ADAPTIVE ENERGY AMPLIFICATION FOR ROBUST TIME SERIES FORECASTING

Anonymous authors
Paper under double-blind review

ABSTRACT

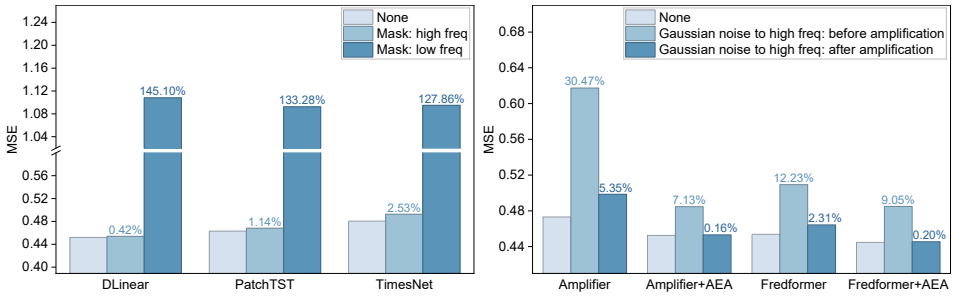
Deep learning models for time series forecasting often exhibit a spectral bias, prioritizing high-energy, low-frequency components while underfitting predictive but low-energy, high-frequency signals. Existing efforts attempt to correct this by amplifying high-frequency components but suffer from indiscriminate amplification, enhancing both meaningful signals and task-irrelevant noise, which destabilizes training and impairs generalization. To address this, we propose **AEA** (**A**daptive **E**nergy **A**mplification), a novel framework that reframes the problem as one of adaptive signal enhancement. AEA introduces two synergistic innovations: (1) a **Spectral Mirroring** mechanism that constructs a phase-preserving, low-frequency surrogate to guide targeted, distortion-free amplification of high-frequency signals; and (2) a lightweight **Differential Embedding** module that operates in a latent space to adaptively suppress common-mode noise. By decoupling signal amplification from noise suppression, AEA selectively enhances only informative features. Extensive experiments on eight benchmark datasets show that our model-agnostic framework consistently improves the forecasting performance of four state-of-the-art backbones, while significantly enhancing training stability and generalization. The code repository is available at `https://anonymous.4open.science/r/AEA-685E/`.

1 INTRODUCTION

Time series forecasting (TSF) is critical in various real-world applications, including traffic flow prediction (Wu et al., 2020), energy management (Zhou et al., 2021), weather forecasting (Liang et al., 2023), financial investment (Oreshkin et al., 2020), human healthcare (Qiu et al., 2024), *etc.* Recent deep learning-based methods, which have powerful nonlinear modeling capabilities to learn complex patterns and feature representations, achieving remarkable performance on TSF, such as convolutional-based (Wu et al., 2023; donghao & wang xue, 2024), Transformer-based (Nie et al., 2023; Liu et al., 2024), and MLP-based methods (Zeng et al., 2023; Wang et al., 2024).

Despite these advances, such models exhibit a fundamental spectral bias: they consistently prioritize high-energy, low-frequency components while overlooking subtle yet predictive high-frequency signals (Xu et al., 2024; Yi et al., 2024). As shown in Figure 1a, masking low-frequency components causes a drastic drop in performance, while masking high-frequency components only marginally impacts performance, revealing the models' over-reliance on low-frequency information with limited capability for modeling high-frequency signals. This learning pathology originates from the model's optimization bias on low-frequency components with high energy. According to *Parseval's Theorem* (Lathi & Green, 1998; Yi et al., 2023), the energy is equivalent between the time and frequency domains. In most real-world time series data, low-frequency components possess substantially higher amplitudes than their high-frequency counterparts, meaning energy is concentrated in the low-frequency part of the spectrum. As a result, the predictive loss landscape becomes dominated by errors from these low-frequency components with high amplitude. This skews the optimization process, compelling the learning algorithm to primarily allocate model capacity toward fitting these dominant, low-frequency signals, while the informative yet low-energy high-frequency details are consequently underfitted (Liu et al., 2023; Piao et al., 2024; Fei et al., 2025).

To address the issue, recent efforts have focused on amplifying the energy of high-frequency components to recalibrate their influence during model optimization. These methods can be broadly



(a) Spectral Bias (b) Indiscriminate Amplification

Figure 1: The average degradation in forecasting performance (Values denote relative increase in MSE (%) compared to the "None" baseline) during the inference stage on ETTh1. (a) When the lower *v.s.* higher 50% of frequency bands are masked (set to zero), the significantly smaller performance drop after high-frequency masking confirms the base model's reliance on low-frequency information and its insensitivity to high-frequency components. (b) Comparing the vanilla amplification method with our enhanced version (+AEA) when artificially injecting Gaussian noise of the same intensity into high frequencies. The results show that the vanilla methods' performance degrades drastically, proving that they are susceptible to noise. In contrast, our method successfully suppresses noise, resulting in significantly enhanced robustness. We present more details in Appendix B.2.

categorized into two main strategies: indirect and direct enhancement. Specifically, the **indirect enhancement** approaches mitigate energy disparity through normalization. For instance, Fredformer (Piao et al., 2024) implements frequency-wise local normalization, which segments the spectrum and normalizes each sub-band individually to eliminate amplitude disparity. On the other hand, the **direct enhancement** strategy, conversely, explicitly modifies the spectral energy distribution. Amplifier (Fei et al., 2025) is one representative work, whose key innovation is spectrum flipping. This technique inverts the spectrum to leverage high-energy signals as a template for boosting low-energy signals.

However, despite their different mechanisms, these approaches share a fundamental flaw: their amplification is indiscriminate. High-frequency bands inherently contain a mixture of predictive signals (*e.g.*, subtle seasonal variations and trends) and task-irrelevant noise (*e.g.*, sensor artifacts and background noise) (Eldele et al., 2024; Kou et al., 2025; Yi et al., 2025). By uniformly elevating the energy across the high-frequency bands, existing methods inevitably amplify noise alongside the valuable signals. This indiscriminate enhancement introduces spectral disturbances that destabilize the optimization process and ultimately impair the model's generalization performance. As empirically demonstrated in Figure 1b, when noise is injected into the high-frequency bands, both indirect and direct enhancement methods suffer a significant performance degradation, underscoring the negative impact of their indiscriminate amplification and revealing an inherent inability to distinguish between informative signals and spurious noise.

To address this limitation, we argue that *the key to unlocking the potential of high-frequency signals lies not in indiscriminate amplification, but in adaptive enhancement*. We introduce AEA (**A**daptive **E**nergy **A**mplification), a novel framework that fundamentally reframes the problem by simultaneously amplifying signals and suppressing noise. AEA achieves this through two synergistic innovations designed to provide a more principled and reasonable energy amplification. First, (1) **Spectral Mirroring** addresses the amplification itself by leveraging the typically cleaner, high-signal-to-noise ratio of the low-frequency spectrum. It constructs a phase-preserving surrogate from these reliable low-frequency components to serve as a structured template, guiding a targeted amplification of high-frequency signals without introducing spectral distortion. Second, to explicitly tackle noise, (2) **Differential Embedding** operates in a learned latent space to identify and filter out common-mode noise, which indiscriminate methods inadvertently amplify. By integrating these two mechanisms, AEA ensures that only the informative, discriminative features within the high-frequency bands are selectively enhanced, thereby resolving the core issue of indiscriminate amplification by separating the targeted enhancement of predictive signals from the active suppression of noise.

In summary, our contributions can be highlighted as followings:

- We systematically identify the problem of "indiscriminate amplification" in forecasting models against spectral bias, establishing a novel connection between targeted energy amplification and adaptive noise suppression.
- We propose AEA, a model-agnostic framework that employs spectral mirroring for distortion-free amplification and differential embedding for adaptive noise suppression, seamlessly integrating with various forecasting backbones.
- We empirically demonstrate that AEA consistently improves accuracy, stability, and generalization across eight benchmark datasets and four state-of-the-art backbones, offering a robust new paradigm for frequency-aware time series forecasting.

2 Related Work

2.1 Time Series Forecasting Models

Traditional time series forecasting methods such as ARIMA (Zhang, 2003) and Prophet (Taylor & Letham, 2018; Triebe et al., 2021) are effective at capturing trend and seasonal components in time series (Cleveland et al., 1990; Ahmed et al., 2010; Wen et al., 2020; Zeng et al., 2023; Stitsyuk & Choi, 2025). With the continuous growth in data availability, deep learning methods have brought revolutionary advances to the field, introducing more complex and efficient models (Torres et al., 2021; Lim & Zohren, 2021). Convolutional Neural Networks (CNNs) (Bai et al., 2018; Wan et al., 2019; Sen et al., 2019; Liu et al., 2022a; Wu et al., 2023) have been widely adopted to capture local temporal dependencies, while Recurrent Neural Networks (RNNs) (Rangapuram et al., 2018; Smyl, 2020; Salinas et al., 2020; Hewamalage et al., 2021), although proficient at processing sequential information, often struggle with long-sequence modeling. Transformer-based models (Zhou et al., 2021; Wu et al., 2021; Liu et al., 2022b; Zhang & Yan, 2022; Nie et al., 2023; Liu et al., 2024; Wen et al., 2023; Tang & Matteson, 2021; Zhou et al., 2022b; Liu et al., 2021; Feng et al., 2024), typically equipped with self-attention mechanisms (Vaswani et al., 2017), excel at capturing long-range dependencies, albeit at considerable computational cost. Recently, linear models (Oreshkin et al., 2020; Zhang et al., 2022; Das et al., 2023) such as DLinear (Zeng et al., 2023) and TSMixer (Chen et al., 2023) have gained popularity due to their simplicity and strong performance in long-term forecasting, though they may underperform on highly non-linear and complex patterns. Furthermore, multi-periodicity analysis (Benaouda et al., 2006; Percival & Walden, 2000; Wu et al., 2023; Wang et al., 2022; Chen et al., 2024; Yi et al., 2023; Zhou et al., 2022a) continues to play an essential role in the preprocessing stages of advanced modeling pipelines.

