataset, our model reduced the MSE by 3.99%, 2.76%, 2.58%, and 2.00% for prediction horizons of 96, 192, 336, and 720, respectively, compared to the WPMixer model, thereby showcasing its predictive advantages on low-dimensional time series data. Furthermore, our model also exhibited strong robustness and competitiveness on other low-dimensional datasets.

Beyond accuracy, FreqMixer also exhibits clear advantages in parameter efficiency on high-dimensional datasets.

TABLE II

MODEL PARAMETERS ON THE ELECTRICITY DATASET UNDER DIFFERENT PREDICTION LENGTHS.

T	FreqMixer	WPMixer	SimpleTM	iTransformer
96	**282,081**	3,921,842	842,344	4,833,888
192	**319,041**	4,093,970	867,016	4,883,136
336	**374,481**	4,352,162	904,024	4,957,008
720	**522,321**	5,040,674	1,002,712	5,154,000

As shown in Table II, FreqMixer requires substantially fewer parameters on the Electricity dataset while maintaining competitive forecasting performance. This efficiency stems from its channel-decoupling design combined with frequency-adaptive segmentation and lightweight cross-frequency attention, which enables effective information interaction across frequency branches without excessive parameter growth.

In contrast, SimpleTM embeds all variables into a shared latent space, which inevitably dilutes the representational capacity allocated to each variable as the number of variables grows, thereby requiring a rapid increase in model dimensionality and parameter size. Conversely, WPMixer decomposes the task into independent univariate frequency branches. While this design isolates frequency-specific features, it also induces localized learning and structural information silos due to the absence of sufficient cross-scale interactions, leading to redundant parameterization across branches.

TABLE III
ABLATION STUDY. THE AVERAGE MSE/MAE WITH DIFFERENT PREDICTION LENGTHS WAS USED FOR EVALUATION.

	ETTh1		ETTh2		ETTm1		ETTm2	
	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE
FreqMixer	**0.420**	**0.426**	**0.366**	**0.396**	**0.370**	**0.389**	**0.273**	**0.318**
w/o SWT	0.459	0.436	0.387	0.406	0.397	0.399	0.285	0.333
w/o Frequency-Adaptive Patching Strategy	0.433	0.431	0.374	0.402	0.379	0.397	0.275	0.321
w/o Patch-Level Cross-Frequency Attention	0.436	0.433	0.378	0.404	0.375	0.392	0.276	0.322
w/o Wavelet-level Cross-Frequency Attention	0.439	0.432	0.377	0.401	0.382	0.397	0.277	0.324
w/o Attention	0.449	0.436	0.382	0.404	0.395	0.396	0.287	0.329

We further analyze the sensitivity of FreqMixer to the number of wavelet decomposition levels. As illustrated in Fig. 3, model performance varies systematically with the decomposition depth, particularly for longer prediction horizons, indicating that an appropriate frequency hierarchy is crucial for balancing global trends and fine-grained dynamics.

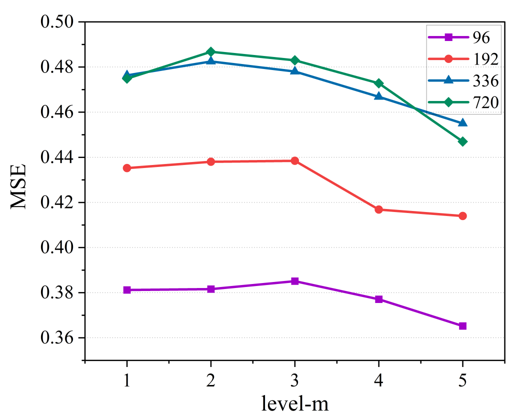

Fig. 3. The performance of FreqMixer varies with the number of decomposition levels m. Among them, 96, 192, 336, and 720 are the prediction lengths of the model.

B. Ablation Study

We conducted extensive ablation studies using the ETT dataset to evaluate the individual contribution of each module in the model. The specific results are shown in Table III.

Stationary Wavelet Transform. In ablation studies, removing the SWT module leads to a significant increase in error, and the number of decomposition levels, m, is an important hyperparameter affecting accuracy, requiring optimization for different datasets and prediction lengths.

Frequency-Adaptive Patching Strategy. Removing the frequency-adaptive patching strategy leads to a significant increase in prediction errors. Specifically, on the ETTh2 dataset, the MSE increases by approximately 2.19%, and the MAE increases by 1.36%. This demonstrates the effectiveness of this strategy in handling different frequency components and capturing the global trends and local variations of time series.

