

ADAKAN: KOLMOGOROV-ARNOLD NETWORKS WITH ADAPTIVE SPECTRAL DECOMPOSITION FOR TIME SERIES FORECASTING

Anonymous authors

Paper under double-blind review

ABSTRACT

Real-world time series typically contain intertwined frequency components: low-frequency (trends), mid-frequency (periodicities), and high-frequency (short-term fluctuations or unexpected events), posing a significant challenge for accurate time series forecasting. To address this, we propose **AdaKAN**, a novel time–frequency Kolmogorov–Arnold Network (**KAN**) equipped with an **Adaptive Spectral Filter Module**. Specifically: (i) AdaKAN adaptively decomposes time series into low-, mid-, and high-frequency components via learnable spectral thresholds. (ii) Fourier KAN captures global dependencies and periodic patterns, while Temporal KAN focuses on local structures and temporal dependencies. (iii) Each frequency band is processed with KANs of different orders with a soft weighting mechanism to better capture frequency-specific dynamics. Finally, time and frequency features are fused to form a comprehensive representation. Extensive experiments on multiple benchmarks demonstrate that AdaKAN consistently outperforms existing SOTA methods, offering a superior balance of accuracy and efficiency as an extremely lightweight architecture. Anonymous code is available in Appendix.

1 INTRODUCTION

The widespread deployment of IoT sensors and wearable devices results in the continuous and dynamic collection of time series data. Time series forecasting (TSF), which predicts future trends from historical observations, facilitates efficient planning and decision-making (Wu et al., 2024a). TSF has extensive applications across finance, meteorology, healthcare, and transportation fields (Huang et al., 2024; Wu et al., 2024b; Wang et al., 2024b; Shibo et al., 2025).

Current deep learning approaches for TSF primarily include CNNs, RNNs, MLPs, and Transformers. CNNs are effective at capturing local patterns but struggle with long-term dependencies (Zeng et al., 2024), while RNNs address temporal dependencies better but suffer from low efficiency and poor parallelization. Transformers such as iTransformer (Liu et al., 2024a) and PatchTST (Nie et al., 2023) have become mainstream due to their ability to model long-range dependencies via self-attention, yet their quadratic complexity and tendency to overfit small datasets pose key limitations (Wu et al., 2025). DLlinear (Zeng et al., 2023) has questioned their suitability for TSF, highlighting that permutation-invariant attention may impair temporal information preservation. While linears sometimes surprisingly outperform Transformers, they often fail under noisy or complex patterns. Recently, Kolmogorov–Arnold Network (KANs) have emerged as a promising alternative to MLPs in TSF (Xu et al., 2024a), offering efficient nonlinear representations with adaptive flexibility.

Time series typically exhibit non-stationarity and multi-periodicity, complicating the mapping from historical observations to future predictions. Recent TSF methods, such as Autoformer (Wu et al., 2021) and TimeMixer (Wang et al., 2024a), have integrated prior knowledge via seasonal-trend decomposition to improve performance. Frequency often offers complementary insights by revealing periodic structures and spectral distributions, critical for TSF (Jin et al., 2024), where patterns are often more distinct and interpretable (Eldele et al., 2024). Nevertheless, existing KAN-based research largely remains confined to time-domain modeling (Chen et al., 2025; Han et al., 2024a; Genet & Inzirillo, 2024). Even TimeKAN (Huang et al., 2025) employs FFT/IFFT techniques, it still operates KAN modules in the time domain, limiting their ability to capture localized spectral patterns.

Inspired by these insights, we propose AdaKAN, a lightweight **Adaptive frequency-aware KAN** forecasting framework designed for complex and multi-frequency time series. Specifically, AdaKAN comprises four core components: (i) **Adaptive spectral filter module (ASFM)**: employs adaptive low-pass, band-pass, and high-pass filters with learnable thresholding to decompose the input into low-, mid-, and high-frequency subseries. (e.g., ETTh1 in Figure 1); (ii) **FourierKAN**: models real and imaginary parts within each frequency band using separate KANs; (iii) **TemporalKAN**: applies inverse FFT followed by grouped convolutions and time-domain KANs to extract local temporal features, also employing cross-attention to model interactions among frequencies. Notably, we introduce a soft-weighted multi-order mechanism in both FourierKAN and TemporalKAN, adaptively assigning orders based on input spectral features. (iv) **Fusion module**: integrates dual KANs outputs to comprehensively capture rich representations.

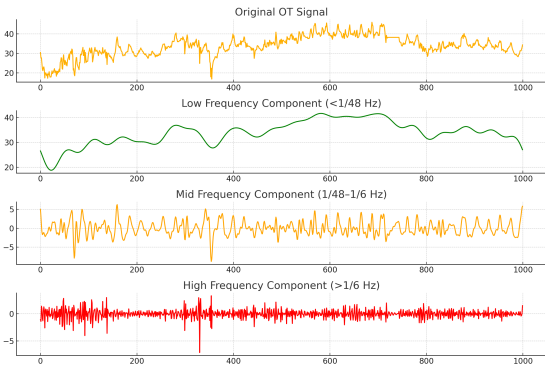


Figure 1: Decomposing 1000 samples of ETTh1 into low-, mid- and high- frequency components.

AdaKAN is lightweight, reducing complexity from $O(N^2)$ to $O(N \log N)$ employing the efficient Fast Fourier Transform (FFT). Although AdaKAN incorporates cross-attention mechanisms, they are restricted to frequency or local temporal interactions, avoiding the full quadratic cost of traditional attention, making it more efficient than self-attention (see Complexity analysis).

In summary, our contributions are as follows:

- We propose AdaKAN, a novel lightweight framework that integrates temporal-frequency KAN with an adaptive spectral filter module (ASFM) to effectively model and represent patterns across different frequencies by maximizing the flexibility of KAN.
- We revisit TSF from a frequency decomposition perspective and propose a hybrid framework that combines an adaptive spectral block and a dual-branch KAN. The spectral block uses Fourier transform with learnable thresholds to decompose inputs into low-, mid-, and high-frequency components, capturing trends, periodicity, and abrupt changes. The dual-branch KAN independently processes time- and frequency-domain features, leveraging their complementary strengths.
- Extensive experiments demonstrate that AdaKAN outperforms SOTA methods, validating its effectiveness and superiority in long-term TSF task.

2 RELATED WORK

Kolmogorov–Arnold Network (KAN). The Kolmogorov-Arnold theorem states that any multivariate function can be expressed by combinations of univariate functions and addition operations (Liu et al., 2024b). KAN replaces traditional linear weights with spline-based functions, enabling dynamic activation learning and improved interpretability. This flexibility makes KAN a compelling alternative to MLPs with fixed activation functions. KAN has been applied in diverse fields. U-KAN (Li et al., 2025) integrates KAN for efficient medical image segmentation. ViaSHAP (Alkhatib et al., 2025) learns Shapley value functions for direct prediction. GrokFormer (Ai et al., 2025) learns spectral filters via Graph Fourier KAN on classification task. Studies show KAN can surpass MLPs with fewer parameters (Vaca-Rubio et al., 2024). For TSF, T-KAN (Xu et al., 2024a) detects concept drift with symbolic regression. TimeKAN (Huang et al., 2025) integrates cascaded frequency decomposition with KANs to capture complex frequency patterns in the time domain. MMK (Han et al., 2024a) improves forecasting by assigning variables to KAN experts. Yet, most focus on the time domain, with limited use in the frequency domain. Given the multi-periodic and non-stationary nature of time series, we introduce three adaptive Fourier filters that dynamically divide the time series into high-frequency (sudden events), mid-frequency (seasonal cycles), and low-frequency (long-term trends) components, improving interpretability and robustness.

Time Series Decomposition. Real-world time series often consist of multiple latent patterns. To better capture these diverse dynamics, recent methods increasingly adopt decomposition strategies. N-BEATS (Oreshkin et al., 2020) and N-HiTS (Challu et al., 2023) incorporate decomposition into deep learning and achieve strong forecasting performance, but they overlook inter-channel dependencies critical to time series. DLinear (Zeng et al., 2023) uses moving averages to decouple trend and seasonality, while TimeMixer (Wang et al., 2024a) follows a fine-to-coarse strategy to separate sequences into multiple time spans. TimeKAN (Huang et al., 2025) proposes a decomposition learning block to accurately simulate complex patterns. Meanwhile, TimesNet (Wu et al., 2023) applies Fourier analysis to decompose sequences by learned periodic structures. Motivated by these advances, we propose a novel frequency decomposition method, employ adaptive low-pass, band-pass, and high-pass filters with learnable thresholds to divide the input sequence into low-, mid-, and high-frequency components, enabling precise modeling of frequency-specific dynamics.

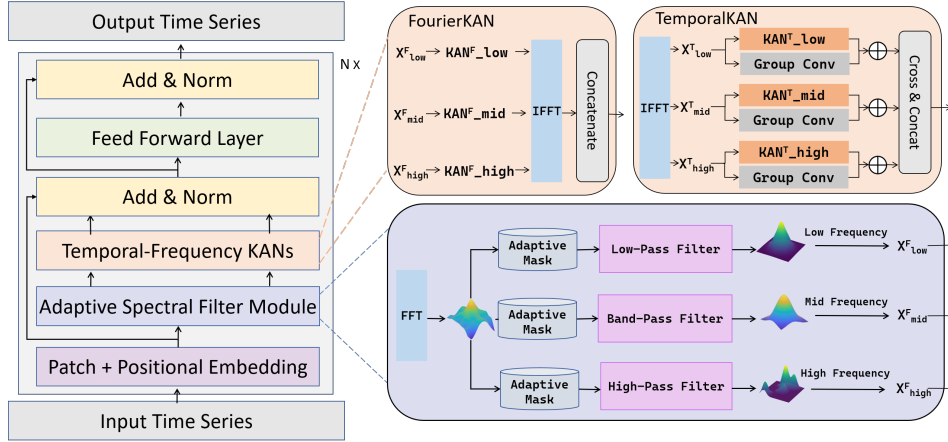


Figure 2: The structure of our proposed AdaKAN

3 METHOD

3.1 PRELIMINARIES

Kolmogorov–Arnold Network (KAN). The Kolmogorov–Arnold Representation Theorem (KART) serves as the mathematical foundation of KAN (Liu et al., 2024b), providing KAN with superior fitting capability and interpretability than MLP, which is grounded on the Universal Approximation Theorem (Kolmogorov, 1963), which states that for a function f :

$$f(x_1, \dots, x_n) = \sum_{i=1}^{2n+1} \Phi_i \left(\sum_{j=1}^n \phi_{i,j}(x_j) \right) \quad (1)$$

where $\phi_{i,j}$ are univariate functions that map each input variable x_j , and Φ_i are continuous functions. Each hidden neuron constructs an arbitrary function by composing multiple nonlinear transformations over the input features. For a single-layer KAN Φ with input dimension n_{in} and output dimension n_{out} , the mapping is given by:

$$x_j^{out} = \sum_{i=1}^{n_{in}} \phi_{i,j}(x_i^{in}) \quad (2)$$

where x_i denotes the i -th dimension of x , and $\phi_{i,j}$ represents a learnable nonlinear function.