2.2 Frequency Domain Methods in Time Series Forecasting

Recent studies have increasingly leveraged frequency-domain techniques to enhance the accuracy and efficiency of time series forecasting (Yi et al., 2025). Prominent examples include FEDformer (Zhou et al., 2022b), which accelerates attention via frequency-domain low-rank approximation; FreTS (Yi et al., 2023), which integrates global frequency properties into an efficient MLP architecture; and FITS (Xu et al., 2024), which employs frequency interpolation as an effective low-pass filter. A common characteristic of these approaches is their tendency to prioritize high-energy, low-frequency components, a design choice that aligns with the natural energy distribution of many real-world time series. However, this emphasis may lead to insufficient use of subtle yet predictive high-frequency signals, which often carry critical short-term variations and anomaly patterns. The challenge of effectively balancing frequency components without amplifying noise has thus emerged as a key issue in frequency-aware forecasting. Recent efforts have attempted to address this spectral imbalance. Fredformer (Piao et al., 2024) mitigates frequency bias in Transformers by promoting more balanced feature learning across bands, yet its architecture-specific design limits generalizability. Amplifier (Fei et al., 2025) directly elevates high-frequency energy to match low-frequency levels, aiming to equalize gradient scales across the spectrum. However, such uniform amplification risks enhancing high-frequency noise alongside signals. In contrast to these end-to-end architectures, our proposed AEA is designed as a model-agnostic plugin that decouples signal amplification from noise suppression. By combining targeted energy amplification and adaptive noise suppression, AEA achieves more nuanced enhancement while maintaining robustness and efficiency across diverse forecasting backbones.

3 PRELIMINARIES

Time Series Forecasting. Formally, let $X = [\boldsymbol{x}_1, \dots \boldsymbol{x}_T] \in \mathbb{R}^{T \times C}$ be a time series, where T is the length of historical data. $\boldsymbol{x}_t \in \mathbb{R}^C$ represents the observation at time t. C denotes the number of variates (*i.e.,* channels). The objective is to construct a predictive model f that estimates the future values of the series, $Y = [\hat{\boldsymbol{x}}_{T+1}, \dots, \hat{\boldsymbol{x}}_{T+H}] \in \mathbb{R}^{H \times C}$, where H is the forecasting horizon.

Real Fast Fourier Transform. Given a real-valued sequence $x[n]$ of length N, we employ the Real Fast Fourier Transform (rFFT) (Sorensen et al., 1987) to efficiently convert it into the frequency domain, and transform it back using the inverse rFFT (irFFT). The rFFT/irFFT exploits the conjugate symmetry of real-valued inputs, reducing the computational complexity from $O(N^2)$ to $O(N \log N)$ while compressing the output to $N/2 + 1$ complex-valued frequency components. The resulting spectrum $\mathcal{X} \in \mathbb{C}^{N/2+1}$ contains both magnitude and phase information:

$$A[k] = |\mathcal{X}[k]|, \quad \theta[k] = \angle \mathcal{X}[k] \tag{1}$$

where $A[k]$ represents amplitude and $\theta[k]$ phase at frequency $\omega_k = 2\pi k/N$. We provide more details of the Fourier Transform in Appendix A.1.

4 PROPOSED METHOD

4.1 OVERALL ARCHITECTURE

We propose the Adaptive Energy Amplification (AEA) framework to address the limitation of indiscriminate amplification in existing frequency-domain forecasting methods. As illustrated in Figure 2, AEA operates primarily in the frequency domain and consists of two core innovations: (1) a Spectral Mirroring module that performs targeted amplification of high-frequency signals via a phase-preserving surrogate spectrum, and (2) a Differential Embedding module that suppresses common-mode noise in a latent space to enhance discriminative features. To ensure spectral consistency, we incorporate an Energy Predictor that aligns the predictions with the original data distribution. The entire framework is model-agnostic and seamlessly integrates with various forecasting backbones. We present the pseudo-code in Algorithm 1.

4.2 SPECTRAL MIRRORING

The Spectral Mirroring component aims to amplify the energy of high-frequency components in a targeted and distortion-free manner. To enhance attention to low-energy, high-frequency components as well as high-energy, low-frequency components, we reverse the entire spectrum to create a phase-preserving surrogate (Fei et al., 2025). For an input spectrum $\mathcal{X}[k]$ with $k = 0, 1, \dots, F-1$ (where $F = \lfloor T/2 \rfloor + 1$), the reversed spectrum is obtained by:

$$\mathcal{X}_{\text{reverse}}[k] = \mathcal{X}[F - 1 - k]. \tag{2}$$

This inverts the energy distribution, allowing high-energy low-frequency components to serve as a template for amplifying low-energy high-frequency ones. A learnable scaling matrix $M \in \mathbb{R}^{F \times C}$ is applied to control the degree of amplification per frequency and channel adaptively:

$$\mathcal{X}_{\text{scaled}}[k, c] = \mathcal{X}_{\text{reverse}}[k, c] \cdot M[k, c], \quad \text{for } k = 0, 1, \dots, F-1; c = 0, 1, \dots, C-1. \tag{3}$$

The key to avoiding distortion lies in how we mix the original and mirrored spectra. A simple linear combination of amplitudes and phases would likely result in destructive interference (Demirel & Holz, 2025). Instead, we employ a *phase mixing* strategy that minimizes disruptive phase discontinuities. The *phase mixing* involves calculating the circular difference between the original and scaled phases, adjusting it to the shortest angular path within $[-\pi, \pi]$, and then blending the phases accordingly:

$$\Delta\theta[k, c] = (\theta_1[k, c] - \theta_2[k, c]) \mod 2\pi, \tag{4}$$

$$\Delta\theta_{\text{adjusted}}[k, c] = \begin{cases} \Delta\theta[k, c] - 2\pi, & \text{if } \Delta\theta[k, c] > \pi, \\ \Delta\theta[k, c], & \text{otherwise}, \end{cases} \tag{5}$$

$$\theta_{\text{mix}}[k, c] = \theta_1[k, c] + \Delta\theta_{\text{adjusted}}[k, c], \tag{6}$$

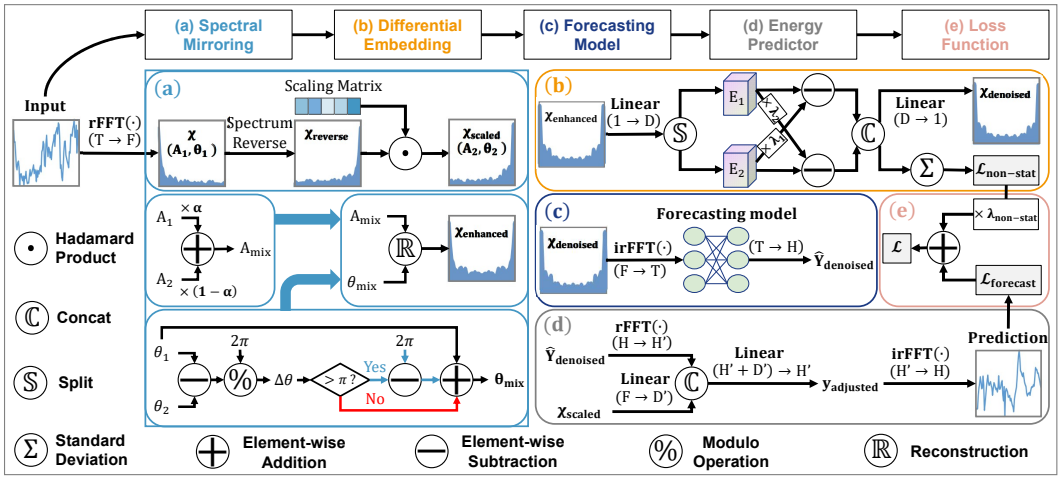


Figure 2: An illustration of the proposed AEA framework. The input time series is first transformed into the frequency domain. The framework consists of five components: (a) The Spectral Mirroring module (Section 4.2) that reverses the spectrum to adaptively amplify high-frequency signals without distortion through a learned scaling matrix and phase-preserving mixing. (b) The Differential Embedding module (Section 4.3) that projects the enhanced spectrum into a latent space to suppress common-mode noise via a differential operation, and yields the non-stationarity loss to stabilize learning. (c) The denoised spectrum is converted back to the time domain via irFFT for the base model to forecast. (d) The Energy Predictor module (Section 4.4) that aligns the output of the base model with the original data's spectral properties. (e) The optimization of the entire framework by a combined loss function (Section 4.5) comprising the forecast error and the non-stationarity loss.

where $\theta_1[k,c] = \angle(\mathcal{X}[k,c])$ and $\theta_2[k,c] = \angle(\mathcal{X}_{\text{scaled}}[k,c])$.

The mixed amplitude $A_{\text{mix}}[k,c]$ is computed as:

$$A_{\text{mix}}[k,c] = \alpha \cdot A_1[k,c] + (1-\alpha) \cdot A_2[k,c], \tag{7}$$

where $A_1[k,c] = |\mathcal{X}[k,c]|$, $A_2[k,c] = |\mathcal{X}_{\text{scaled}}[k,c]|$, α is a mixing ratio, and we set it to 0.5.

The enhanced spectrum is then reconstructed as:

$$\mathcal{X}_{\text{enhanced}}[k,c] = A_{\text{mix}}[k,c] \cdot e^{j\theta_{\text{mix}}[k,c]}. \tag{8}$$

This process preserves the temporal structure of the original signal while amplifying informative high-frequency components. The output of this operation provides a refined input for the subsequent differential embedding module.

4.3 Differential Embedding with Non-Stationarity Loss

The Differential Embedding module suppresses common-mode noise while preserving discriminative signals in the enhanced spectrum $\mathcal{X}_{\text{enhanced}}$. It projects the input into an embedding space, applies a differential operation for noise suppression, and incorporates a regularization loss to stabilize training. The input spectrum is first projected into a complex-valued embedding space:

$$E_1 || E_2 = W_e \cdot \mathcal{X}_{\text{enhanced}} + b_e, \tag{9}$$

where $W_e \in \mathbb{C}^{D\times 1}$ and $b_e \in \mathbb{C}^D$ are learnable parameters, and D is the embedding dimension. $E_1 \in \mathbb{C}^{F\times C\times \frac{D}{2}}$ and $E_2 \in \mathbb{C}^{F\times C\times \frac{D}{2}}$ are the subspaces of the differential embedding.

Inspired by the principle of *differential attention* (Ye et al., 2025; Wang et al., 2025), we apply a differential operation in the embedding space as follows:

$$E_1' = E_1 - \lambda_1 \cdot E_2, \quad E_2' = E_2 - \lambda_2 \cdot E_1, \tag{10}$$

where λ_1 and λ_2 are learnable scalars. These scalars are initialized with a random constant λ_{init} and a Softplus function (Nair & Hinton, 2010) to ensure they remain positive throughout training:

$$\lambda_1 = \text{Softplus}(\lambda_{\text{init}}), \quad \lambda_2 = \text{Softplus}(\lambda_{\text{init}}), \tag{11}$$

The results are concatenated to form the denoised embedding:

$$E' = \text{Concat}(E'_1, E'_2), \quad E' \in \mathbb{C}^{F \times C \times D}. \tag{12}$$

This operation is theoretically grounded in noise suppression (see Proposition 4.1). The denoised embedding is projected back to the frequency domain:

$$\mathcal{X}_{\text{denoised}} = W_p \cdot E' + b_p, \quad \mathcal{X}_{\text{denoised}} \in \mathbb{C}^{F \times C}, \tag{13}$$

where $W_p \in \mathbb{C}^{1 \times D}$ and $b_p \in \mathbb{C}^1$ are learnable parameters.