Dual-Level Cross-Frequency Attention Mechanism. To further examine the role of attention mechanisms, Fig. 4 reports the performance impact of removing different attention components on ETTh1. Removing either the patch-level or wavelet-level attention increases prediction errors, while removing both leads to the most severe degradation. This demonstrates that effective cross-frequency interaction requires joint modeling at both patch and wavelet levels.

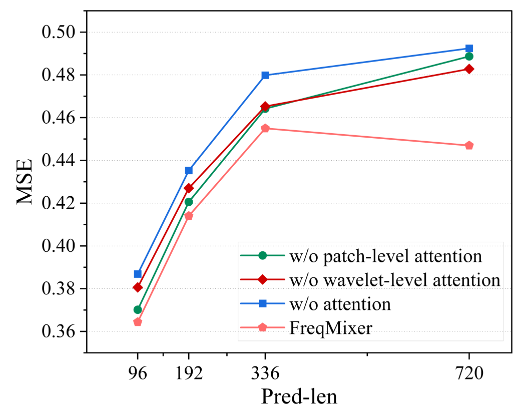

Fig. 4. The performance of the model on the ETTh1 dataset after removing different attention mechanisms.

V. CONCLUSION

In this study, we propose an efficient long-term time series forecasting model for multivariate time series. The FreqMixer model utilizes multi-level wavelet decomposition to capture information in both the time and frequency domains of the time series. It then employs an adaptive patching strategy and a dual cross-frequency attention mechanism to further enhance information interaction between different frequencies, thereby improving forecasting performance. Our experimental results demonstrate that our model, FreqMixer, is competitive compared to existing baseline models and holds significant potential for broad applications in time series analysis.

ACKNOWLEDGMENT

This work was supported by the Guangdong Basic and Applied Basic Research Foundation (2025A1515011306), Guangzhou Science and Technology Planning Project (2025B03J0056) and Guangdong Provincial Key Laboratory of Cyber-Physical System (2020B1212060069).

DECLARATION OF GENERATIVE AI AND AI-ASSISTED TECHNOLOGIES IN THE WRITING PROCESS

While preparing this work, the authors used OpenAI's ChatGPT to assist in language polishing and refining the clarity of the manuscript. After using this tool, the authors reviewed and edited the content as needed and take full responsibility for the content of the published article.

References

[1] L. Chen, S. Xiao, S. Ding, S. Hu, and L. Sun, "Integrating sequence and image modeling in irregular medical time series through self-supervised learning," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 39, no. 15, 2025, pp. 15 821–15 829.

[2] L. Cao, B. Wang, G. Jiang, Y. Yu, and J. Dong, "Spatiotemporal-aware trend-seasonality decomposition network for traffic flow forecasting," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 39, no. 11, 2025, pp. 11 463–11 471.

[3] C. Yang, J. Wang, X. Jiang, and J. Wu, "Learning universal multi-level market irrationality factors to improve stock return forecasting," in *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining V.1*, ser. KDD '25. New York, NY, USA: Association for Computing Machinery, 2025, p. 1739–1750. [Online]. Available: https://doi.org/10.1145/3690624.3709328

[4] Z. Liu, M. Cheng, Z. Li, L. Zhu, L. Chen, and J. Xu, "Adaptive normalization for non-stationary time series forecasting: A temporal slice perspective," *Advances in Neural Information Processing Systems*, vol. 36, pp. 14 273–14 292, 2023.

[5] Y. Liu, C. Li, J. Wang, and M. Long, "Koopa: Learning non-stationary time series dynamics with Koopman predictors," *Advances in Neural Information Processing Systems*, vol. 36, pp. 12 271–12 290, 2023.

[6] P. Liu, B. Wu, Y. Hu, N. Li, T. Dai, J. Bao, and S.-T. Xia, "Timebridge: Non-stationarity matters for long-term time series forecasting," in *Forty-second International Conference on Machine Learning*, 2025.

[7] A. A. Ariyo, A. O. Adewumi, and C. K. Ayo, "Stock price prediction using the arima model," in *2014 UKSim-AMSS 16th international conference on computer modelling and simulation*. IEEE, 2014, pp. 106–112.

[8] Q. Wen, T. Zhou, C. Zhang, W. Chen, Z. Ma, J. Yan, and L. Sun, "Transformers in time series: a survey," in *Proceedings of the Thirty-Second International Joint Conference on Artificial Intelligence*, 2023, pp. 6778–6786.

[9] A. Zeng, M. Chen, L. Zhang, and Q. Xu, "Are transformers effective for time series forecasting?" in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 37, no. 9, 2023, pp. 11 121–11 128.