Based on the Kolmogorov–Arnold Representation Theorem, the architecture of KAN is designed as:

$$\text{KAN}(x) = (\Phi_{L-1} \circ \dots \circ \Phi_1 \circ \Phi_0) x \quad (3)$$

where Φ_l is a layer of univariate functions Unlike traditional MLP (fixed activation functions), KAN innovatively trains learnable univariate activation functions $\phi_{i,j}$ using B-splines:

$$\phi(x) = w_b b(x) + w_s \text{spline}(x) \quad (4)$$

$$\text{spline}(x) = \sum_i c_i B_i(x) \quad (5)$$

where w_b , w_s and c_i are learnable weights, $B(x)$ is the basic function, and $\text{spline}(x)$ is parameterized as a linear combination of B-spline basis functions B_i .

3.2 OVERALL ARCHITECTURE

Our framework integrates two novel components in Figure 2: Adaptive Frequency Filter Block (ASFM) and Temporal-Frequency KAN, addressing the complexity of multi-frequency time series. ASFM decomposes the input into low-, mid-, and high-frequency components via learnable low-pass, band-pass, and high-pass filters with trainable thresholds, making it easier to learn valuable features of each frequency. After processing, these components are reconstructed into time-domain subseries using IFFT. FourierKAN models spectral characteristics within each frequency band, while TemporalKAN captures temporal dependencies using KANs with varying orders and group convolutions with different convolution kernels, allowing fine-grained extraction of local patterns. Together, they address both global and local dynamics, effectively learning trends, seasonality, and unexpected events for interpretable TSF.

3.3 EMBEDDING LAYER

Given an input time series $X \in R^{C \times L}$, where C is the channel number and L is the sequence length, X is divided into N non-overlapping patches $\{P_1, P_2, \dots, P_N\}$, each of length p , totally patches $P_i \in R^{C \times p}$. Each patch is mapped to a new dimension d , resulting in $P_i \rightarrow P'_i \in R^{C \times d}$. Position embeddings E_i are added to each patch to preserve temporal order, giving $X_{PE_i} = P'_i + E_i$. The complete set of latent patches is $X_{PE} = X_{PE_1}, X_{PE_2}, \dots, X_{PE_N}$, with the positional embeddings enhancing temporal correlation capture.

3.4 ADAPTIVE SPECTRAL FILTER MODULE

Time series are typically composed of multiple frequency components, and different frequency bands contain varying levels of dynamic information. Low frequencies reflect global trends, middle frequencies encompass periodic and medium-range state changes, and high frequencies comprise short-term fluctuations and noise interference. We model and learn the entire spectrum of the signal to decompose the whole frequency and independently learning each frequency can improve model robustness and generalization (Huang et al., 2025). To this end, we propose a frequency decomposition module based on learnable filters, which decomposes the original series \mathbf{X} into low-frequency \mathbf{X}_{low}^F , middle-frequency \mathbf{X}_{mid}^F , and high-frequency \mathbf{X}_{high}^F . The entire structure is as follows.

Fast Fourier Transformations. For a given X_{PE} , its representation is calculated as:

$$\begin{aligned} X_F &= \mathcal{F}[X_{PE}] \in \mathcal{C}^{C \times L'_F} \\ &= \int_{-\infty}^{\infty} X_{PE} \cdot e^{-j2\pi f v} dv \\ &= \int_{-\infty}^{\infty} X_{PE} \cdot \cos(2\pi f v) dv - j \int_{-\infty}^{\infty} X_{PE} \cdot \sin(2\pi f v) dv \end{aligned} \quad (6)$$

Here, $\mathcal{F}[\cdot]$ denotes the 1D FFT operation, and L'_F is the sequence length after Fourier transformation. f denotes the frequency variable, v denotes the integral variable, $j = \sqrt{-1}$ denotes the imaginary unit. We define three types of learnable filters in the frequency domain:

Adaptive Low Pass Filter: Extract stable global trends helps with TSF.

$$\mathbf{X}_{low}^F = X_F \odot (|F| \leq \theta_{low}) \quad (7)$$

where $|F| \leq \theta_{low}$ represents a binary mask where frequencies below the threshold θ_{low} are retained, while others are filtered out.

Adaptive Band Pass Filter: Capture the middle frequency to enhance model ability.

$$\mathbf{X}_{mid}^F = X_F \odot (\theta_{low} < |F| \leq \theta_{high}) \quad (8)$$

where $\theta_{low} < F \leq \theta_{high}$ denotes frequencies between the learnable thresholds θ_{low} and θ_{high} are retained, and others are discarded.

Adaptive High Pass Filter: Retain small fluctuations can help improve responsiveness to changes.

$$\mathbf{X}_{high}^F = X_F \odot (|F| > \theta_{high}) \quad (9)$$

Similarly, $|F| > \theta_{high}$ means frequencies above θ_{high} retained, while others are discarded.

3.5 TEMPORAL-FREQUENCY KAN

ChebyshevKAN. Vanilla KAN (Liu et al., 2024b) uses spline functions as the learnable univariate basic function ϕ , but its complex recursive computation process hinders the efficiency of KAN. Thus, we instead adopt ChebyshevKAN to replace the B-spline function (SS et al., 2024) for its key advantages: (i) It can approximate smooth functions more efficiently with fewer parameters. (ii) Its inherent polynomial structure makes it particularly effective at capturing frequency patterns such as periodicity. ChebyshevKAN builds ϕ through a linear combination of Chebyshev polynomials of multiple orders. The Chebyshev polynomial is defined by:

$$T_k(x) = \cos(k \arccos(x)) \quad (10)$$

where k is the highest order of the Chebyshev polynomials. Then, a single-layer ChebyshevKAN is:

$$x_j^{\text{out}} = \sum_{i=1}^{n_{in}} \phi_{i,j} (x_i^{\text{in}}) = \sum_{i=1}^{n_{in}} \sum_{k=0}^K \Theta_{i,k} T_k(\tanh(x_{in,i})) \quad (11)$$

Where $\Theta_{i,k}$ is the coefficient of k -th order Chebyshev polynomials acting on the $x_{in,i}$, and \tanh is used to normalize the inputs to between -1 and 1 . By adjusting the highest order, we can control the fitting capability of KAN. This also inspires our design of the multi-order KAN to dynamically represent different frequencies.

Multi-order with Soft Weighting. In KAN, order determines the nonlinear modeling capability of each channel (Huang et al., 2025). Adaptive selection of polynomial orders—e.g., assigning higher orders to high-frequency channels—can effectively enhance model performance and stability. We introduce a multi-order with soft weighting mechanism that assigns a learnable weight to each basis function order. Specifically, for each frequency band x_{low} , x_{mid} , and x_{high} , we assign a learnable weight vector $\alpha^{(f)} = [\alpha_0^{(f)}, \alpha_1^{(f)}, \dots, \alpha_n^{(f)}]$. The univariate $\phi(x)$ is computed as a weighted sum of basis functions, where the order weights are obtained via a softmax over trainable logits. This design adaptively emphasizes low- or high-order components, aligning model complexity with frequency hierarchy to enhance temporal pattern representation while avoiding overfitting and redundancy.

$$\phi^{(f)}(x) = \sum_{k=0}^n \alpha_k^{(f)} \cdot T_k(x) \quad (12)$$

where $f \in \{\text{low, mid, high}\}$, $\alpha^{(f)} = \text{softmax}(\mathbf{w}^{(f)})$, $\mathbf{w}^{(f)} = [w_0^{(f)}, w_1^{(f)}, \dots, w_n^{(f)}]$ is a set of learnable parameters. This mechanism enables the model to **automatically bias toward using a few polynomial orders** that best fit the corresponding frequency band.

FourierKAN. To handle complex-valued inputs $f \in X_{\text{low}}, X_{\text{mid}}, X_{\text{high}}$, we decompose it into real and imaginary parts $Re(X_f^F) = \int_{-\infty}^{\infty} X_{PE} \cdot \cos(2\pi fv) dv$ and $Im(X_f^F) = \int_{-\infty}^{\infty} X_{PE} \cdot \sin(2\pi fv) dv$. FourierKAN is composed of $\text{KAN}_{\text{low}}^F$, $\text{KAN}_{\text{mid}}^F$, $\text{KAN}_{\text{high}}^F$, and each owns a two-layer KAN; the network processes the real and imaginary components of X_f^F separately. More specifically, the real part $Re(X_f^F)$ is firstly input into the two layer KAN to get the real output $Z_f^{\text{Re}} = \text{FourierKAN}(X_f^F)$. Then, the imaginary part $Im(X_f^F)$ is input into the network to get the imaginary output $Z_f^{\text{Im}} = \text{FourierKAN}(X_f^F)$. Due to the local plasticity of KAN, the features captured from the real and imaginary parts are integrated. Using the same KAN network for both parts ensures consistent feature learning and parameter sharing for meaningful signal reconstruction. Finally, the real and imaginary outputs are combined to form the final complex-valued representation:

$$\mathbf{Z}_f^F = \mathbf{Z}_f^{\text{Re}} + j \cdot \mathbf{Z}_f^{\text{Im}} \quad (13)$$

where $X_f^F \in \{X_{\text{low}}^F, X_{\text{mid}}^F, X_{\text{high}}^F\}$.

TemporalKAN. We first use IFFT ($\mathcal{F}^{-1}(\cdot)$) to transfer $X_{\text{low}}^F, X_{\text{mid}}^F, X_{\text{high}}^F$ back to the time domain: $\mathbf{X}_{\text{low}}^T = \mathcal{F}^{-1}(\mathbf{X}_{\text{low}}^F)$, $\mathbf{X}_{\text{mid}}^T = \mathcal{F}^{-1}(\mathbf{X}_{\text{mid}}^F)$, $\mathbf{X}_{\text{high}}^T = \mathcal{F}^{-1}(\mathbf{X}_{\text{high}}^F)$.