Theoretical analysis 4.7 shows that this differential operation reduces gradient bias from common-mode noise while preserving beneficial stochastic variance. To further enhance stability, we introduce a non-stationarity loss that penalizes excessive variability in embedding magnitudes across batches:

$$\mathcal{L}_{\text{non-stat}} = \sqrt{\text{Var}_{\mathbf{x} \sim \mathcal{B}}\left(|E'|\right)}, \tag{14}$$

where $\mathcal{B}$ represents the current batch of samples. This loss term serves as a regularizer that encourages the model to learn stable, stationary representations that are robust to batch-wise variations. The final denoised spectrum $\mathcal{X}_{\text{denoised}}$ is transformed back to the time domain via inverse rFFT for forecasting.

4.4 Energy Predictor

The Spectral Mirroring and Differential Embedding alter the energy distribution of the input signal. While this enhancement improves the model's ability to capture high-frequency components, directly using the base model's predictions on this enhanced and denoised signal would therefore yield outputs with inconsistent energy characteristics (Liu et al., 2023). To ensure spectral consistency with the original data, our Energy Predictor learns a frequency-domain mapping. It takes the scaled historical spectrum $\mathcal{X}_{\text{scaled}}$ and the base model's preliminary prediction $\hat{Y}_{\text{denoised}}$ as inputs. First, $\hat{Y}_{\text{denoised}}$ is transformed to the frequency domain via rFFT to obtain $\mathcal{Y}_{\text{denoised}}$. Then, $\mathcal{X}_{\text{scaled}}$ is projected into a latent embedding to represent the historical context. This embedding is concatenated with $\mathcal{Y}_{\text{denoised}}$, and the combined representation is passed through a final linear projection. This step yields the adjusted spectrum $\mathcal{Y}_{\text{adjusted}}$, effectively aligning the prediction's spectral properties with the original data. These operations are defined precisely as follows:

$$\mathcal{Y}_{\text{denoised}} = \text{rFFT}(\hat{Y}_{\text{denoised}}), \qquad \mathcal{Y}_{\text{denoised}} \in \mathbb{C}^{H' \times C}, \tag{15}$$

$$\mathcal{E} = W_1 \cdot \mathcal{X}_{\text{scaled}} + b_1, \qquad \mathcal{E} \in \mathbb{C}^{D' \times C}, \tag{16}$$

$$\mathcal{Y}_{\text{adjusted}} = W_2 \cdot \text{Concat}(\mathcal{E}, \mathcal{Y}_{\text{denoised}}) + b_2, \qquad \mathcal{Y}_{\text{adjusted}} \in \mathbb{C}^{(D'+H') \times C}, \tag{17}$$

where $H' = \lfloor H/2 \rfloor + 1$, and $W_1 \in \mathbb{C}^{D' \times F}$, $W_2 \in \mathbb{C}^{H' \times (D'+H')}$, $b_1 \in \mathbb{C}^{D'}$, and $b_2 \in \mathbb{C}^{H'}$ are learnable parameters.

Finally, the adjusted spectrum is transformed back to the time domain to produce the final prediction:

$$\hat{Y} = \text{irFFT}(\mathcal{Y}_{\text{adjusted}}). \tag{18}$$

These designs enable the predictor to learn sophisticated adjustments based on the full spectral information rather than just summary statistics. By operating in the frequency domain and utilizing the mirrored spectrum as a conditioning signal, our Energy Predictor effectively bridges the distributional gap between the enhanced input space and the original data characteristics, resulting in predictions that maintain both the improved representational quality and appropriate energy distribution.

4.5 Loss Function

We follow a multi-task optimization framework (Vandenhende et al., 2022) that simultaneously ensures accurate forecasting while maintaining representation stability and formulates the loss function $\mathcal{L}$ as:

$$\mathcal{L} = \mathcal{L}_{\text{forecast}} + \lambda_{\text{non-stat}} \cdot \mathcal{L}_{\text{non-stat}}, \tag{19}$$

where $\mathcal{L}_{\text{forecast}}$ is the Mean Squared Error (MSE) between predictions $\hat{Y}$ and ground truth Y, $\mathcal{L}_{\text{non-stat}}$ is the regularization term introduced in Equation 14, and $\lambda_{\text{non-stat}}$ is a balancing hyperparameter.

4.6 Computational Complexity Analysis

AEA is designed as a plug-and-play framework whose computational overhead is typically negligible compared to forecasting backbones, especially those with quadratic complexity. The complexity is dominated by FFT operations and linear projections across its modules. The rFFT/irFFT transformations require $O(T \log T)$ per channel. Spectral Mirroring performs element-wise operations in $O(T)$ time. Both Differential Embedding and Energy Predictor involve linear projections with complexity $O(F \cdot C \cdot D)$ or $O(F \cdot C \cdot D')$, where $F = \lfloor T/2 \rfloor + 1 \approx T$, D and D' are fixed (typically 64-128). Thus, the overall complexity of AEA is linear in both sequence length and number of channels, i.e., $O(T \cdot C)$. This is significantly more efficient than the quadratic complexity $O(T^2 \cdot C)$ of transformer backbones, making AEA a practical enhancement for real-world forecasting applications.

4.7 Theoretical Analysis

Notation. Let $e^{(1)}, e^{(2)} \in \mathbb{C}^{F \times C \times \frac{D}{2}}$ denote the two embedding subspaces from Equation 9, Θ the model parameters of loss L, $\lambda \in \mathbb{R}^+$ a learnable scaling parameter, s_i the true signal component, $n_i^{(c)}$ the common-mode noise, and ϵ_i the independent stochastic noise.

Proposition 4.1 (Adaptive Noise Suppression via Differential Embedding)**.** *The differential embedding mechanism $e^{(diff)} = e^{(1)} - \lambda e^{(2)}$ provides adaptive suppression of common-mode noise while preserving discriminative signals. The resulting gradient estimates $\hat{g} = g + \delta$ exhibit superior bias-variance trade-off:*

$$\mathbb{E}[\delta] = (1 - \lambda^*)\mathbf{b}_g, \quad Var(\delta) = (1 - \lambda^*)^2 \sigma_c^2 + (1 + \lambda^{*2})\sigma_\epsilon^2 \tag{20}$$

where λ^* is the optimal value minimizing the training objective, $\mathbf{b}_g$ is the bias introduced by common-mode noise, δ is the gradient noise, and σ_c^2, σ_ϵ^2 represent the gradient variances from common-mode and stochastic noise components, respectively.

Proof. We begin by decomposing the embedding into signal and noise components under Assumption A.1, which is supported by previous studies (Ye et al., 2025; Wang et al., 2025). This common-mode noise often stems from systematic biases present in the input data (*e.g.*, stop words in NLP, background regions in spatio-temporal data, or certain frequency components in the spectrum (Eldele et al., 2024)). The differential operation yields:

$$e^{(\text{diff})} = s^{(\text{diff})} + (1 - \lambda)n^{(c)} + (\epsilon^{(1)} - \lambda \epsilon^{(2)}) \tag{21}$$

Taking expectation over stochastic noise ($\mathbb{E}[\epsilon^{(1)}] = \mathbb{E}[\epsilon^{(2)}] = 0$):

$$\mathbb{E}[\delta] = (1 - \lambda)\mathbb{E}\left[\frac{\partial L}{\partial e^{(\text{diff})}} \cdot \frac{\partial n^{(c)}}{\partial \Theta}\right] = (1 - \lambda^*)\mathbf{b}_g \tag{22}$$

The variance analysis follows from the uncorrelatedness of noise components. The complete derivation, including detailed expectations, variance decompositions, and convergence guarantees, is provided in Appendix A.4. This proposition validates our differential embedding by ensuring that: (i) common-mode noise amplified during spectral mirroring is effectively suppressed, (ii) the non-stationarity loss $\mathcal{L}_{\text{non-stat}}$ in Equation 14 stabilizes training by controlling gradient variance, and (iii) the adaptive parameter λ^* optimally balances noise suppression against signal preservation throughout optimization. □

5 Experiments

5.1 Experimental Setup

Datasets. We conduct extensive experiments on eight widely used real-world multivariate time series forecasting datasets, including ETT (ETTh1, ETTh2, ETTm1, and ETTm2), Electricity, Traffic, and Weather, which are utilized by Autoformer (Wu et al., 2021). For a fair comparison, we follow the same standard protocol (Wu et al., 2023) and split all forecasting datasets into training, validation, and test sets by the ratio of 6:2:2 for the ETT dataset and 7:1:2 for the other datasets. More can be found in the Appendix B.3.

Table 1: Overall performance comparison *w.r.t.* forecasting models with their counterparts enhanced by the AEA in terms of MSE and MAE, the lower the better. The forecasting horizons are $\{96, 192, 336, 720\}$. The better performance in each setting is shown in **bold**. The best results for each row are underlined. 'Avg' denotes the average results of four forecasting horizons; The last column, 'IMP (%)', shows the average percentage of MSE/MAE improvement over four base models.