[10] Z. Zhou and R. Yu, "Can LLMs understand time series anomalies?" in *The Thirteenth International Conference on Learning Representations*, 2025. [Online]. Available: https://openreview.net/forum?id=LGafQ1g2D2

[11] J. Ye, W. Zhang, K. Yi, Y. Yu, Z. Li, J. Li, and F. Tsung, "A survey of time series foundation models: Generalizing time series representation with large language model," *CoRR*, vol. abs/2405.02358, 2024. [Online]. Available: https://arxiv.org/abs/2405.02358

[12] Y. M, Q. Wen, Y. Liang *et al.*, "Large models for time series and spatio-temporal data: A survey and outlook," *CoRR*, vol. abs/2305.10316, 2023. [Online]. Available: https://arxiv.org/abs/2305.10316

[13] S. Wang, H. Wu, X. Shi, T. Hu, H. Luo, L. Ma, J. Y. Zhang, and J. ZHOU, "Timemixer: Decomposable multiscale mixing for time series forecasting," in *The Twelfth International Conference on Learning Representations*, 2024.

[14] Y. Wang, Y. Qiu, P. Chen, Y. Shu, Z. Rao, L. Pan, B. Yang, and C. Guo, "Lightgts: A lightweight general time series forecasting model," in *Proceedings of the 42nd International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research. PMLR, 2025. [Online]. Available: https://arxiv.org/abs/2506.06005

[15] J. Ni, Z. Liu, S. Wang, Y. Zhang, and J. Li, "Timedistill: Efficient long-term time series forecasting with mlp via cross-architecture distillation," *CoRR*, vol. abs/2501.12345, 2025. [Online]. Available: https://arxiv.org/abs/2501.12345

[16] Q. Huang, L. Shen, R. Zhang *et al.*, "Hdmixer: Hierarchical dependency with extendable patch for multivariate time series forecasting," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 38, no. 11, 2024, pp. 12 608–12 616.

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

[18] H. Wu, J. Xu, J. Wang, and M. Long, "Autoformer: Decomposition transformers with auto-correlation for long-term series forecasting," *Advances in neural information processing systems*, vol. 34, pp. 22 419–22 430, 2021.

[19] T. Zhou, Z. Ma, Q. Wen, X. Wang, L. Sun, and R. Jin, "Fedformer: Frequency enhanced decomposed transformer for long-term series forecasting," in *International conference on machine learning*. PMLR, 2022, pp. 27 268–27 286.

[20] Y. Nie, N. H. Nguyen, P. Sinthong, and J. Kalagnanam, "A time series is worth 64 words: Long-term forecasting with transformers," in *The Eleventh International Conference on Learning Representations*, 2023.

[21] Y. Liu, T. Hu, H. Zhang, H. Wu, S. Wang, L. Ma, and M. Long, "itransformer: Inverted transformers are effective for time series forecasting," in *The Twelfth International Conference on Learning Representations*, 2024.

[22] S.-A. Chen, C.-L. Li, S. Ö. Arik, N. C. Yoder, and T. Pfister, "Tsmixer: An all-mlp architecture for time series forecast-ing," *Trans. Mach. Learn. Res.*, 2023.

[23] M. M. N. Murad, M. Aktukmak, and Y. Yilmaz, "Wpmixer: Efficient multi-resolution mixing for long-term time series forecasting," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 39, no. 18, 2025, pp. 19 581–19 588.

[24] S. Lin, W. Lin, X. Hu, W. Wu, R. Mo, and H. Zhong, "Cyclenet: Enhancing time series forecasting through modeling periodic patterns," *Advances in Neural Information Processing Systems*, vol. 37, pp. 106 315–106 345, 2024.

[25] T. Kim, J. Kim, Y. Tae, C. Park, J.-H. Choi, and J. Choo, "Reversible instance normalization for accurate time-series forecasting against distribution shift," in *International conference on learning representations*, 2021.

[26] G. P. Nason and B. W. Silverman, "The stationary wavelet transform and some statistical applications," in *Wavelets and statistics*. Springer, 1995, pp. 281–299.

[27] H. Chen, V. Luong, L. Mukherjee, and V. Singh, "Simpletm: A simple baseline for multivariate time series forecasting," in *The Thirteenth International Conference on Learning Representations*, 2025.

[28] S. Huang, Z. Zhao, C. Li, and L. Bai, "Timekan: Kan-based frequency decomposition learning architecture for long-term time series forecasting," *arXiv preprint arXiv:2502.06910*, 2025.

[29] H. Wu, T. Hu, Y. Liu, H. Zhou, J. Wang, and M. Long, "Timesnet: Temporal 2d-variation modeling for general time series analysis," in *The Eleventh International Conference on Learning Representations*, 2023.