Similar to FourierKAN, TemporalKAN also has three KAN modules, $\text{KAN}_{\text{low}}^T, \text{KAN}_{\text{mid}}^T, \text{KAN}_{\text{high}}^T$. Each adopts a dual-branch parallel architecture to separately model temporal representation learning and temporal dependency learning in a frequency-specific way, using separate KANs to learn the representation and employing group convolution to capture the temporal dependency. Specifically, group convolution employs D groups of convolution kernels to perform independent convolution operations on the series of each channel. This allows the model to focus on capturing temporal patterns without interference from inter-channel relationships. For each input $X_f^T \in \{X_{\text{low}}^T, X_{\text{mid}}^T, X_{\text{high}}^T\}$, the process of group convolution and TemporalKAN is:

$$Z_{f,1}^T = \text{Conv}_{D \rightarrow D}(X_f^T, \text{group} = D) \quad (14)$$

$$Z_{f,2}^T = \text{TemporalKAN}(X_f^T) \quad (15)$$

Final output of TemporalKAN is the sum of the outputs from each KAN and group convolution:

$$Z_f^T = Z_{f,1}^T + Z_{f,2}^T \quad (16)$$

4 EXPERIMENTS

4.1 EXPERIMENTAL SETUP

Datasets. We evaluate AdaKAN on 8 benchmarks: 4 ETTs (ETTh1, ETTh2, ETTm1, ETTm2), Electricity (ECL), Exchange, Traffic, and Weather (Wu et al., 2021). Details are in Appendix.

Baselines. We assessed AdaKAN against ten baselines, including: 1) KANs: TimeKAN (Huang et al., 2025), MMK (Han et al., 2024b); 2) Mamba: DTMamba (Wu et al., 2024c). 3) Transformers: iTransformer (Liu et al., 2024a), PatchTST (Nie et al., 2023), Crossformer (Zhang & Yan, 2023); 4) MLPs: Dlinear (Zeng et al., 2023), TimeMixer (Wang et al., 2024a); 5) CNNs: TimesNet (Wu et al., 2023); 6) Frequency-based methods: FreTS (Yi et al., 2024).

Implementation Details are detailed in Appendix.

4.2 RESULTS

Table 1 compares AdaKAN with SOTA baselines across various datasets. AdaKAN consistently achieves strong performance, while iTransformer performs best on the Electricity (321 dim) and Traffic (862 dim) datasets due to its channel-wise self-attention, which effectively models dependencies in high-dimensional data. AdaKAN, TimeKAN, and TimeMixer also perform well on long-term forecasting, underscoring the effectiveness and generality of well-designed time series decomposition architecture. Unlike other baselines, AdaKAN introduces a novel adaptive frequency decomposition framework that learns hierarchical low-, mid-, and high-frequency components in both time and frequency domains, integrated with a multi-order KAN for superior long-term forecasting. Furthermore, our AdaKAN achieves 53 wins out of 80 tests under different metrics and conditions and maintains strong performance, underscoring its versatility and effectiveness.

Table 1: Multivariate long-term series forecasting results on input length 96 and prediction lengths $\in \{96, 192, 336, 720\}$. **Bold**: best, underlined: second best, 1^{st} Count: the number of the best results. Avg. means the average results from all four prediction lengths.

Method	AdaKAN (Ours)		TimeKAN (2025)		MMK (2024)		TimeMixer (2024)		DTMamba (2024)		iTransformer (2024)		PatchTST (2023)		Crossformer (2023)		TimesNet (2023)		FreTS (2024)		Dlinear (2023)		
	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	
ETTh1	96	<u>0.374</u>	0.393	0.367	<u>0.395</u>	0.374	0.397	0.386	0.400	0.386	0.399	0.386	0.405	0.414	0.419	0.423	0.448	0.389	0.412	0.402	0.416	0.386	0.400
	192	0.414	<u>0.424</u>	<u>0.414</u>	0.420	0.419	0.429	0.428	0.426	0.426	0.424	0.441	0.436	0.460	0.445	0.471	0.474	0.440	0.442	0.472	0.462	0.437	0.432
	336	0.443	0.432	<u>0.445</u>	<u>0.435</u>	0.461	0.450	0.489	0.454	0.480	0.450	0.487	0.458	0.501	0.466	0.570	0.546	0.495	0.470	0.518	0.484	0.481	0.459
	720	<u>0.452</u>	0.455	0.444	<u>0.459</u>	0.474	0.467	0.487	0.472	0.484	0.470	0.503	0.491	0.500	0.488	0.653	0.621	0.520	0.495	0.573	0.548	0.519	0.516
	Avg.	<u>0.421</u>	0.426	0.417	<u>0.427</u>	0.432	0.436	0.448	0.438	0.444	0.435	0.454	0.448	0.469	0.455	0.529	0.522	0.461	0.455	0.491	0.478	0.456	0.452
ETTm2	96	0.285	0.334	0.290	<u>0.340</u>	0.301	0.353	0.303	0.351	0.290	0.340	0.297	0.349	0.302	0.348	0.745	0.584	0.332	0.370	0.347	0.399	0.333	0.387
	192	0.370	0.391	0.375	<u>0.392</u>	0.379	0.405	0.370	0.393	0.366	0.392	0.380	0.400	0.388	0.400	0.877	0.656	0.397	0.410	0.480	0.478	0.477	0.476
	336	0.356	0.395	0.423	<u>0.435</u>	0.432	0.446	<u>0.401</u>	<u>0.422</u>	0.380	0.409	0.428	0.432	0.426	0.433	1.043	0.731	0.446	0.448	0.519	0.509	0.594	0.541
	720	0.408	0.433	0.443	<u>0.449</u>	0.446	0.463	<u>0.412</u>	<u>0.434</u>	0.416	0.437	0.427	0.445	0.431	0.446	1.104	0.763	0.435	0.449	0.780	0.638	0.831	0.657
	Avg.	0.354	0.388	0.383	0.404	0.390	0.417	<u>0.372</u>	<u>0.400</u>	0.363	0.395	0.383	0.407	0.387	0.407	0.942	0.684	0.400	0.419	0.532	0.606	0.559	0.515
ETTh1	96	0.312	0.355	0.322	0.361	0.320	0.358	0.321	0.361	0.325	0.360	0.334	0.368	0.329	0.367	0.404	0.426	0.347	0.383	0.352	0.385	0.345	0.372
	192	0.340	0.361	0.357	<u>0.383</u>	0.364	0.383	0.367	0.387	0.375	0.386	0.377	0.391	0.367	0.385	0.450	0.451	0.382	0.399	0.394	0.406	0.380	0.389
	336	0.371	0.386	<u>0.382</u>	<u>0.401</u>	0.395	0.405	0.391	0.403	0.396	0.405	0.426	0.420	0.399	0.410	0.532	0.515	0.409	0.420	0.430	0.431	0.413	0.413
	720	0.431	0.425	<u>0.445</u>	<u>0.435</u>	0.457	0.440	0.457	0.444	0.454	0.442	0.491	0.459	0.454	0.439	0.666	0.589	0.465	0.447	0.494	0.472	0.474	0.453
	Avg.	0.364	0.382	<u>0.376</u>	<u>0.395</u>	0.384	0.397	0.384	0.399	0.388	0.399	0.407	0.410	0.387	0.400	0.513	0.496	0.401	0.412	0.418	0.424	0.403	0.407
ETTm2	96	0.169	0.253	<u>0.174</u>	<u>0.255</u>	0.176	0.261	0.176	0.258	0.177	0.259	0.180	0.264	0.175	0.259	0.287	0.366	0.189	0.266	0.194	0.290	0.193	0.292
	192	0.232	0.298	0.239	0.299	0.240	0.302	0.236	0.298	0.240	0.300	0.250	0.309	0.241	0.302	0.414	0.492	0.252	0.307	0.283	0.359	0.284	0.362
	336	0.277	0.335	0.301	0.340	0.299	0.342	<u>0.298</u>	<u>0.338</u>	0.310	0.345	0.311	0.348	0.305	0.343	0.597	0.542	0.321	0.349	0.360	0.407	0.369	0.427
	720	0.372	0.383	0.395	<u>0.396</u>	0.397	0.401	<u>0.394</u>	<u>0.398</u>	0.395	0.394	0.412	0.407	0.402	0.400	1.730	1.042	0.419	0.406	0.545	0.516	0.554	0.522
	Avg.	0.262	0.317	0.277	<u>0.322</u>	0.278	0.327	<u>0.276</u>	<u>0.323</u>	0.281	0.325	0.288	0.332	0.281	0.326	0.757	0.610	0.295	0.332	0.346	0.393	0.350	0.401
Electricity	96	0.164	0.261	0.174	0.266	0.166	0.256	<u>0.156</u>	<u>0.247</u>	0.166	0.256	0.148	0.240	0.181	0.270	0.219	0.314	0.171	0.274	0.189	0.276	0.197	0.282
	192	0.178	0.273	0.182	0.273	0.187	0.274	<u>0.169</u>	<u>0.260</u>	0.178	0.268	0.162	0.253	0.188	0.274	0.231	0.322	0.180	0.284	0.191	0.279	0.196	0.285
	336	0.192	0.282	0.197	0.286	0.204	0.290	0.186	0.277	0.197	0.289	0.178	0.269	0.204	0.293	0.246	0.337	0.200	0.306	0.296	0.299	0.301	
	720	0.227	0.316	0.236	0.320	0.247	0.323	<u>0.227</u>	<u>0.312</u>	0.243	0.326	0.225	0.317	0.246	0.324	0.280	0.363	0.234	0.324	0.246	0.332	0.245	0.333
	Avg.	0.190	0.283	0.197	0.286	0.201	0.286	<u>0.185</u>	<u>0.274</u>	0.196	0.285	0.178	0.270	0.205	0.290	0.244	0.334	0.196	0.296	0.208	0.296	0.212	0.300
Exchange	96	0.083	<u>0.203</u>	0.086	0.205	0.096	0.219	0.091	0.215	<u>0.083</u>	0.201	0.086	0.206	0.088	0.205	0.256	0.367	0.122	0.255	0.094	0.222	0.088	0.218
	192	0.170	0.295	0.173	0.296	0.190	0.311	0.197	0.318	<u>0.173</u>	<u>0.295</u>	0.177	0.299	0.176	0.299	0.470	0.509	0.256	0.361	0.222	0.350	0.176	0.315
	336	0.342	<u>0.424</u>	0.366	0.438	0.383	0.447	0.416	0.472	0.346	0.427	0.331	0.417	0.301	0.397	1.268	0.883	0.432	0.479	0.431	0.492	<u>0.313</u>	0.427
	720	0.705	0.608	0.975	0.740	0.908	0.718	0.968	0.725	0.868	0.698	0.847	0.691	0.901	0.714	1.767	1.068	1.031	0.780	1.007	0.764	<u>0.829</u>	<u>0.695</u>
	Avg.	0.325	0.382	0.400	0.420	0.394	0.424	0.418	0.433	0.368	0.405	0.360	0.403	0.367	0.404	0.940	0.707	0.460	0.469	0.439	0.457	<u>0.354</u>	0.414
Traffic	96	0.527	0.338	0.575	0.378	0.511	0.324	<u>0.462</u>	<u>0.285</u>	0.487	0.317	0.395	0.268	0.462	0.295	0.522	0.290	0.591	0.315	0.557	0.329	0.650	0.396
	192	0.535	0.344	0.550	0.362	0.529	0.330	0.473	<u>0.296</u>	0.498	0.325	0.417	0.276	0.466	0.296	0.530	0.293	0.619	0.323	0.569	0.338	0.598	0.370
	336	0.541	0.349	0.576	0.378	0.545	0.334	0.498	<u>0.296</u>	0.511	0.334	0.433	0.283	0.482	0.304	0.558	0.305	0.637	0.339	0.566	0.337	0.605	0.373
	720	0.560	0.357	0.602	0.380	0.580	0.351	<u>0.506</u>	<u>0.313</u>	0.533	0.326	0.467	0.302	0.514	0.322	0.589	0.328	0.663	0.353	0.603	0.357	0.645	0.394
	Avg.	0.540	0.347	0.576	0.375	0.541	0.335	0.484	0.297	0.507	0.326	0.428	0.282	<u>0.481</u>	<u>0.304</u>	0.550	0.304	0.628	0.333	0.574	0.340	0.625	0.383
Weather	96	0.150	0.201	0.162	<u>0.208</u>	0.164	0.210	0.163	0.209	0.171	0.218	0.174	0.214	0.177	0.218	<u>0.158</u>	0.230	0.173	0.220	0.183	0.238	0.196	0.255
	192	0.192	0.241	0.207	<u>0.249</u>	0.210	0.251	0.211	0.254	0.220	0.257	0.221	0.254	0.225	0.259	<u>0.206</u>	0.277	0.219	0.260	0.251	0.312	0.237	0.296
	336	0.242	0.286	<u>0.263</u>	<u>0.290</u>	0.265	0.290	0.263	0.293	0.274	0.296	0.278	0.296	0.278	0.297	0.272	0.335	0.285	0.306	0.272	0.316	0.233	0.335
	720	0.312	0.338	<u>0.338</u>	<u>0.340</u>	0.343	0.342	0.344	0.348	0.349	0.346	0.358	0.349	0.354	0.348	0.398	0.418	0.376	0.365	0.349	0.377	0.345	0.381
	Avg.	0.224	0.266	<u>0.242</u>	<u>0.272</u>	0.246	0.273	0.245	0.276	0.254	0.279	0.258	0.278	0.259	0.281	0.259	0.315	0.263	0.288	0.264	0.311	0.265	0.317
1^{st} Count		53		4		0		1		1		19		2		0		0		0		0	0