Model		DLinear		+ AEA		PatchTST		+ AEA		TimesNet		+ AEA		Amplifier		+ AEA		IMP (%)
Metric		MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	
ETTh1	96	0.384	0.397	**0.380**	**0.392**	0.387	0.403	**0.374**	**0.396**	0.390	0.414	**0.395**	**0.410**	0.437	0.439	**0.397**	**0.407**	2.956
	192	0.433	0.427	**0.432**	**0.422**	0.455	0.444	**0.428**	**0.431**	0.519	0.486	**0.441**	**0.436**	0.451	0.436	**0.440**	**0.428**	5.457
	336	0.481	0.459	**0.479**	**0.458**	0.490	0.463	**0.466**	**0.452**	0.471	0.458	**0.473**	**0.452**	0.497	0.457	**0.482**	**0.447**	1.775
	720	0.509	0.506	**0.495**	**0.489**	0.510	0.496	**0.479**	**0.478**	0.543	0.511	**0.490**	**0.476**	0.508	0.484	**0.492**	**0.475**	4.890
	Avg	0.452	0.447	**0.447**	**0.440**	0.460	0.451	**0.437**	**0.439**	0.480	0.467	**0.450**	**0.443**	0.473	0.454	**0.452**	**0.439**	3.815
ETTm1	96	0.344	0.371	**0.332**	**0.368**	0.327	0.366	**0.327**	**0.365**	0.357	0.389	**0.338**	**0.377**	0.323	0.363	**0.321**	**0.361**	1.805
	192	0.381	0.393	**0.370**	**0.386**	**0.367**	0.388	0.369	**0.387**	0.440	0.427	**0.375**	**0.391**	**0.365**	**0.382**	**0.365**	**0.382**	3.913
	336	0.416	0.418	**0.402**	**0.408**	0.405	0.417	**0.406**	**0.410**	0.410	0.421	**0.410**	**0.414**	0.397	0.403	**0.393**	**0.401**	2.592
	720	0.478	0.458	**0.464**	**0.444**	0.481	0.464	**0.458**	**0.445**	0.518	0.468	**0.478**	**0.451**	0.476	0.443	**0.468**	**0.437**	3.705
	Avg	0.405	0.410	**0.392**	**0.402**	0.395	0.409	**0.390**	**0.402**	0.431	0.427	**0.400**	**0.408**	0.390	0.398	**0.387**	**0.395**	2.728
ETTh2	96	0.336	0.386	**0.329**	**0.384**	0.302	0.352	**0.290**	**0.339**	**0.322**	0.360	**0.322**	**0.359**	0.291	0.344	**0.286**	**0.337**	1.777
	192	0.452	0.459	**0.450**	**0.457**	0.416	0.424	**0.375**	**0.392**	0.428	0.427	**0.419**	**0.417**	0.370	0.396	**0.369**	**0.389**	3.047
	336	0.579	0.536	**0.576**	**0.533**	0.500	0.480	**0.416**	**0.425**	0.466	0.454	**0.427**	**0.435**	0.427	0.437	**0.415**	**0.430**	5.898
	720	**0.784**	**0.638**	0.795	0.641	0.482	0.478	**0.421**	**0.439**	0.421	0.440	**0.418**	**0.438**	0.439	0.452	**0.424**	**0.443**	3.177
	Avg	**0.538**	0.505	**0.538**	**0.504**	0.425	0.434	**0.375**	**0.399**	0.409	0.420	**0.397**	**0.412**	0.382	0.407	**0.374**	**0.400**	3.610
ETTm2	96	**0.188**	**0.283**	0.189	0.286	**0.181**	0.266	0.182	**0.265**	0.187	0.264	**0.175**	**0.257**	0.178	0.260	**0.177**	**0.260**	0.971
	192	0.282	0.360	**0.271**	**0.347**	0.249	0.311	**0.242**	**0.300**	0.253	0.306	**0.247**	**0.304**	0.241	0.301	**0.239**	**0.299**	2.369
	336	**0.360**	**0.411**	0.372	0.420	0.311	0.352	**0.305**	**0.342**	0.323	0.349	**0.316**	**0.345**	0.299	0.340	**0.297**	**0.339**	0.416
	720	0.546	0.518	**0.541**	**0.515**	0.406	0.404	**0.399**	**0.399**	0.423	0.408	**0.430**	**0.409**	0.394	**0.396**	**0.392**	**0.396**	0.447
	Avg	0.344	0.393	**0.343**	**0.392**	0.287	0.333	**0.282**	**0.326**	0.296	0.332	**0.292**	**0.329**	0.278	0.324	**0.276**	**0.323**	0.950
Electricity	96	0.196	0.280	**0.192**	**0.278**	0.174	0.260	**0.172**	**0.259**	0.175	0.278	**0.148**	**0.248**	0.178	0.267	**0.177**	**0.260**	4.418
	192	0.195	0.283	**0.190**	**0.279**	0.194	0.275	**0.193**	**0.274**	0.189	0.290	**0.158**	**0.254**	0.247	0.306	**0.243**	**0.303**	5.108
	336	0.208	0.299	**0.202**	**0.294**	0.210	0.295	**0.207**	**0.288**	0.207	0.303	**0.185**	**0.283**	0.308	0.343	**0.302**	**0.340**	3.717
	720	0.243	0.331	**0.237**	**0.326**	0.237	0.318	**0.233**	**0.313**	0.254	0.338	**0.216**	**0.311**	0.398	0.396	**0.397**	**0.396**	4.236
	Avg	0.211	0.298	**0.205**	**0.294**	0.204	0.287	**0.201**	**0.283**	0.206	0.302	**0.177**	**0.274**	0.283	0.328	**0.279**	**0.326**	4.337
Exchange	96	0.080	**0.199**	**0.077**	**0.199**	0.091	0.211	**0.084**	**0.204**	0.109	0.240	**0.099**	**0.223**	**0.084**	0.203	**0.084**	**0.201**	4.188
	192	0.161	0.296	**0.158**	**0.295**	0.191	0.312	**0.186**	**0.306**	**0.193**	**0.323**	0.196	0.319	0.179	0.300	**0.178**	**0.299**	0.924
	336	0.302	0.414	**0.270**	**0.394**	0.325	0.414	**0.323**	**0.410**	0.394	0.465	**0.346**	**0.428**	0.337	0.419	**0.327**	**0.412**	5.544
	720	0.778	**0.666**	**0.773**	0.667	1.004	0.759	**0.877**	**0.707**	**1.013**	**0.770**	1.063	0.776	0.874	0.703	**0.851**	**0.694**	2.297
	Avg	0.330	0.394	**0.320**	**0.389**	0.403	0.424	**0.367**	**0.407**	0.427	0.449	**0.426**	**0.436**	0.369	0.406	**0.360**	**0.402**	3.008
Weather	96	0.198	0.259	**0.169**	**0.244**	0.185	0.228	**0.173**	**0.219**	0.168	0.218	**0.162**	**0.209**	0.173	0.219	**0.171**	**0.216**	5.124
	192	0.235	0.294	**0.213**	**0.286**	0.226	0.260	**0.218**	**0.256**	0.232	0.269	**0.216**	**0.256**	0.221	0.258	**0.217**	**0.255**	3.995
	336	0.288	0.343	**0.266**	**0.327**	0.278	0.297	**0.273**	**0.296**	0.290	0.309	**0.276**	**0.297**	0.274	0.295	**0.272**	**0.293**	3.065
	720	0.350	0.389	**0.336**	**0.378**	0.353	0.346	**0.351**	**0.345**	0.355	0.351	**0.355**	**0.351**	0.351	0.345	**0.348**	**0.342**	1.197
	Avg	0.268	0.321	**0.246**	**0.309**	0.260	0.283	**0.254**	**0.279**	0.261	0.287	**0.252**	**0.278**	0.255	0.279	**0.252**	**0.277**	3.042
Traffic	96	0.652	0.400	**0.611**	**0.388**	0.488	0.308	**0.481**	**0.306**	0.604	0.322	**0.456**	**0.290**	0.554	0.360	**0.536**	**0.350**	7.610
	192	0.601	0.375	**0.589**	**0.370**	0.499	0.307	**0.486**	**0.301**	0.625	0.329	**0.470**	**0.300**	0.542	0.352	**0.530**	**0.340**	7.004
	336	0.608	0.378	**0.597**	**0.374**	0.513	0.314	**0.503**	**0.309**	0.651	0.341	**0.506**	**0.320**	0.555	0.358	**0.536**	**0.342**	6.168
	720	0.648	0.399	**0.634**	**0.392**	0.547	0.336	**0.541**	**0.328**	0.695	0.365	**0.569**	**0.347**	0.592	0.370	**0.569**	**0.359**	5.188
	Avg	0.627	0.388	**0.614**	**0.383**	0.512	0.316	**0.503**	**0.311**	0.644	0.339	**0.500**	**0.314**	0.561	0.360	**0.543**	**0.348**	6.436

Base model and Experimental details. AEA is a model-agnostic framework that can be seamlessly integrated with arbitrary time series forecasting models to enhance their performance. We select four state-of-the-art models as base models: DLinear (Zeng et al., 2023), Amplifier (Fei et al., 2025), PatchTST (Nie et al., 2023), and TimesNet Wu et al. (2023), which collectively represent three dominant forecasting paradigms: linear models, Transformers, and convolutional architectures. Following the established evaluation protocol in TimesNet, we adopt Mean Squared Error (MSE) and Mean Absolute Error (MAE) as the primary evaluation metrics, as well as set the historical input length to 96, and forecasting horizons are evaluated at $\{96, 192, 336, 720\}$. To ensure a fair comparison, we consistently use the same experimental configuration as the original implementations. All experiments are conducted using PyTorch (Paszke et al., 2019) on a single NVIDIA RTX A100 80GB GPU. Experiment configurations and implementations are detailed in Appendix B.4.

5.2 Main Results

We report the MSE and MAE on eight real-world datasets for long-term forecasting evaluation in Table 1. The forecasting horizon is $\{96, 192, 336, 720\}$. From the table, we observe that the model enhanced with AEA outperforms the base model in general. Specifically, AEA improves forecasting performance in 96.563% cases in MSE and 97.19% cases in MAE. Remarkably, AEA achieves a substantial boost on TimesNet, with a significant reduction on MSE by 8.108% and MAE by 4.337%. The last column of the table quantifies the average percentage improvement in terms of MSE/MAE, at 3.551%, which underscores the consistent enhancement brought by AEA across all forecasting horizons and datasets.

Table 2: Ablation study results across five datasets. Models are compared in terms of MSE and MAE (lower values are better) using the DLinear backbone under a forecasting horizon of 96. The best result for each dataset is highlighted in **bold**. 'Avg' denotes the average results of MSE and MAE. The last column, 'Drop (%)', shows the average performance deterioration percentage of all datasets.

	ETTh1	ETTh2	Weather	Exchange	Traffic	Avg	Drop (%)
AEA	**0.386**	**0.356**	**0.207**	**0.138**	**0.500**	**0.317**	**-**
w/o Spectral Mirroring	0.393	0.388	0.229	0.147	0.529	0.338	8.139
w/o Phase Mixing	0.389	0.404	0.210	0.143	0.522	0.333	5.737
w/o Differential Embedding	0.392	0.391	0.243	0.151	0.532	0.342	10.520
w/o Non-stationarity Loss	0.391	0.380	0.231	0.150	0.524	0.335	7.799
w/o Energy Predictor	0.402	0.374	0.233	0.164	0.539	0.342	11.508

Table 3: Parameter sensitivity study. Forecasting performance *w.r.t.* different Differential Embedding dimensions D with DLinear as backbone on four datasets under a forecasting horizon of 96.

Dimension	D=64				D=128				D=256				D=512			
Metric	MSE	MAE	Params	Time	MSE	MAE	Params	Time	MSE	MAE	Params	Time	MSE	MAE	Params	Time
ETTh1	**0.380**	**0.392**	195	7.617	0.383	0.394	387	10.972	0.382	0.393	771	14.713	**0.380**	**0.392**	1539	27.907
Exchange	**0.077**	**0.199**	195	7.617	0.083	0.210	387	10.972	0.081	0.206	771	14.713	0.081	0.206	1539	27.907
Weather	**0.169**	**0.244**	195	7.617	0.178	0.256	387	10.972	0.172	0.247	771	14.713	0.177	0.257	1539	27.907
Traffic	**0.611**	0.388	195	7.617	**0.611**	**0.386**	387	10.972	0.620	0.389	771	14.713	0.633	0.392	1539	27.907

5.3 Model Analysis

Ablation Study. We conduct an ablation study on the DLinear backbone under a forecasting horizon of 96 to validate the contribution of each component in AEA, wherein individual modules are systematically excluded ('w/o'). The results, summarized in Table 2, demonstrate that the complete AEA framework—integrating *Spectral Mirroring*, *Phase Mixing*, *Differential Embedding*, *Non-stationarity Loss*, and *Energy Predictor*—achieves the best performance. The degradation observed in all ablated settings confirms the necessity of the proposed modules. Notably, the absence of the *Energy Predictor* leads to the most significant performance drop (11.508% deterioration), underscoring its critical role in aligning the distribution of the denoised signal with the original data. Removing the *Differential Embedding* module also causes a notable decline (10.520% deterioration), highlighting its importance in common-mode noise suppression for learning robust representations. The *Spectral Mirroring* module proves essential, as its removal results in an average result of 0.338 (8.139% deterioration), validating its effectiveness in high-frequency amplification. In contrast, ablating *Phase Mixing* or the *Non-stationarity Loss* consistently degrades performance, further affirming their contributions to stable and distortion-free feature enhancement.

Sensitivity of Differential Embedding dimension D. As mentioned in 4.6, the complexity of the Differential Embedding module is $O(F \cdot C \cdot D)$, dominated by embedding dimension D. We evaluate the influence of different D values on prediction accuracy, parameters, and running time (ms/iter) in Table 3 across four datasets on the DLinear backbone under a forecasting horizon of 96. Results show that stable forecasting accuracy is achieved across dimensions, with the smallest setting ($D = 64$, only 0.20K parameters, 7.6 ms) already attaining competitive results, even outperforming larger dimensions on Weather and Exchange. These observations confirm that the differential embedding module is both lightweight and effective, requiring only a modest number of parameters to deliver strong performance. Due to the page limit, we provide more sensitivity analysis in Appendix C.1.

6 Conclusion

We have systematically identified the problem of indiscriminate amplification in existing frequency-aware forecasting methods, which amplifies both informative high-frequency signals and task-irrelevant noise, leading to unstable training and compromised generalization. To tackle this issue, we introduce AEA, a novel model-agnostic framework that reformulates frequency enhancement as a dual process of targeted signal amplification and adaptive noise suppression. Extensive experiments on eight real-world datasets demonstrate that AEA consistently improves the forecasting accuracy, robustness, and training stability across four state-of-the-art forecasting backbones.

REFERENCES

Nesreen K Ahmed, Amir F Atiya, Neamat El Gayar, and Hisham El-Shishiny. An empirical comparison of machine learning models for time series forecasting. *Econometric reviews*, 29(5-6): 594–621, 2010.

Shaojie Bai, J Zico Kolter, and Vladlen Koltun. An empirical evaluation of generic convolutional and recurrent networks for sequence modeling. *arXiv preprint arXiv:1803.01271*, 2018.

Djamel Benaouda, Fionn Murtagh, J-L Starck, and Olivier Renaud. Wavelet-based nonlinear multiscale decomposition model for electricity load forecasting. *Neurocomputing*, 70(1-3):139–154, 2006.

Peng Chen, Yingying ZHANG, Yunyao Cheng, Yang Shu, Yihang Wang, Qingsong Wen, Bin Yang, and Chenjuan Guo. Pathformer: Multi-scale transformers with adaptive pathways for time series forecasting. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=lJkOCMP2aW.

Si-An Chen, Chun-Liang Li, Sercan O Arik, Nathanael Christian Yoder, and Tomas Pfister. TSMixer: An all-MLP architecture for time series forecast-ing. *Transactions on Machine Learning Research*, 2023. ISSN 2835-8856. URL https://openreview.net/forum?id=wbpxTuXgm0.

Robert B Cleveland, William S Cleveland, Jean E McRae, and Irma Terpenning. Stl: A seasonal-trend decomposition. *J. Off. Stat*, 6(1):3–73, 1990.

Abhimanyu Das, Weihao Kong, Andrew Leach, Shaan K Mathur, Rajat Sen, and Rose Yu. Long-term forecasting with tiDE: Time-series dense encoder. *Transactions on Machine Learning Research*, 2023. ISSN 2835-8856. URL https://openreview.net/forum?id=pCbC3aQB5W.

Berken Utku Demirel and Christian Holz. Shifting the paradigm: A diffeomorphism between time series data manifolds for achieving shift-invariancy in deep learning. In Y. Yue, A. Garg, N. Peng, F. Sha, and R. Yu (eds.), *International Conference on Representation Learning*, volume 2025, pp. 99210–99249, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/file/f631e778fd3c1b871e9e3a94369335e9-Paper-Conference.pdf.

Luo donghao and wang xue. ModernTCN: A modern pure convolution structure for general time series analysis. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=vpJMJerXHU.

Emadeldeen Eldele, Mohamed Ragab, Zhenghua Chen, Min Wu, and Xiaoli Li. TSLANet: Rethinking transformers for time series representation learning. In Ruslan Salakhutdinov, Zico Kolter, Katherine Heller, Adrian Weller, Nuria Oliver, Jonathan Scarlett, and Felix Berkenkamp (eds.), *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 12409–12428. PMLR, 21–27 Jul 2024. URL https://proceedings.mlr.press/v235/eldele24a.html.

Jingru Fei, Kun Yi, Wei Fan, Qi Zhang, and Zhendong Niu. Amplifier: bringing attention to neglected low-energy components in time series forecasting. In *Proceedings of the Thirty-Ninth AAAI Conference on Artificial Intelligence and Thirty-Seventh Conference on Innovative Applications of Artificial Intelligence and Fifteenth Symposium on Educational Advances in Artificial Intelligence*, AAAI'25/IAAI'25/EAAI'25. AAAI Press, 2025. ISBN 978-1-57735-897-8. doi: 10.1609/aaai.v39i11.33267. URL https://doi.org/10.1609/aaai.v39i11.33267.

Aosong Feng, Jialin Chen, Juan Garza, Brooklyn Berry, Francisco Salazar, Yifeng Gao, Rex Ying, and Leandros Tassiulas. Efficient high-resolution time series classification via attention kronecker decomposition. *arXiv preprint arXiv:2403.04882*, 2024.

Huan He, Owen Queen, Teddy Koker, Consuelo Cuevas, Theodoros Tsiligkaridis, and Marinka Zitnik. Domain adaptation for time series under feature and label shifts. In *Proceedings of the 40th International Conference on Machine Learning*, ICML'23. JMLR.org, 2023.

Hansika Hewamalage, Christoph Bergmeir, and Kasun Bandara. Recurrent neural networks for time series forecasting: Current status and future directions. *International Journal of Forecasting*, 37 (1):388–427, 2021.

Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In Yoshua Bengio and Yann LeCun (eds.), *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015. URL http://arxiv.org/abs/1412.6980.

Feifei Kou, Jiahao Wang, Lei Shi, Yuhan Yao, Yawen Li, Suguo Zhu, Zhongbao Zhang, and Junping Du. CFPT: Empowering time series forecasting through cross-frequency interaction and periodic-aware timestamp modeling. In *Forty-second International Conference on Machine Learning*, 2025. URL https://openreview.net/forum?id=hHYjiOJFum.

Bhagwandas Pannalal Lathi and Roger A Green. *Signal processing and linear systems*, volume 2. Oxford university press Oxford, 1998.

Yuxuan Liang, Yutong Xia, Songyu Ke, Yiwei Wang, Qingsong Wen, Junbo Zhang, Yu Zheng, and Roger Zimmermann. Airformer: Predicting nationwide air quality in china with transformers. *Proceedings of the AAAI Conference on Artificial Intelligence*, 37(12):14329–14337, Jun. 2023. doi: 10.1609/aaai.v37i12.26676. URL https://ojs.aaai.org/index.php/AAAI/article/view/26676.

Bryan Lim and Stefan Zohren. Time-series forecasting with deep learning: a survey. *Philosophical Transactions of the Royal Society A*, 379(2194):20200209, 2021.

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, 2022a.

Shizhan Liu, Hang Yu, Cong Liao, Jianguo Li, Weiyao Lin, Alex X Liu, and Schahram Dustdar. Pyraformer: Low-complexity pyramidal attention for long-range time series modeling and forecasting. In *International Conference on Learning Representations*, 2021.

Yong Liu, Haixu Wu, Jianmin Wang, and Mingsheng Long. Non-stationary transformers: Exploring the stationarity in time series forecasting. In S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh (eds.), *Advances in Neural Information Processing Systems*, volume 35, pp. 9881–9893. Curran Associates, Inc., 2022b. URL https://proceedings.neurips.cc/paper_files/paper/2022/file/4054556fcaa934b0bf76da52cf4f92cb-Paper-Conference.pdf.

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

Zhiding Liu, Mingyue Cheng, Zhi Li, Zhenya Huang, Qi Liu, Yanhu Xie, and Enhong Chen. Adaptive normalization for non-stationary time series forecasting: A temporal slice perspective. In A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine (eds.), *Advances in Neural Information Processing Systems*, volume 36, pp. 14273–14292. Curran Associates, Inc., 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/file/2e19dab94882bc95ed094c4399cfda02-Paper-Conference.pdf.

Vinod Nair and Geoffrey E. Hinton. Rectified linear units improve restricted boltzmann machines. In *Proceedings of the 27th International Conference on International Conference on Machine Learning*, ICML'10, pp. 807–814, Madison, WI, USA, 2010. Omnipress. ISBN 9781605589077.

Yuqi Nie, Nam H Nguyen, Phanwadee Sinthong, and Jayant Kalagnanam. A time series is worth 64 words: Long-term forecasting with transformers. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=Jbdc0vTOcol.

Boris N. Oreshkin, Dmitri Carpov, Nicolas Chapados, and Yoshua Bengio. N-beats: Neural basis expansion analysis for interpretable time series forecasting. In *International Conference on Learning Representations*, 2020. URL https://openreview.net/forum?id=r1ecqn4YwB.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Köpf, Edward Yang, Zach DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. *PyTorch: an imperative style, high-performance deep learning library*. Curran Associates Inc., Red Hook, NY, USA, 2019.

Donald B Percival and Andrew T Walden. *Wavelet methods for time series analysis*, volume 4. Cambridge university press, 2000.