4.3 ABLATION STUDY

Various Variants of AdaKAN. To investigate the effectiveness of AdaKAN’s various components, we conduct ablation studies on ETTh1, ETTm1, and Weather presented in Table 2. Removing group convolution and cross-attention significantly degrades performance, highlighting the importance of modeling temporal dependencies. Using fixed-order KANs at 2 yields inferior results compared to our multi-order with soft weighting design, confirming the effectiveness of gradually increasing the order to adapt to different frequency components. Among all components, removing ASFM causes the most substantial performance drop, with MSE increasing by 3.7%, 5.7%, and 5.9% on ETTh1, ETTm1, and Weather, demonstrating the value of our adaptive frequency decomposition. Moreover, eliminating either FourierKAN, TemporalKAN, or their fusion (T-F KAN) leads to consistent declines, MSE increased by 1.7%, 2.1%, and 3.0% on ETTh1, reflecting the importance of jointly modeling both spectral and temporal features. More details are in Appendix.

KAN vs. MLP. To investigate the contributions of KAN, we conduct ablation studies by replacing them with standard MLPs in different parts: (1) MLP_time and (2

Table 2: Ablation study of each component in AdaKAN.

Variants	ETTh1 (avg.)		ETTm1 (Avg.)		Weather (avg.)	
	MSE	MAE	MSE	MAE	MSE	MAE
w/o Grop conv	0.423	0.427	0.366	0.383	0.225	0.267
w/o Cross-Att	0.426	0.430	0.369	0.388	0.229	0.270
w/o Multi-order	0.424	0.429	0.367	0.385	0.228	0.268
w/o ASFM	0.437	0.439	0.386	0.397	0.238	0.277
w/o FourierKAN	0.429	0.432	0.372	0.391	0.230	0.271
w/o TemporalKAN	0.430	0.434	0.374	0.392	0.232	0.273
w/o T-F KAN	0.434	0.436	0.378	0.395	0.234	0.275
AdaKAN	0.421	0.426	0.364	0.382	0.224	0.266

MLP in the time and frequency domains, (3) MLP_time_freq: replace KANs in both domains. Table 3 reports the average results over four forecasting horizons $\in \{96, 192, 336, 720\}$ on ETTh1, ETTm1, and Weather. Across all settings, our multi-order KANs consistently outperform MLPs, highlighting their superior representation capacity and effectiveness as a compelling alternative.

Table 3: Ablation study by replacing KAN with MLP.

Variants	ETTh1 (avg.)		ETTm1 (avg.)		Weather (avg.)	
	MSE	MAE	MSE	MAE	MSE	MAE
MLP_time	0.425	0.429	0.368	0.385	0.227	0.269
MLP_freq	0.424	0.427	0.367	0.384	0.226	0.267
MLP_time_freq	0.427	0.430	0.371	0.388	0.230	0.271
AdaKAN	0.421	0.426	0.364	0.382	0.224	0.266

Varying Look-back Window. In TSF, extending the look-back window theoretically allows the model to access richer historical context, which may improve prediction accuracy. However, this benefit is not consistently observed in most Transformer-based models (Zeng et al., 2023). As shown in Figure 3, we vary the look-back window in $\{48, 96, 192, 336, 512, 720\}$, with the prediction horizon fixed at 96. Results show that iTransformer, PatchTST, and the CNN-based TSLANet fail to benefit from longer windows and even sometimes suffer performance degradation, revealing their limited temporal modeling capability. In contrast, the MSE of TimeKAN and our AdaKAN steadily decreases with longer windows, and AdaKAN consistently outperforms TimeKAN across all settings. This highlights AdaKAN’s superior ability to capture both short- and long-term dependencies and to effectively extract information from extended historical sequences.

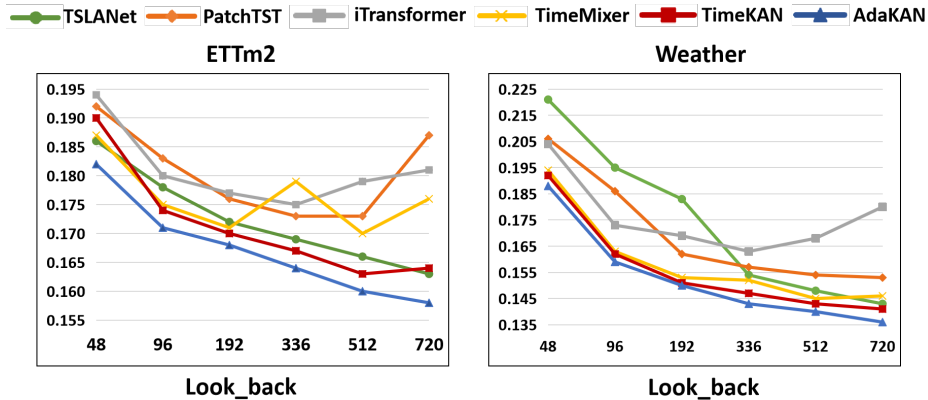


Figure 3: MSE on ETTm2 and Weather with varying look-back windows $\in \{48, 96, 192, 336, 512, 720\}$, and the prediction length is fixed to 96.

4.4 COMPLEXITY ANALYSIS

We evaluate the complexity of AdaKAN against other various TSF models—PatchTST, TimeKAN, Informer (Zhou et al., 2021), Fedformer (Zhou et al., 2022b), and FiLM (Zhou et al., 2022a) on Electricity, focusing on training parameters, MACs, and MSE under 96-step look-back and 720-step forecasting horizon, displayed in Figure 4. AdaKAN achieves superior predictive accuracy with substantially lower computational cost. While Transformer, Informer, Autoformer, FEDformer, and FiLM require 13.61M–20.68M parameters and 3.93G–5.97G MACs, their MSEs remain higher than AdaKAN. In contrast, AdaKAN matches the parameter scale of lightweight PatchTST and TimeKAN, yet reduces MACs by 85.94% and 80.64%, and improves MSE by 8.84% and 3.81%, confirming AdaKAN’s efficiency in TSF. These gains stem from AdaKAN’s lightweight design: grouped convolutions, and the multi-order KAN structure. Thus, we cannot overlook that AdaKAN achieves excellent performance while requiring minimal computational overhead.

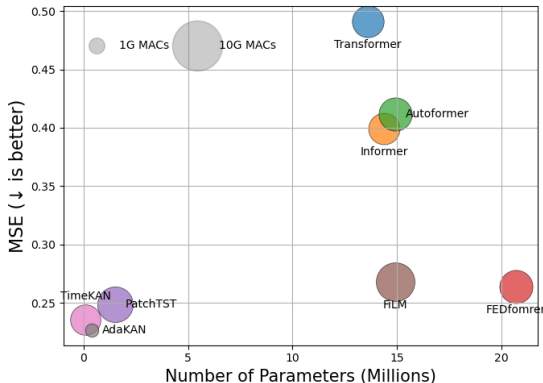


Figure 4: AdaKAN vs. baselines in terms of parameters and MACs count against MSE on Electricity with 96 look-back window and 720 forecasting length.

4.5 SENSITIVITY ANALYSIS

To obtain the optimal performance, we conducted a sensitivity analysis on ETTh1 and Weather, focusing on two key parameters: mask_ratio and dropout. As shown in Table 4, AdaKAN performs best when mask_rate is set to 0.25. A proper mask_rate value can ensure accurate learning representations while improving its robustness to adapt to unstable or data loss scenarios in reality. Dropout is used to prevent overfitting by randomly discarding a portion of neurons during training, making the network independent of any set of features. Table 5 displays that a dropout = 0.3 yields the best results on both ETTh1 and Weather. This indicates that appropriate dropout not only enhances generalization to unseen data, but also reduces the impact of noisy inputs and complex internal dependencies.

Table 4: Sensitivity experiments of mask_rate

mask_rate	p=0.01	p=0.1	p=0.2	p=0.25	p=0.4	p=0.5
ETTh1	0.382	0.379	0.376	0.374	0.377	0.380
Weather	0.162	0.159	0.153	0.150	0.156	0.164

Table 5: Sensitivity experiments of dropout

dropout	p=0.01	p=0.1	p=0.2	p=0.3	p=0.5	p=0.6
ETTh1	0.401	0.389	0.383	0.374	0.379	0.386
Weather	0.168	0.161	0.155	0.150	0.157	0.172

5 CONCLUSION

We present AdaKAN, a novel lightweight framework for TSF that seamlessly integrates dual KANs with an adaptive spectral decomposition module. AdaKAN employs three learnable filters—low-pass, band-pass, and high-pass—to adaptively decompose input sequences into low-, mid-, and high-frequency components based on trainable thresholds. To effectively learn and represent these specific components, we introduce a soft weighted multi-order mechanism in both Fourier KAN and Temporal KAN, which automatically learns weight assignments based on spectral features and further utilizes the high flexibility of KAN to capture long-term trends, periodic patterns, and short-term fluctuations in frequency and time domains. Extensive experiments on real-world datasets demonstrate that AdaKAN achieves SOTA forecasting performance and lightweight computational overhead. AdaKAN has opened up new avenues for TSF as a fundamental model.

REFERENCES

- 486
487
488 Guoguo Ai, Guansong Pang, Hezhe Qiao, Yuan Gao, and Hui Yan. Grokformer: Graph fourier
489 kolmogorov-arnold transformers. *In International Conference on Machine Learning (ICML)*,
490 2025.
- 491 Amr Alkhatib, Roman Bresson, Henrik Boström, and Michalis Vazirgiannis. Prediction via shapley
492 value regression. *In International Conference on Machine Learning (ICML)*, 2025.
- 493
494 Defu Cao, Furong Jia, Sercan O Arik, Tomas Pfister, Yixiang Zheng, Wen Ye, and Yan Liu. Tempo:
495 Prompt-based generative pre-trained transformer for time series forecasting. *The Eleventh Inter-*
496 *national Conference on Learning Representations (ICLR)*, 2024.
- 497 Cristian Challu, Kin G Olivares, Boris N Oreshkin, Federico Garza Ramirez, Max Mergenthaler
498 Canseco, and Artur Dubrawski. Nhits: Neural hierarchical interpolation for time series forecast-
499 ing. In *Proceedings of the AAAI conference on artificial intelligence*, volume 37, pp. 6989–6997,
500 2023.
- 501 Zechuan Chen, Tianming Sha, Ziyi Tang, and Keze Wang. Tskan: A transparent architecture for
502 improving the interpretability of multivariate time series forecasting. In *ICLR 2025 Workshop on*
503 *Navigating and Addressing Data Problems for Foundation Models*, 2025.
- 504
505 Emadeldeen Eldele, Mohamed Ragab, Zhenghua Chen, Min Wu, and Xiaoli Li. Tslanet: Rethinking
506 transformers for time series representation learning. *In International Conference on Machine*
507 *Learning (ICML)*, 2024.
- 508
509 Remi Genet and Hugo Inzirillo. Tkan: Temporal kolmogorov-arnold networks. *arXiv preprint*
510 *arXiv:2405.07344*, 2024.
- 511 Xiao Han, Xinfeng Zhang, Yiling Wu, Zhenduo Zhang, and Zhe Wu. Kan4tsf: Are kan and kan-
512 based models effective for time series forecasting? *arXiv preprint arXiv:2408.11306*, 2024a.
- 513
514 Xiao Han, Xinfeng Zhang, Yiling Wu, Zhenduo Zhang, and Zhe Wu. Are kans effective for multi-
515 variate time series forecasting. *arXiv preprint arXiv:2408.11306*, 2024b.
- 516
517 Hongbin Huang, Minghua Chen, and Xiao Qiao. Generative learning for financial time series with
518 irregular and scale-invariant patterns. In *The Twelfth International Conference on Learning Rep-*
resentations (ICLR), 2024.
- 519
520 Songtao Huang, Zhen Zhao, Can Li, and LEI BAI. TimeKAN: KAN-based frequency decomposi-
521 tion learning architecture for long-term time series forecasting. In *The Thirteenth International*
522 *Conference on Learning Representations (ICLR)*, 2025. URL <https://openreview.net/forum?id=wTLc79YNbh>.
- 523
524 Ming Jin, Huan Yee Koh, Qingsong Wen, Daniele Zambon, Cesare Alippi, Geoffrey I Webb, Irwin
525 King, and Shirui Pan. A survey on graph neural networks for time series: Forecasting, classifi-
526 cation, imputation, and anomaly detection. *IEEE Transactions on Pattern Analysis and Machine*
527 *Intelligence*, 2024.
- 528
529 Lutz Kilian and Helmut Lütkepohl. *Structural vector autoregressive analysis*. Cambridge University
530 Press, 2017.
- 531
532 Taesung Kim, Jinhee Kim, Yunwon Tae, Cheonbok Park, Jang-Ho Choi, and Jaegul Choo. Re-
533 versible instance normalization for accurate time-series forecasting against distribution shift. In
International Conference on Learning Representations, 2021.
- 534
535 Andrey N. Kolmogorov. On the representation of continuous functions of many variables by su-
536 perposition of continuous functions of one variable and addition. *Translations of the American*
537 *Mathematical Society*, 2(28):55–59, 1963.
- 538
539 Chenxin Li, Xinyu Liu, Wuyang Li, Cheng Wang, Hengyu Liu, Yifan Liu, Zhen Chen, and Yix-
uan Yuan. U-kan makes strong backbone for medical image segmentation and generation. In
Proceedings of the AAAI Conference on Artificial Intelligence, volume 39, pp. 4652–4660, 2025.

- 540 Zhe Li, Shiyi Qi, Yiduo Li, and Zenglin Xu. Revisiting long-term time series forecasting: An inves-
541 tigation on linear mapping. *arXiv preprint arXiv:2305.10721*, 2023.
- 542 Bryan Lim and Stefan Zohren. Time-series forecasting with deep learning: a survey. *Philosophical*
543 *Transactions of the Royal Society A*, 379(2194):20200209, 2021.
- 544 Minhao Liu, Ailing Zeng, Muxi Chen, Zhijian Xu, Qiuxia Lai, Lingna Ma, and Qiang Xu. Scinet:
545 Time series modeling and forecasting with sample convolution and interaction. *Advances in*
546 *Neural Information Processing Systems*, 35:5816–5828, 2022.
- 547 Yong Liu, Tengge Hu, Haoran Zhang, Haixu Wu, Shiyu Wang, Lintao Ma, and Mingsheng Long.
548 itransformer: Inverted transformers are effective for time series forecasting. In *International Con-*
549 *ference on Learning Representations (ICLR), 2024*, 2024a.
- 550 Ziming Liu, Yixuan Wang, Sachin Vaidya, Fabian Ruehle, James Halverson, Marin Soljačić,
551 Thomas Y Hou, and Max Tegmark. Kan: Kolmogorov-arnold networks. *arXiv preprint*
552 *arXiv:2404.19756*, 2024b.
- 553 Yuqi Nie, Nam H Nguyen, Phanwadee Sinthong, and Jayant Kalagnanam. A time series is worth
554 64 words: Long-term forecasting with transformers. In *International Conference on Learning*
555 *Representations (ICLR), 2023*, 2023.
- 556 Boris N Oreshkin, Dmitri Carпов, Nicolas Chapados, and Yoshua Bengio. N-beats: Neural basis ex-
557 pansion analysis for interpretable time series forecasting. *The Eleventh International Conference*
558 *on Learning Representations (ICLR)*, 2020.
- 559 Feng Shibo, Peilin Zhao, Liu Liu, Pengcheng Wu, and Zhiqi Shen. Hdt: Hierarchical discrete trans-
560 former for multivariate time series forecasting. In *Proceedings of the AAAI Conference on Artificial*
561 *Intelligence*, volume 39, pp. 746–754, 2025.
- 562 Sidharth SS, Keerthana AR, Anas KP, et al. Chebyshev polynomial-based kolmogorov-arnold
563 networks: An efficient architecture for nonlinear function approximation. *arXiv preprint*
564 *arXiv:2405.07200*, 2024.
- 565 CJ Vaca-Rubio, L Blanco, R Pereira, and M Caus. Kolmogorov-arnold networks (kans) for time
566 series analysis. arxiv 2024. *arXiv preprint arXiv:2405.08790*, 2024.
- 567 Huiqiang Wang, Jian Peng, Feihu Huang, Jince Wang, Junhui Chen, and Yifei Xiao. Micn: Multi-
568 scale local and global context modeling for long-term series forecasting. In *The eleventh interna-*
569 *tional conference on learning representations (ICLR)*, 2023.
- 570 Shiyu Wang, Haixu Wu, Xiaoming Shi, Tengge Hu, Huakun Luo, Lintao Ma, James Y Zhang, and
571 Jun Zhou. Timemixer: Decomposable multiscale mixing for time series forecasting. *The Twelfth*
572 *International Conference on Learning Representations (ICLR)*, 2024a.
- 573 Yihe Wang, Yu Han, Haishuai Wang, and Xiang Zhang. Contrast everything: A hierarchical con-
574 trastive framework for medical time-series. *Advances in Neural Information Processing Systems*,
575 36, 2024b.
- 576 Haixu Wu, Jiehui Xu, Jianmin Wang, and Mingsheng Long. Autoformer: Decomposition trans-
577 formers with auto-correlation for long-term series forecasting. *Advances in neural information*
578 *processing systems*, 34:22419–22430, 2021.
- 579 Haixu Wu, Tengge Hu, Yong Liu, Hang Zhou, Jianmin Wang, and Mingsheng Long. Timesnet:
580 Temporal 2d-variation modeling for general time series analysis. *The Eleventh International*
581 *Conference on Learning Representations (ICLR)*, 2023.
- 582 Yuhan Wu, Xiyu Meng, Yang He, Junru Zhang, Haowen Zhang, Yabo Dong, and Dongming Lu.
583 Multi-view self-supervised contrastive learning for multivariate time series. In *Proceedings of the*
584 *32nd ACM International Conference on Multimedia*, pp. 9582–9590, 2024a.
- 585 Yuhan Wu, Xiyu Meng, Junru Zhang, Yang He, Joseph A Romo, Yabo Dong, and Dongming Lu.
586 Effective lstms with seasonal-trend decomposition and adaptive learning and niching-based back-
587 tracking search algorithm for time series forecasting. *Expert Systems with Applications*, 236:
588 121202, 2024b.