Xihao Piao, Zheng Chen, Taichi Murayama, Yasuko Matsubara, and Yasushi Sakurai. Fredformer: Frequency debiased transformer for time series forecasting. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, KDD '24, pp. 2400–2410, New York, NY, USA, 2024. Association for Computing Machinery. ISBN 9798400704901. doi: 10.1145/3637528.3671928. URL https://doi.org/10.1145/3637528.3671928.

Xiangfei Qiu, Jilin Hu, Lekui Zhou, Xingjian Wu, Junyang Du, Buang Zhang, Chenjuan Guo, Aoying Zhou, Christian S. Jensen, Zhenli Sheng, and Bin Yang. Tfb: Towards comprehensive and fair benchmarking of time series forecasting methods. *Proc. VLDB Endow.*, 17(9):2363–2377, May 2024. ISSN 2150-8097. doi: 10.14778/3665844.3665863. URL https://doi.org/10.14778/3665844.3665863.

Syama Sundar Rangapuram, Matthias W Seeger, Jan Gasthaus, Lorenzo Stella, Yuyang Wang, and Tim Januschowski. Deep state space models for time series forecasting. *Advances in neural information processing systems*, 31, 2018.

David Salinas, Valentin Flunkert, Jan Gasthaus, and Tim Januschowski. Deepar: Probabilistic forecasting with autoregressive recurrent networks. *International Journal of Forecasting*, 36(3): 1181–1191, 2020.

Rajat Sen, Hsiang-Fu Yu, and Inderjit S Dhillon. Think globally, act locally: A deep neural network approach to high-dimensional time series forecasting. *Advances in neural information processing systems*, 32, 2019.

Slawek Smyl. A hybrid method of exponential smoothing and recurrent neural networks for time series forecasting. *International Journal of Forecasting*, 36(1):75–85, 2020.

H. Sorensen, D. Jones, M. Heideman, and C. Burrus. Real-valued fast fourier transform algorithms. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 35(6):849–863, 1987. doi: 10.1109/TASSP.1987.1165220.

Artyom Stitsyuk and Jaesik Choi. xpatch: Dual-stream time series forecasting with exponential seasonal-trend decomposition. *Proceedings of the AAAI Conference on Artificial Intelligence*, 39 (19):20601–20609, Apr. 2025. doi: 10.1609/aaai.v39i19.34270. URL https://ojs.aaai.org/index.php/AAAI/article/view/34270.

Binh Tang and David S Matteson. Probabilistic transformer for time series analysis. *Advances in Neural Information Processing Systems*, 34:23592–23608, 2021.

Sean J. Taylor and Benjamin Letham. Forecasting at scale. *The American Statistician*, 72(1): 37–45, 2018. doi: 10.1080/00031305.2017.1380080. URL https://doi.org/10.1080/00031305.2017.1380080.

José F Torres, Dalil Hadjout, Abderrazak Sebaa, Francisco Martínez-Álvarez, and Alicia Troncoso. Deep learning for time series forecasting: a survey. *Big Data*, 9(1):3–21, 2021.

Oskar Triebe, Hansika Hewamalage, Polina Pilyugina, Nikolay Laptev, Christoph Bergmeir, and Ram Rajagopal. Neuralprophet: Explainable forecasting at scale, 2021.

Simon Vandenhende, Stamatios Georgoulis, Wouter Van Gansbeke, Marc Proesmans, Dengxin Dai, and Luc Van Gool. Multi-task learning for dense prediction tasks: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(7):3614–3633, 2022. doi: 10.1109/TPAMI.2021.3054719.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

Renzhuo Wan, Shuping Mei, Jun Wang, Min Liu, and Fan Yang. Multivariate temporal convolutional network: A deep neural networks approach for multivariate time series forecasting. *Electronics*, 8 (8):876, 2019.

Haichen Wang, Liu Yang, Xinyuan Zhang, Haomin Yu, Ming Li, and Jilin Hu. Adformer: Aggregation differential transformer for passenger demand forecasting, 2025. URL https://arxiv.org/abs/2506.02576.

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*, 2022.

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. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=7oLshfEIC2.

Qingsong Wen, Zhe Zhang, Yan Li, and Liang Sun. Fast robuststl: Efficient and robust seasonal-trend decomposition for time series with complex patterns. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pp. 2203–2213, 2020.

Qingsong Wen, Tian Zhou, Chaoli Zhang, Weiqi Chen, Ziqing Ma, Junchi Yan, and Liang Sun. Transformers in time series: a survey. In *Proceedings of the Thirty-Second International Joint Conference on Artificial Intelligence*, IJCAI '23, 2023. ISBN 978-1-956792-03-4. doi: 10.24963/ijcai.2023/759. URL https://doi.org/10.24963/ijcai.2023/759.

Shmuel Winograd. On computing the discrete fourier transform. *Proceedings of the National Academy of Sciences*, 73(4):1005–1006, 1976. doi: 10.1073/pnas.73.4.1005. URL https://www.pnas.org/doi/abs/10.1073/pnas.73.4.1005.

Haixu Wu, Jiehui Xu, Jianmin Wang, and Mingsheng Long. Autoformer: decomposition transformers with auto-correlation for long-term series forecasting. In *Proceedings of the 35th International Conference on Neural Information Processing Systems*, NIPS '21, Red Hook, NY, USA, 2021. Curran Associates Inc. ISBN 9781713845393.

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

Zonghan Wu, Shirui Pan, Guodong Long, Jing Jiang, Xiaojun Chang, and Chengqi Zhang. Connecting the dots: Multivariate time series forecasting with graph neural networks. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, KDD '20, pp. 753–763, New York, NY, USA, 2020. Association for Computing Machinery. ISBN 9781450379984. doi: 10.1145/3394486.3403118. URL https://doi.org/10.1145/3394486.3403118.

Zhijian Xu, Ailing Zeng, and Qiang Xu. Fits: Modeling time series with 10k parameters. In B. Kim, Y. Yue, S. Chaudhuri, K. Fragkiadaki, M. Khan, and Y. Sun (eds.), *International Conference on Representation Learning*, volume 2024, pp. 26295–26318, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/file/701251e1db4a2e4dd2ef23f5265d5936-Paper-Conference.pdf.

Tianzhu Ye, Li Dong, Yuqing Xia, Yutao Sun, Yi Zhu, Gao Huang, and Furu Wei. Differential transformer. In Y. Yue, A. Garg, N. Peng, F. Sha, and R. Yu (eds.), *International Conference on Representation Learning*, volume 2025, pp. 144–164, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/file/00b67df24009747e8bbed4c2c6f9c825-Paper-Conference.pdf.

Kun Yi, Qi Zhang, Wei Fan, Shoujin Wang, Pengyang Wang, Hui He, Ning An, Defu Lian, Longbing Cao, and Zhendong Niu. Frequency-domain MLPs are more effective learners in time series forecasting. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. URL https://openreview.net/forum?id=iif9mGCTfy.

Kun Yi, Jingru Fei, Qi Zhang, Hui He, Shufeng Hao, Defu Lian, and Wei Fan. Filternet: Harnessing frequency filters for time series forecasting. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024. URL https://openreview.net/forum?id=ugL2D9idAD.

Kun Yi, Qi Zhang, Wei Fan, Longbing Cao, Shoujin Wang, Hui He, Guodong Long, Liang Hu, Qingsong Wen, and Hui Xiong. A survey on deep learning based time series analysis with frequency transformation. In *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining V.2*, KDD '25, pp. 6206–6215, New York, NY, USA, 2025. Association for Computing Machinery. ISBN 9798400714542. doi: 10.1145/3711896.3736571. URL https://doi.org/10.1145/3711896.3736571.

Ailing Zeng, Muxi Chen, Lei Zhang, and Qiang Xu. Are transformers effective for time series forecasting? In *Proceedings of the Thirty-Seventh AAAI Conference on Artificial Intelligence and Thirty-Fifth Conference on Innovative Applications of Artificial Intelligence and Thirteenth Symposium on Educational Advances in Artificial Intelligence*, AAAI'23/IAAI'23/EAAI'23. AAAI Press, 2023. ISBN 978-1-57735-880-0. doi: 10.1609/aaai.v37i9.26317. URL https://doi.org/10.1609/aaai.v37i9.26317.

G.Peter Zhang. Time series forecasting using a hybrid arima and neural network model. *Neurocomputing*, 50:159–175, 2003. ISSN 0925-2312. doi: https://doi.org/10.1016/S0925-2312(01)00702-0. URL https://www.sciencedirect.com/science/article/pii/S0925231201007020.

Tianping Zhang, Yizhuo Zhang, Wei Cao, Jiang Bian, Xiaohan Yi, Shun Zheng, and Jian Li. Less is more: Fast multivariate time series forecasting with light sampling-oriented mlp structures. *arXiv preprint arXiv:2207.01186*, 2022.

Yunhao Zhang and Junchi Yan. Crossformer: Transformer utilizing cross-dimension dependency for multivariate time series forecasting. In *The Eleventh International Conference on Learning Representations*, 2022.

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

Tian Zhou, Ziqing Ma, Qingsong Wen, Liang Sun, Tao Yao, Wotao Yin, Rong Jin, et al. Film: Frequency improved legendre memory model for long-term time series forecasting. *Advances in Neural Information Processing Systems*, 35:12677–12690, 2022a.

Tian Zhou, Ziqing Ma, Qingsong Wen, Xue Wang, Liang Sun, and Rong Jin. FEDformer: Frequency enhanced decomposed transformer for long-term series forecasting. In Kamalika Chaudhuri, Stefanie Jegelka, Le Song, Csaba Szepesvari, Gang Niu, and Sivan Sabato (eds.), *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 27268–27286. PMLR, 17–23 Jul 2022b. URL https://proceedings.mlr.press/v162/zhou22g.html.

A Theoretical Analysis

A.1 Notion

Discrete Fourier Transform Given a sequence $x[n]$ with length N, the Discrete Fourier Transform (DFT) Winograd (1976) converts $x[n]$ into the frequency domain, and transforms it back using the inverse DFT (iDFT), which can be defined as:

$$\begin{aligned} \text{DFT}: \quad \mathcal{X}[k] &= \sum_{n=0}^{N-1} x[n] e^{-j(2\pi/N)kn},\ s.t.,\ k = 0, 1, ..., N-1 \\ \text{iDFT}: \quad x[n] &= \frac{1}{N} \sum_{k=0}^{N-1} \mathcal{X}[k] e^{j(2\pi/N)kn},\ \ s.t.,\ n = 0, 1, ..., N-1 \end{aligned} \tag{23}$$

where j is the imaginary unit and $\mathcal{X}[k]$ represents the spectrum of $x[n]$ at the frequency $\omega_k = 2\pi k/N$. The spectrum $\mathcal{X} \in \mathbb{C}^k$ consists of real parts $\text{Re} = \sum_{n=0}^{N-1} x[n] \cos(2\pi/N)kn \in \mathbb{R}^k$ and imaginary parts $\text{Im} = -\sum_{n=0}^{N-1} x[n] \sin(2\pi/N)kn \in \mathbb{R}^k$ as:

$$\mathcal{X} = \text{Re} + j\,\text{Im}\,. \tag{24}$$

The amplitude part A and phase part θ of $\mathcal{X}$ are defined as:

$$A = \sqrt{\text{Re}^2 + \text{Im}^2}. \tag{25}$$

$$\theta = \arctan(\frac{\text{Im}}{\text{Re}}). \tag{26}$$

The computational complexity of the DFT is typically $O(N^2)$ (Zhou et al. (2022b)). In practice, we use the Fast Fourier Transform (FFT) to efficiently compute the DFT of complex sequences, which reduces the computational complexity to $O(N \log N)$. Additionally, by employing the Real FFT (rFFT), we can compress an input sequence of N real numbers into a signal sequence in the complex frequency domain containing $N/2 + 1$ frequency components.