- 594 Yuhan Wu, Xiyu Meng, Huajin Hu, Junru Zhang, Yabo Dong, and Dongming Lu. Affirm: Interactive
595 mamba with adaptive fourier filters for long-term time series forecasting. In *Proceedings of the*
596 *AAAI Conference on Artificial Intelligence*, volume 39, pp. 21599–21607, 2025.
- 597 Zexue Wu, Yifeng Gong, and Aoqian Zhang. Dtmamba: Dual twin mamba for time series forecast-
598 ing. *arXiv preprint arXiv:2405.07022*, 2024c.
- 600 Kunpeng Xu, Lifei Chen, and Shengrui Wang. Kolmogorov-arnold networks for time series: Bridg-
601 ing predictive power and interpretability. *arXiv preprint arXiv:2406.02496*, 2024a.
- 602 Zhijian Xu, Ailing Zeng, and Qiang Xu. Fits: Modeling time series with 10k parameters. In *Inter-
603 national Conference on Learning Representations (ICLR)*, 2024, 2024b.
- 605 Kun Yi, Qi Zhang, Wei Fan, Shoujin Wang, Pengyang Wang, Hui He, Ning An, Defu Lian, Long-
606 bing Cao, and Zhendong Niu. Frequency-domain mlps are more effective learners in time series
607 forecasting. *Advances in Neural Information Processing Systems*, 36:76656–76679, 2024.
- 608 Ailing Zeng, Muxi Chen, Lei Zhang, and Qiang Xu. Are transformers effective for time series
609 forecasting? In *Proceedings of the AAAI conference on artificial intelligence*, volume 37, pp.
610 11121–11128, 2023.
- 612 Chaolv Zeng, Zhanyu Liu, Guanjie Zheng, and Linghe Kong. C-mamba: Channel correla-
613 tion enhanced state space models for multivariate time series forecasting. *arXiv preprint*
614 *arXiv:2406.05316*, 2024.
- 615 G Peter Zhang. Time series forecasting using a hybrid arima and neural network model. *Neurocom-
616 puting*, 50:159–175, 2003.
- 617 Yunhao Zhang and Junchi Yan. Crossformer: Transformer utilizing cross-dimension dependency
618 for multivariate time series forecasting. In *The eleventh international conference on learning*
619 *representations (ICLR)*, 2023.
- 621 Haoyi Zhou, Shanghang Zhang, Jieqi Peng, Shuai Zhang, Jianxin Li, Hui Xiong, and Wancai Zhang.
622 Informer: Beyond efficient transformer for long sequence time-series forecasting. In *Proceedings*
623 *of the AAAI conference on artificial intelligence*, volume 35, pp. 11106–11115, 2021.
- 624 Tian Zhou, Ziqing Ma, Qingsong Wen, Liang Sun, Tao Yao, Wotao Yin, Rong Jin, et al. Film:
625 Frequency improved legendre memory model for long-term time series forecasting. *Advances in*
626 *neural information processing systems*, 35:12677–12690, 2022a.
- 627 Tian Zhou, Ziqing Ma, Qingsong Wen, Xue Wang, Liang Sun, and Rong Jin. Fedformer: Frequency
628 enhanced decomposed transformer for long-term series forecasting. In *International conference*
629 *on machine learning (ICML)*, pp. 27268–27286. PMLR, 2022b.
- 630 Tian Zhou, Peisong Niu, Liang Sun, Rong Jin, et al. One fits all: Power general time series analysis
631 by pretrained lm. *Advances in neural information processing systems*, 36:43322–43355, 2023.
- 632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647

A RELATED WORK

A.1 TIME SERIES FORECASTING

Time series forecasting aims to predict future values based on historical observations. Traditional methods such as VAR (Kilian & Lütkepohl, 2017) and ARIMA (Zhang, 2003) offer interpretability but often lack accuracy. In contrast, deep learning methods have gained traction for their ability to model nonlinear and complex dependencies, including CNNs, RNNs, Transformers, and MLPs (Lim & Zohren, 2021). CNNs, such as SCINet (Liu et al., 2022) focus on extracting local temporal patterns via convolution but struggle with long-range dependencies. RNNs like LSTM and GRU improve sequence modeling but are limited by sequential computation and slow training. Transformers like Informer (Zhou et al., 2021) and Crossformer (Zhang & Yan, 2023), have become dominant due to their global attention mechanisms, with variants like PatchTST (Nie et al., 2023) and iTransformer (Liu et al., 2024a) further enhancing performance. However, their high parameter counts and memory usage remain challenges. Recently, lightweight MLP-based models have shown superior accuracy and efficiency. For example, DLinear (Zeng et al., 2023) uses trend-seasonal decomposition, and FITS (Xu et al., 2024b) applies frequency-domain linear transformations. Building on this, we explore the potential of the KAN and propose an adaptive frequency decomposition module to better utilize frequency patterns for time series forecasting.

B EXPERIMENTS

Datasets. To verify the model’s performance, we evaluate the performance of our proposed Flames on 8 popular, well-established benchmarks: 4 ETT datasets (ETTh1, ETTh2, ETTm1, ETTm2), Electricity, Exchange, Traffic, and Weather. The datasets have been extensively adopted for benchmarking LSTF models and are publicly available on (Wu et al., 2021), covering domains such as electricity, transportation, energy, weather, and economy. Notably, we would like to highlight 3 large datasets: Electricity, Traffic, and Weather. They have more dimensions, so the results will be more stable and less prone to overfitting than other smaller datasets. (1) **ETT**¹ comprises two granularities collected from different regions in China, containing two hourly-level datasets (ETTh1, ETTh2) and two 15-minute-level datasets (ETTm1, ETTm2). Each dataset includes six power load features and a target “oil temperature” variable from July 2016 to July 2018. (2) **Electricity** tracks the power electricity consumption of 321 clients, converted to hourly measurements. This provides insights into usage patterns and enables demand forecasting, which is crucial for optimizing power generation and distribution. (3) **Exchange Rate** features daily exchange rates of different currencies from 8 different countries against the US dollar, allowing the model to predict currency fluctuations based on historical data. (4) **Traffic** records the road occupancy rate of different sensors on the San Francisco highway, including the traffic volume of 94 interstate westbound traffic in the Twin Cities metropolitan area, which can predict the traffic flow pattern, crucial to congestion management and urban planning. (5) **Weather**² contains 35,136 data points from a PV power station in northwest China, 15-minute intervals from January 1, 2020, to December 31, 2020. Besides, we will publicize this data as one benchmark. The statistics details are highlighted in Table 6.

Implementation Details. We conduct all experiments on an NVIDIA GeForce RTX 4090 Ti GPU with 64-bit Linux 5.15.0-56-generic, with 60/20/20 train/validation/test split for ETTs, and 70/10/20 for other datasets. Similar to (Zhou et al., 2023; Eldele et al., 2024) settings, we use the look-back window of 336 for ETTs, 96 for Exchange and Electricity, 512 for Traffic and Weather. All datasets are normalized during training (Kim et al., 2021). For baselines, we report their best results if setups match; otherwise, we rerun their code.

B.1 MORE COMPARATIVE BASELINES

We compare our model with other common time series forecasting baselines, i.e. 1) Transformer-based: Autoformer (Wu et al., 2021) , FEDformer (Zhou et al., 2022b); 2) CNN-based: SCINet (Liu et al., 2022), MICN (Wang et al., 2023); 3) Linear-based: Rlinear (Li et al., 2023), TiDE(?);

¹<https://github.com/zhouhaoyi/ETDataset>

²<https://www.ncei.noaa.gov/data/local-climatological-data/>

Dataset	Dim	Dataset Size	Frequency	Domain
ETTh1	7	(8545, 2881, 2881)	Hourly	Temperature
ETTh2	7	(8545, 2881, 2881)	Hourly	Temperature
ETTm1	7	(34465, 11521, 11521)	15min	Temperature
ETTm2	7	(34465, 11521, 11521)	15min	Temperature
Exchange	8	(5120, 665, 1422)	Daily	Economy
Electricity	321	(18317, 2633, 5261)	Hourly	Electricity
Traffic	862	(12185, 1757, 3509)	Hourly	Transportation
Weather	21	(36792, 5271, 10540)	10min	Weather

Table 6: Dataset detailed descriptions. Dim denotes the variate number of each dataset. Frequency represents the sampling interval of time points. The dataset size is organized in (Train, Validation, Test).

4) Frequency-based: FiLM (Zhou et al., 2022a); 5) LLM-based: Tempo (Cao et al., 2024). Table 7 shows that our proposed Flames still exhibits significant advantages in all datasets and all lengths. For instance, on Weather, the MSE of our method decreases by 20.6%, 17.6%, 17.3%, 16.4%, 17.3%, 23.3%, 27.5%, and 33.7%. Similarly, the MAE exceeds other models by 15.8%, 8.6%, 16.9%, 17.1%, 8.3%, 26.7%, 26.1%, and 30.4%, which indicates the effectiveness and superiority of our framework in time series forecasting. Overall, our Affirm wins in 64 out of 80 results, and 8 second-best in the table.

Method	AdaKAN		Tempo		Rlinear		TIDE		MICN		FILM		SCINet		FEDformer		Autoformer			
	(Ours)		2024		2023		2023		2023		2022		2022		2022		2021			
Metrics	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE		
ETTh1	96	0.374	0.393	0.400	0.406	0.386	<u>0.395</u>	0.479	0.464	0.426	0.446	0.438	0.433	0.654	0.599	<u>0.376</u>	0.419	0.449	0.459	
	192	0.414	<u>0.424</u>	0.426	0.421	0.437	0.424	0.525	0.492	0.454	0.464	0.494	0.466	0.719	0.631	<u>0.420</u>	0.448	0.500	0.482	
	336	<u>0.443</u>	0.432	0.441	0.430	0.479	<u>0.446</u>	0.565	0.515	0.493	0.487	0.547	0.495	0.778	0.659	0.459	0.465	0.521	0.496	
	720	<u>0.452</u>	0.455	0.443	0.451	0.481	<u>0.470</u>	0.594	0.558	0.526	0.526	0.586	0.538	0.836	0.699	0.506	0.507	0.514	0.512	
	Avg.	0.421	0.426	<u>0.428</u>	<u>0.427</u>	0.446	0.434	0.541	0.507	0.475	0.480	0.516	0.483	0.747	0.647	0.440	0.460	0.496	0.487	
ETTh2	96	0.285	0.334	0.301	0.353	<u>0.288</u>	<u>0.338</u>	0.400	0.440	0.372	0.424	0.322	0.364	0.707	0.621	0.358	0.397	0.346	0.388	
	192	<u>0.370</u>	0.391	0.355	0.389	0.374	<u>0.390</u>	0.528	0.509	0.492	0.492	0.405	0.414	0.860	0.689	0.429	0.439	0.456	0.452	
	336	0.356	0.395	<u>0.379</u>	<u>0.408</u>	0.415	0.426	0.643	0.571	0.607	0.555	0.435	0.445	1.000	0.744	0.496	0.487	0.482	0.486	
	720	0.408	0.433	<u>0.409</u>	<u>0.440</u>	0.420	0.440	0.420	0.440	0.874	0.679	0.824	0.655	0.445	0.457	1.249	0.838	0.463	0.474	0.515
	Avg.	0.354	0.388	<u>0.361</u>	<u>0.398</u>	0.374	0.399	0.611	0.550	0.574	0.531	0.402	0.420	0.954	0.723	0.437	0.449	0.450	0.459	
ETTm1	96	0.312	0.355	0.438	0.424	0.355	0.376	0.364	0.387	0.365	0.387	<u>0.353</u>	<u>0.370</u>	0.418	0.438	0.379	0.419	0.505	0.475	
	192	0.340	0.361	0.461	0.432	0.391	0.392	0.398	0.404	0.403	0.408	<u>0.389</u>	<u>0.387</u>	0.439	0.450	0.426	0.441	0.553	0.496	
	336	0.371	0.386	0.515	0.467	0.424	0.415	0.428	0.425	0.436	0.431	<u>0.421</u>	<u>0.408</u>	0.490	0.485	0.445	0.459	0.621	0.537	
	720	0.431	0.425	0.591	0.509	0.487	0.450	0.487	0.461	0.489	0.462	<u>0.481</u>	<u>0.441</u>	0.595	0.550	0.543	0.490	0.671	0.561	
	Avg.	0.364	0.382	0.501	0.458	0.414	0.408	0.419	0.419	0.423	0.422	<u>0.412</u>	<u>0.402</u>	0.485	0.481	0.448	0.452	0.588	0.517	
ETTm2	96	0.169	0.253	0.185	0.267	<u>0.182</u>	<u>0.265</u>	0.207	0.305	0.197	0.296	0.183	0.266	0.286	0.377	0.203	0.287	0.255	0.339	
	192	0.232	0.298	<u>0.243</u>	<u>0.304</u>	0.246	0.304	0.290	0.364	0.284	0.361	0.248	0.305	0.399	0.445	0.269	0.328	0.281	0.340	
	336	0.277	0.335	0.309	0.345	<u>0.307</u>	<u>0.342</u>	0.377	0.422	0.381	0.429	0.309	0.343	0.637	0.591	0.325	0.366	0.399	0.372	
	720	0.372	0.383	<u>0.386</u>	<u>0.395</u>	0.407	0.398	0.558	0.524	0.549	0.522	0.410	0.400	0.960	0.735	0.421	0.415	0.433	0.432	
	Avg.	0.262	0.317	<u>0.280</u>	0.328	0.286	<u>0.327</u>	0.358	0.404	0.353	0.402	0.288	0.328	0.571	0.537	0.305	0.349	0.327	0.371	
Electricity	96	0.164	0.261	<u>0.178</u>	0.276	0.201	0.281	0.237	0.329	0.180	0.293	0.198	<u>0.274</u>	0.247	0.345	0.193	0.308	0.201	0.317	
	192	0.178	0.273	0.198	0.293	0.201	0.283	0.236	0.330	<u>0.189</u>	0.302	0.198	<u>0.278</u>	0.257	0.355	0.201	0.315	0.222	0.334	
	336	0.192	0.282	0.209	0.309	0.215	<u>0.298</u>	0.249	0.344	<u>0.198</u>	0.312	0.217	0.300	0.269	0.369	0.214	0.329	0.231	0.338	
	720	<u>0.227</u>	0.316	0.279	0.355	0.257	<u>0.331</u>	0.284	0.373	0.217	0.330	0.278	0.356	0.299	0.390	0.246	0.355	0.254	0.361	
	Avg.	0.190	0.283	0.216	0.308	0.219	<u>0.298</u>	0.251	0.344	0.196	0.309	0.223	0.302	0.268	0.365	0.214	0.327	0.227	0.338	
Exchange	96	0.083	0.203	0.097	0.211	0.093	0.217	0.094	0.218	0.097	0.228	0.098	0.256	0.267	0.396	0.148	0.278	0.197	0.323	
	192	0.170	0.295	0.193	<u>0.304</u>	<u>0.184</u>	0.307	0.184	0.307	0.288	0.312	0.282	0.335	0.351	0.459	0.171	0.315	0.300	0.369	
	336	0.342	0.424	0.362	0.438	0.351	0.432	<u>0.349</u>	0.431	0.387	0.442	0.425	0.487	1.423	0.853	0.460	<u>0.427</u>	0.509	0.524	
	720	0.705	0.608	<u>0.793</u>	<u>0.689</u>	0.886	0.714	<u>0.852</u>	0.698	0.904	0.716	0.912	0.732	1.058	0.797	1.195	0.695	1.447	0.941	
	Avg.	0.325	0.382	<u>0.361</u>	<u>0.411</u>	0.379	0.418	0.370	0.413	0.419	0.425	0.429	0.453	0.750	0.626	0.519	0.429	0.613	0.539	
Traffic	96	0.527	0.338	0.476	0.343	0.649	0.389	0.805	0.493	0.519	0.309	<u>0.514</u>	0.304	0.788	0.499	0.587	0.366	0.613	0.388	
	192	0.535	0.344	0.496	0.355	0.601	0.366	0.756	0.474	0.537	0.315	<u>0.528</u>	<u>0.329</u>	0.789	0.505	0.604	0.373	0.616	0.382	
	336	0.541	0.349	0.503	0.356	0.609	0.369	0.762	0.477	<u>0.534</u>	0.313	0.537	<u>0.334</u>	0.797	0.508	0.621	0.383	0.622	0.337	
	720	0.560	<u>0.357</u>	0.538	0.376	0.647	0.387	0.719	0.449	<u>0.577</u>	0.325	0.622	0.363	0.841	0.523	0.626	0.382	0.660	0.408	
	Avg.	<u>0.540</u>	0.347	0.503	0.358	0.627	0.378	0.760	0.473	0.542	0.316	0.550	<u>0.333</u>	0.804	0.509	0.610	0.376	0.628	0.379	
Weather	96	0.150	0.201	0.211	0.254	<u>0.192</u>	<u>0.232</u>	0.202	0.261	0.198	0.261	0.195	0.236	0.221	0.306	0.217	0.296	0.266	0.336	
	192	0.192	0.241	0.254	0.298	0.240	<u>0.271</u>	0.242	0.298	<u>0.239</u>	0.299	0.239	0.271	0.261	0.340	0.276	0.336	0.307	0.367	
	336	0.242	0.286	0.292	0.332	0.292	0.307	0.287	0.335	<u>0.285</u>	0.336	0.289	<u>0.306</u>	0.309	0.378	0.339	0.380	0.359	0.395	
	720	0.312	0.338	0.370	0.379	0.364	0.353	<u>0.351</u>	0.386	0.351	0.388	0.360	<u>0.351</u>	0.377	0.427	0.403	0.428	0.419	0.428	
	Avg.	0.224	0.266	0.282	0.316	0.272	0.291	0.271	0.320	<u>0.268</u>	0.321	0.271	<u>0.290</u>	0.292	0.363	0.309	0.360	0.338	0.382	
1 st Count	64		10		0		0		5		1		0		0		0			

Table 7: Multivariate long-term series forecasting results on the same input length=96, and different prediction lengths $\in \{96, 192, 336, 720\}$. A lower value indicates better performance. **Bold**: best, underlined: second best. 1st Count: the number of the best results. Avg. means the average results from all four prediction lengths.

756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809

B.2 ABLATION STUDY

Ablation study of each component: (1) "w/o Group conv": remove Group convolution. (2) "w/o Cross-att": remove Cross-attention. (3) "w/o Soft-order": maintain a fixed low order 2. (4) "w/o ASFM": remove Adaptive Spectral Filter Module. (5) "w/o FourierKAN" and (6) "w/o TemporalKAN": remove FourierKAN and TemporalKAN. (7) "w/o T-F KAN": remove Temporal-Frequency KANs.