A.2 Proof

Assumption A.1 (Decomposition of Embedding)**.** Let e_i denote the embedding at any sample i. The embedded components e_i can be decomposed into a true signal term s_i and a noise term n_i:

$$e_i = s_i + n_i.$$

For two distinct segments from the embedding space, each associated with potentially different representation properties, their respective noise terms admit a further decomposition into a common-mode noise component $n_i^{(c)}$ and independent-mode noise components $\epsilon_i^{(1)}$ and $\epsilon_i^{(2)}$:

$$n_i^{(1)} = n_i^{(c)} + \epsilon_i^{(1)}, \quad n_i^{(2)} = n_i^{(c)} + \epsilon_i^{(2)},$$

where $\mathbb{E}[\epsilon_i^{(1)}] = \mathbb{E}[\epsilon_i^{(2)}] = 0$, $\text{Var}(\epsilon_i^{(1)}) = \text{Var}(\epsilon_i^{(2)}) = \sigma_\epsilon^2$, and $\epsilon_i^{(1)}$ and $\epsilon_i^{(2)}$ are independent.

We assume that the common-mode noise $n_i^{(c)}$ corresponds to a shared noise component in the embedding space, an assumption supported by previous studies (Ye et al., 2025; Wang et al., 2025). This shared noise often stems from systematic biases present in the input data (*e.g.*, stop words in NLP, background regions in spatio-temporal data, or certain frequency components in spectral (Eldele et al., 2024)). These features can introduce consistent bias into attention scores, as the softmax function is sensitive to large values even when they originate from irrelevant features.

Theorem A.2 (Non-zero Expectation of Common-mode Noise)**.** *Under realistic data distributions $\mathcal{D}$, the common-mode noise $n_i^{(c)}$ has a non-zero expectation:*

$$\mathbb{E}[n_i^{(c)}] \neq 0.$$

Proof. Let $n_i^{(c)} = f(X_i; \Theta)$, where $X_i \sim \mathcal{D}$ and f capture systematic biases with parameters Θ:

$$\mathbb{E}[n_i^{(c)}] = \mathbb{E}_{X_i \sim \mathcal{D}}[f(X_i; \Theta)].$$

Real-world data distributions $\mathcal{D}$ often contain biased features that are statistically frequent but non-causal or task-irrelevant Demirel & Holz (2025). Let $\phi_j(X_i)$ denote the j-th such feature function—each ϕ_j maps the input X_i to a scalar value representing the intensity of a particular spurious attribute. Through training, the model may develop dependence on these features. We therefore approximate the learned mapping $f(X_i; \Theta)$ as a linear combination of these feature functions:

$$f(X_i; \Theta) \approx \sum_j \alpha_j \phi_j(X_i),$$

where $\alpha_j > 0$ are weight coefficients. Since each ϕ_j is frequent, $\mathbb{E}_{X_i \sim \mathcal{D}}[\phi_j(X_i)] > 0$. By linearity of expectation:

$$\mathbb{E}_i[n_i^{(c)}] = \mathbb{E}\left[\sum_j \alpha_j \phi_j(X_i)\right] = \sum_j \alpha_j \mathbb{E}[\phi_j(X_i)] > 0,$$

unless $\alpha_j = 0$ for all j or $\mathbb{E}[\phi_j(X_i)] = 0$, both of which are uncommon in practice since the model leverages any available signal to minimize loss. □

Corollary A.3 (The Non-zero Expectation of Common-mode Noise After Training)**.** *During training, parameters Θ are updated via gradient descent to minimize the loss $\mathcal{L}$. However, if features $\phi_j(X_i)$ are correlated with the label (without causality), the model may learn to rely on them as shortcuts rather than suppressing their contribution He et al. (2023). Thus, α_j tends to remain positive, and $\mathbb{E}[n_i^{(c)}] > 0$ persists throughout optimization.*

Proposition A.4 (Adaptive Noise Suppression via Differential Embedding)**.** *The differential embedding mechanism, defined as $e_i^{(diff)} = e_i^{(1)} - \lambda e_i^{(2)}$ with a learnable parameter λ, provides adaptive suppression of common-mode noise. This results in gradient estimates $\hat{g} = g + \delta$ that exhibit a superior bias-variance trade-off for optimization, specifically:*

1. Suppression of Systematic Bias: The mechanism attenuates the bias introduced by common-mode noise: $\mathbb{E}[\delta] = (1 - \lambda^*)\mathbf{b}_g$, where $|1 - \lambda^*| < 1$.

2. Preservation of Beneficial Variance: It retains the variance from stochastic noise, which acts as a regularizer: $\mathrm{Var}(\delta) = (1 - \lambda^*)^2\sigma_c^2 + (1 + \lambda^{*2})\sigma_\epsilon^2$.

Here, λ^* is the value of λ that minimizes the training objective, $\mathbf{b}_g$ is the bias from common-mode noise, and σ_c^2, σ_ϵ^2 are the variances of the gradients of the common-mode and stochastic noise components, respectively.

Proof. We prove the two properties of the gradient noise δ by analyzing its expectation and variance.

The gradient noise δ arises from the backpropagation through the differential embedding. Consider the total gradient of the loss L with respect to the parameters Θ:

$$\frac{\partial L}{\partial \Theta} = \frac{\partial L}{\partial e_i^{(\text{diff})}} \cdot \frac{\partial e_i^{(\text{diff})}}{\partial \Theta}.$$

Under Assumption 1, we decompose the differential embedding into a true signal term and a noise term: $e_i^{(\text{diff})} = s_i^{(\text{diff})} + n_i^{(\text{diff})}$, where $n_i^{(\text{diff})} = (1 - \lambda)n_i^{(c)} + (\epsilon_i^{(1)} - \lambda\epsilon_i^{(2)})$. Substituting this decomposition yields:

$$\frac{\partial L}{\partial \Theta} = \frac{\partial L}{\partial e_i^{(\text{diff})}} \cdot \left(\frac{\partial s_i^{(\text{diff})}}{\partial \Theta} + \frac{\partial n_i^{(\text{diff})}}{\partial \Theta}\right).$$

The true gradient g is defined as $g = \frac{\partial L}{\partial e_i^{(\text{diff})}} \cdot \frac{\partial s_i^{(\text{diff})}}{\partial \Theta}$. This suggests that the gradient noise δ is:

$$\delta = \frac{\partial L}{\partial \Theta} - g = \frac{\partial L}{\partial e_i^{(\text{diff})}} \cdot \frac{\partial n_i^{(\text{diff})}}{\partial \Theta}.$$

1. Expectation of Gradient Noise ($\mathbb{E}[\delta]$):

Substituting the expression for the effective noise, $n_i^{(\text{diff})} = (1-\lambda)n_i^{(c)} + (\epsilon_i^{(1)} - \lambda\epsilon_i^{(2)})$, we get:

$$\delta = \frac{\partial L}{\partial e_i^{(\text{diff})}} \cdot \left[(1-\lambda)\frac{\partial n_i^{(c)}}{\partial \Theta} + \frac{\partial(\epsilon_i^{(1)} - \lambda\epsilon_i^{(2)})}{\partial \Theta} \right].$$

We now take the expectation of δ over the distributions of the stochastic noises $\epsilon_i^{(1)}$ and $\epsilon_i^{(2)}$. Under Assumption A.1, these stochastic noises are zero-mean, independent of the model parameters Θ and the common-mode noise $n_i^{(c)}$:

$$\mathbb{E}[\epsilon_i^{(k)}] = 0, \quad \mathbb{E}\left[\frac{\partial \epsilon_i^{(k)}}{\partial \Theta}\right] = 0, \quad \text{for } k = \{1, 2\}, \quad \text{and} \quad \mathbb{E}[\epsilon_i^{(k)} n_i^{(c)}] = 0.$$

Applying the linearity of expectation and leveraging these properties, the terms involving ϵ_i vanish:

$$\begin{aligned}\mathbb{E}[\delta] &= \mathbb{E}\left[\frac{\partial L}{\partial e_i^{(\text{diff})}} \cdot \left((1-\lambda)\frac{\partial n_i^{(c)}}{\partial \Theta}\right)\right] + \mathbb{E}\left[\frac{\partial L}{\partial e_i^{(\text{diff})}} \cdot \frac{\partial(\epsilon_i^{(1)} - \lambda\epsilon_i^{(2)})}{\partial \theta}\right] \\ &= (1-\lambda)\mathbb{E}\left[\frac{\partial L}{\partial e_i^{(\text{diff})}} \cdot \frac{\partial n_i^{(c)}}{\partial \Theta}\right] + 0.\end{aligned}$$

The remaining expectation term, $\mathbb{E}\left[\frac{\partial L}{\partial e_i^{(\text{diff})}} \cdot \frac{\partial n_i^{(c)}}{\partial \Theta}\right]$, is precisely the systematic bias $\mathbf{b}_g$ introduced into the gradient by the common-mode noise. At convergence, λ reaches a value λ^* that minimizes the loss. Since the loss is minimized by reducing the effect of $n_i^{(c)}$, the learning dynamics drive λ^* towards 1, ensuring $|1-\lambda^*| < 1$. Thus,

$$\mathbb{E}[\delta] = (1-\lambda^*)\mathbf{b}_g,$$

which demonstrates a reduction of the original bias by a factor of $|1-\lambda^*|$.