REFERENCES

- Guoguo Ai, Guansong Pang, Hezhe Qiao, Yuan Gao, and Hui Yan. Grokformer: Graph fourier kolmogorov-arnold transformers. *In International Conference on Machine Learning (ICML)*, 2025.
- Amr Alkhatib, Roman Bresson, Henrik Boström, and Michalis Vazirgiannis. Prediction via shapley value regression. *In International Conference on Machine Learning (ICML)*, 2025.
- Defu Cao, Furong Jia, Serkan O Arik, Tomas Pfister, Yixiang Zheng, Wen Ye, and Yan Liu. Tempo: Prompt-based generative pre-trained transformer for time series forecasting. *The Eleventh International Conference on Learning Representations (ICLR)*, 2024.
- Cristian Challu, Kin G Olivares, Boris N Oreshkin, Federico Garza Ramirez, Max Mergenthaler Canseco, and Artur Dubrawski. Nhits: Neural hierarchical interpolation for time series forecasting. *In Proceedings of the AAAI conference on artificial intelligence*, volume 37, pp. 6989–6997, 2023.
- Zechuan Chen, Tianming Sha, Ziyi Tang, and Keze Wang. Tskan: A transparent architecture for improving the interpretability of multivariate time series forecasting. *In ICLR 2025 Workshop on Navigating and Addressing Data Problems for Foundation Models*, 2025.
- Emadeldeen Eldele, Mohamed Ragab, Zhenghua Chen, Min Wu, and Xiaoli Li. Tslanet: Rethinking transformers for time series representation learning. *In International Conference on Machine Learning (ICML)*, 2024.
- Remi Genet and Hugo Inzirillo. Tkan: Temporal kolmogorov-arnold networks. *arXiv preprint arXiv:2405.07344*, 2024.
- Xiao Han, Xinfeng Zhang, Yiling Wu, Zhenduo Zhang, and Zhe Wu. Kan4tsf: Are kan and kan-based models effective for time series forecasting? *arXiv preprint arXiv:2408.11306*, 2024a.
- Xiao Han, Xinfeng Zhang, Yiling Wu, Zhenduo Zhang, and Zhe Wu. Are kans effective for multivariate time series forecasting. *arXiv preprint arXiv:2408.11306*, 2024b.
- Hongbin Huang, Minghua Chen, and Xiao Qiao. Generative learning for financial time series with irregular and scale-invariant patterns. *In The Twelfth International Conference on Learning Representations (ICLR)*, 2024.
- Songtao Huang, Zhen Zhao, Can Li, and LEI BAI. TimeKAN: KAN-based frequency decomposition learning architecture for long-term time series forecasting. *In The Thirteenth International Conference on Learning Representations (ICLR)*, 2025. URL <https://openreview.net/forum?id=wTlc79YNbh>.
- Ming Jin, Huan Yee Koh, Qingsong Wen, Daniele Zambon, Cesare Alippi, Geoffrey I Webb, Irwin King, and Shirui Pan. A survey on graph neural networks for time series: Forecasting, classification, imputation, and anomaly detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2024.
- Lutz Kilian and Helmut Lütkepohl. *Structural vector autoregressive analysis*. Cambridge University Press, 2017.
- Taesung Kim, Jinhee Kim, Yunwon Tae, Cheonbok Park, Jang-Ho Choi, and Jaegul Choo. Reversible instance normalization for accurate time-series forecasting against distribution shift. *In International Conference on Learning Representations*, 2021.

- 810 Andrey N. Kolmogorov. On the representation of continuous functions of many variables by superposition of continuous functions of one variable and addition. *Translations of the American Mathematical Society*, 2(28):55–59, 1963.
- 811
812
813
- 814 Chenxin Li, Xinyu Liu, Wuyang Li, Cheng Wang, Hengyu Liu, Yifan Liu, Zhen Chen, and Yixuan Yuan. U-kan makes strong backbone for medical image segmentation and generation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 4652–4660, 2025.
- 815
816
- 817 Zhe Li, Shiyi Qi, Yiduo Li, and Zenglin Xu. Revisiting long-term time series forecasting: An investigation on linear mapping. *arXiv preprint arXiv:2305.10721*, 2023.
- 818
819
- 820 Bryan Lim and Stefan Zohren. Time-series forecasting with deep learning: a survey. *Philosophical Transactions of the Royal Society A*, 379(2194):20200209, 2021.
- 821
- 822 Minhao Liu, Ailing Zeng, Muxi Chen, Zhijian Xu, Qiuxia Lai, Lingna Ma, and Qiang Xu. Scinet: Time series modeling and forecasting with sample convolution and interaction. *Advances in Neural Information Processing Systems*, 35:5816–5828, 2022.
- 823
824
- 825 Yong Liu, Tengge Hu, Haoran Zhang, Haixu Wu, Shiyu Wang, Lintao Ma, and Mingsheng Long. itransformer: Inverted transformers are effective for time series forecasting. In *International Conference on Learning Representations (ICLR)*, 2024, 2024a.
- 826
827
828
- 829 Ziming Liu, Yixuan Wang, Sachin Vaidya, Fabian Ruehle, James Halverson, Marin Soljačić, Thomas Y Hou, and Max Tegmark. Kan: Kolmogorov-arnold networks. *arXiv preprint arXiv:2404.19756*, 2024b.
- 830
831
- 832 Yuqi Nie, Nam H Nguyen, Phanwadee Sinthong, and Jayant Kalagnanam. A time series is worth 64 words: Long-term forecasting with transformers. In *International Conference on Learning Representations (ICLR)*, 2023, 2023.
- 833
834
835
- 836 Boris N Oreshkin, Dmitri Carпов, Nicolas Chapados, and Yoshua Bengio. N-beats: Neural basis expansion analysis for interpretable time series forecasting. *The Eleventh International Conference on Learning Representations (ICLR)*, 2020.
- 837
838
- 839 Feng Shibo, Peilin Zhao, Liu Liu, Pengcheng Wu, and Zhiqi Shen. Hdt: Hierarchical discrete transformer for multivariate time series forecasting. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 746–754, 2025.
- 840
841
- 842 Sidharth SS, Keerthana AR, Anas KP, et al. Chebyshev polynomial-based kolmogorov-arnold networks: An efficient architecture for nonlinear function approximation. *arXiv preprint arXiv:2405.07200*, 2024.
- 843
844
845
- 846 CJ Vaca-Rubio, L Blanco, R Pereira, and M Caus. Kolmogorov-arnold networks (kans) for time series analysis. arxiv 2024. *arXiv preprint arXiv:2405.08790*, 2024.
- 847
- 848 Huiqiang Wang, Jian Peng, Feihu Huang, Jince Wang, Junhui Chen, and Yifei Xiao. Micn: Multi-scale local and global context modeling for long-term series forecasting. In *The eleventh international conference on learning representations (ICLR)*, 2023.
- 849
850
- 851 Shiyu Wang, Haixu Wu, Xiaoming Shi, Tengge Hu, Huakun Luo, Lintao Ma, James Y Zhang, and Jun Zhou. Timemixer: Decomposable multiscale mixing for time series forecasting. *The Twelfth International Conference on Learning Representations (ICLR)*, 2024a.
- 852
853
854
- 855 Yihe Wang, Yu Han, Haishuai Wang, and Xiang Zhang. Contrast everything: A hierarchical contrastive framework for medical time-series. *Advances in Neural Information Processing Systems*, 36, 2024b.
- 856
857
- 858 Haixu Wu, Jiehui Xu, Jianmin Wang, and Mingsheng Long. Autoformer: Decomposition transformers with auto-correlation for long-term series forecasting. *Advances in neural information processing systems*, 34:22419–22430, 2021.
- 859
860
861
- 862 Haixu Wu, Tengge Hu, Yong Liu, Hang Zhou, Jianmin Wang, and Mingsheng Long. Timesnet: Temporal 2d-variation modeling for general time series analysis. *The Eleventh International Conference on Learning Representations (ICLR)*, 2023.
- 863

- 864 Yuhan Wu, Xiyu Meng, Yang He, Junru Zhang, Haowen Zhang, Yabo Dong, and Dongming Lu.
865 Multi-view self-supervised contrastive learning for multivariate time series. In *Proceedings of the*
866 *32nd ACM International Conference on Multimedia*, pp. 9582–9590, 2024a.
- 867 Yuhan Wu, Xiyu Meng, Junru Zhang, Yang He, Joseph A Romo, Yabo Dong, and Dongming Lu.
868 Effective lstms with seasonal-trend decomposition and adaptive learning and niching-based back-
869 tracking search algorithm for time series forecasting. *Expert Systems with Applications*, 236:
870 121202, 2024b.
- 871 Yuhan Wu, Xiyu Meng, Huajin Hu, Junru Zhang, Yabo Dong, and Dongming Lu. Affirm: Interactive
872 mamba with adaptive fourier filters for long-term time series forecasting. In *Proceedings of the*
873 *AAAI Conference on Artificial Intelligence*, volume 39, pp. 21599–21607, 2025.
- 874 Zexue Wu, Yifeng Gong, and Aoqian Zhang. Dtmamba: Dual twin mamba for time series forecast-
875 ing. *arXiv preprint arXiv:2405.07022*, 2024c.
- 876 Kunpeng Xu, Lifei Chen, and Shengrui Wang. Kolmogorov-arnold networks for time series: Bridg-
877 ing predictive power and interpretability. *arXiv preprint arXiv:2406.02496*, 2024a.
- 878 Zhijian Xu, Ailing Zeng, and Qiang Xu. Fits: Modeling time series with 10k parameters. In *Inter-*
879 *national Conference on Learning Representations (ICLR), 2024*, 2024b.
- 880 Kun Yi, Qi Zhang, Wei Fan, Shoujin Wang, Pengyang Wang, Hui He, Ning An, Defu Lian, Long-
881 bing Cao, and Zhendong Niu. Frequency-domain mlps are more effective learners in time series
882 forecasting. *Advances in Neural Information Processing Systems*, 36:76656–76679, 2024.
- 883 Ailing Zeng, Muxi Chen, Lei Zhang, and Qiang Xu. Are transformers effective for time series
884 forecasting? In *Proceedings of the AAAI conference on artificial intelligence*, volume 37, pp.
885 11121–11128, 2023.
- 886 Chaolv Zeng, Zhanyu Liu, Guanjie Zheng, and Linghe Kong. C-mamba: Channel correla-
887 tion enhanced state space models for multivariate time series forecasting. *arXiv preprint*
888 *arXiv:2406.05316*, 2024.
- 889 G Peter Zhang. Time series forecasting using a hybrid arima and neural network model. *Neurocom-*
890 *puting*, 50:159–175, 2003.
- 891 Yunhao Zhang and Junchi Yan. Crossformer: Transformer utilizing cross-dimension dependency
892 for multivariate time series forecasting. In *The eleventh international conference on learning*
893 *representations (ICLR), 2023*.
- 894 Haoyi Zhou, Shanghang Zhang, Jieqi Peng, Shuai Zhang, Jianxin Li, Hui Xiong, and Wancai Zhang.
895 Informer: Beyond efficient transformer for long sequence time-series forecasting. In *Proceedings*
896 *of the AAAI conference on artificial intelligence*, volume 35, pp. 11106–11115, 2021.
- 897 Tian Zhou, Ziqing Ma, Qingsong Wen, Liang Sun, Tao Yao, Wotao Yin, Rong Jin, et al. Film:
898 Frequency improved legendre memory model for long-term time series forecasting. *Advances in*
899 *neural information processing systems*, 35:12677–12690, 2022a.
- 900 Tian Zhou, Ziqing Ma, Qingsong Wen, Xue Wang, Liang Sun, and Rong Jin. Fedformer: Frequency
901 enhanced decomposed transformer for long-term series forecasting. In *International conference*
902 *on machine learning (ICML)*, pp. 27268–27286. PMLR, 2022b.
- 903 Tian Zhou, Peisong Niu, Liang Sun, Rong Jin, et al. One fits all: Power general time series analysis
904 by pretrained lm. *Advances in neural information processing systems*, 36:43322–43355, 2023.
- 905
906
907
908
909
910
911
912
913
914
915
916
917