2. Variance of Gradient Noise ($\text{Var}(\delta)$):

We analyze the variance of δ:

$$\text{Var}(\delta) = \text{Var}\left(\frac{\partial L}{\partial e_i^{(\text{diff})}} \cdot \frac{\partial n_i^{(\text{diff})}}{\partial \Theta}\right).$$

For clarity, we define the shorthand:

$$A = \frac{\partial L}{\partial e_i^{(\text{diff})}} \cdot \frac{\partial n_i^{(c)}}{\partial \Theta}, \quad B = \frac{\partial L}{\partial e_i^{(\text{diff})}} \cdot \frac{\partial \epsilon_i^{(1)}}{\partial \Theta}, \quad C = \frac{\partial L}{\partial e_i^{(\text{diff})}} \cdot \frac{\partial \epsilon_i^{(2)}}{\partial \Theta}.$$

This allows us to express δ as:

$$\delta = (1-\lambda)A + B - \lambda C.$$

The variance is then:

$$\text{Var}(\delta) = \text{Var}\left((1-\lambda)A + B - \lambda C\right).$$

We assume A, B, and C are uncorrelated. This is justified by the independence of $n_i^{(c)}$ and $\epsilon_i^{(k)}$. Under this assumption, the covariance terms between A, B, and C are zero. Applying the variance property $\text{Var}(aX + bY) = a^2\text{Var}(X) + b^2\text{Var}(Y)$ for uncorrelated variables, we get:

$$\begin{aligned}\text{Var}(\delta) &= (1-\lambda)^2\text{Var}(A) + \text{Var}(B) + (-\lambda)^2\text{Var}(C) \\ &= (1-\lambda)^2\text{Var}(A) + \text{Var}(B) + \lambda^2\text{Var}(C).\end{aligned}$$

We now define the variances of these components:

$$\text{Var}(A) = \sigma_c^2, \quad \text{Var}(B) = \text{Var}(C) = \sigma_\epsilon^2.$$

Algorithm 1 AEA: Adaptive Energy Amplification for Robust Time Series Forecasting

Input: historical time series $X \in \mathbb{R}^{T \times C}$, forecasting horizon H, mixing ratio α, non-stationarity weight $\lambda_{\text{non-stat}}$, differential embedding dimension D, energy predictor embedding dimension D'
Output: forecasted results $\hat{Y} \in \mathbb{R}^{H \times C}$, total loss $\mathcal{L}$

Initialize learnable parameters: $M, W_e, b_e, W_p, b_p, W_1, b_1, W_2, b_2, \lambda_1, \lambda_2$

Spectral Mirroring (Section 4.2)
$\mathcal{X} \leftarrow \text{rFFT}(X)$ ▷ Transform to frequency domain
$\mathcal{X}_{\text{reverse}}[k] \leftarrow \mathcal{X}[F-1-k], \quad \forall k \in [0, F-1]$ ▷ Spectrum reverse
$\mathcal{X}_{\text{scaled}} \leftarrow \mathcal{X}_{\text{reverse}} \odot M$ ▷ Adaptive scaling per frequency/channel
Phase-preserving mixing to avoid distortion:
for each frequency k, channel c **do**
  $\theta_1, \theta_2 \leftarrow \angle(\mathcal{X}[k,c]), \angle(\mathcal{X}_{\text{scaled}}[k,c])$
  $\Delta\theta \leftarrow (\theta_1 - \theta_2) \mod 2\pi$ ▷ Circular difference modulo 2π
  $\Delta\theta_{\text{adjusted}} \leftarrow \begin{cases} \Delta\theta - 2\pi & \text{if } \Delta\theta > \pi \\ \Delta\theta & \text{otherwise} \end{cases}$ ▷ Shortest angular path
  $\theta_{\text{mix}} \leftarrow \theta_1 + \Delta\theta_{adjusted}$ ▷ Phase mixing (Equation 4)
  $A_{\text{mix}} \leftarrow \alpha \cdot |\mathcal{X}[k,c]| + (1-\alpha) \cdot |\mathcal{X}_{\text{scaled}}[k,c]|$
  $\mathcal{X}_{\text{enhanced}}[k,c] \leftarrow A_{\text{mix}} \cdot e^{j\theta_{\text{mix}}}$ ▷ Reconstruct enhanced spectrum
end for

Differential Embedding (Section 4.3)
$E_1 \| E_2 \leftarrow W_e \cdot \mathcal{X}_{\text{enhanced}} + b_e$ ▷ Project to complex embedding space
$E_1' \leftarrow E_1 - \lambda_1 \cdot E_2, \quad E_2' \leftarrow E_2 - \lambda_2 \cdot E_1$ ▷ Differential operation for noise suppression
$E' \leftarrow \text{Concat}(E_1', E_2')$ ▷ Denoised embedding (Proposition A.4)
$\mathcal{X}_{\text{denoised}} \leftarrow W_p \cdot E' + b_p$ ▷ Project back to frequency domain

---

Energy Prediction & Forecasting (Section. 4.4)
$X_{\text{denoised}} \leftarrow \text{irFFT}(\mathcal{X}_{\text{denoised}})$ ▷ Denoised input for base model
$\hat{Y}_{\text{denoised}} \leftarrow \text{BaseModel}(X_{\text{denoised}})$ ▷ Any forecasting backbone
$\mathcal{Y}_{\text{denoised}} \leftarrow \text{rFFT}(\hat{Y}_{\text{denoised}})$
$\mathcal{E} \leftarrow W_1 \cdot \mathcal{X}_{\text{scaled}} + b_1$ ▷ Encode historical spectral context
$\mathcal{Y}_{\text{adjusted}} \leftarrow W_2 \cdot \text{Concat}(\mathcal{E}, \mathcal{Y}_{\text{denoised}}) + b_2$ ▷ Spectral alignment
$\hat{Y} \leftarrow \text{irFFT}(\mathcal{Y}_{\text{adjusted}})$ ▷ Final consistent prediction

---

Multi-Task Optimization (Section 4.5)
$\mathcal{L}_{\text{forecast}} \leftarrow \text{MSE}(\hat{Y}, Y)$ ▷ Forecasting loss
$\mathcal{L}_{\text{non-stat}} \leftarrow \sqrt{\text{Var}_{\mathbf{x} \sim \mathcal{B}}(|E'|)}$ ▷ Non-stationarity regularization (Equation 14)
$\mathcal{L} \leftarrow \mathcal{L}_{\text{forecast}} + \lambda_{\text{non-stat}} \cdot \mathcal{L}_{\text{non-stat}}$
return $\hat{Y}, \mathcal{L}$

The equality $\text{Var}(B) = \text{Var}(C)$ stems from the assumption that $\epsilon_i^{(1)}$ and $\epsilon_i^{(2)}$ are identically distributed. Substituting these definitions and evaluating at the optimal $\lambda = \lambda^*$ yields:

$$\text{Var}(\delta) = (1-\lambda^*)^2 \sigma_c^2 + (1+\lambda^{*2}) \sigma_\epsilon^2.$$

This final expression shows that the mechanism suppresses the harmful variance from common-mode noise by a factor of $(1-\lambda^*)^2$ while preserving and even amplifying the beneficial stochastic noise by a factor of $(1+\lambda^{*2}) \geq 1$.

Thus, the adaptive parameter λ^* optimally balances the bias-variance trade-off in the gradient estimates, leading to more robust and effective optimization. □

B MORE DETAILS

B.1 MORE DETAILS OF METRICS

We use Mean Squared Error (MSE) and Mean Absolute Error (MAE) as evaluation metrics. Given the ground truth values $\mathbf{X}_i$ and the predicted values $\hat{\mathbf{X}}_i$, these metrics are defined as follows:

$$\text{MSE} = \frac{1}{N}\sum_{i=1}^{N}(\mathbf{X}_i - \hat{\mathbf{X}}_i)^2, \quad \text{MAE} = \frac{1}{N}\sum_{i=1}^{N}|\mathbf{X}_i - \hat{\mathbf{X}}_i|,$$

where N is the total number of predictions.

B.2 MORE DETAILS OF FIGURE 1

Figure 1(a): Spectral Bias. To evaluate the spectral bias of base models, we conduct frequency masking experiments during inference. Given an input time series $X \in \mathbb{R}^{T\times C}$, we compute its frequency representation via rFFT:

$$\mathcal{X} = \text{rFFT}(X), \quad \mathcal{X} \in \mathbb{C}^{F\times C},$$

where $F = \lfloor T/2 \rfloor + 1$ is the number of frequency components. We then create two masked variants: Low-frequency mask: Set the lower 50% of frequencies to zero:

$$\mathcal{X}_{\text{low-mask}}[k] = \begin{cases} 0, & \text{for } k = 0, 1, \ldots, \lfloor F/2 \rfloor, \\ \mathcal{X}[k], & \text{otherwise.} \end{cases}$$

High-frequency mask: Set the higher 50% of frequencies to zero:

$$\mathcal{X}_{\text{high-mask}}[k] = \begin{cases} \mathcal{X}[k], & \text{for } k = 0, 1, \ldots, \lfloor F/2 \rfloor, \\ 0, & \text{otherwise.} \end{cases}$$

Each masked spectrum is converted back to the time domain via inverse rFFT, and forecasting performance is evaluated relative to the unmasked baseline.

Results indicate that masking low-frequency components leads to a severe performance degradation (MSE increase > 100%), whereas masking high-frequency components has a negligible effect (MSE increase < 5%). This pronounced discrepancy confirms that baseline models exhibit a strong reliance on low-frequency information while overlooking high-frequency signals, underscoring a fundamental spectral bias in existing forecasting architectures.

Figure 1(b): Indiscriminate Amplification. To assess robustness to high-frequency noise, we compare vanilla amplification methods with our AEA-enhanced version under two noise injection scenarios on the ETTh1 dataset:

(1) "Gaussian noise to high freq: before amplification" - Noise is injected directly into the high-frequency bands of the input signal before spectral mirroring:

$$\mathcal{X}_{\text{noise-before}}[k] = \begin{cases} \mathcal{X}[k] + \mathcal{N}(0, \sigma^2), & \text{for } k > \lfloor F/2 \rfloor, \\ \mathcal{X}[k], & \text{otherwise,} \end{cases}$$

where $\mathcal{N}(0, \sigma^2)$ denotes Gaussian noise with zero mean and variance σ^2.

(2) "Gaussian noise to high freq: after amplification" - The same noise is introduced into the high-frequency components of the enhanced spectrum output by the Spectral Mirroring module:

$$\mathcal{X}_{\text{enhanced}} = \text{Spectral Mirroring}(\mathcal{X})$$

$$\mathcal{X}_{\text{noise-after}}[k] = \begin{cases} \mathcal{X}_{\text{enhanced}}[k] + \mathcal{N}(0, \sigma^2), & \text{for } k > \lfloor F/2 \rfloor, \\ \mathcal{X}_{\text{enhanced}}[k], & \text{otherwise.} \end{cases}$$

Performance degradation is measured as a relative increase in MSE compared to the "None" baseline.

Vanilla amplification methods suffer severe performance degradation under both noise conditions. This demonstrates that indiscriminate amplification amplifies noise alongside signals, compromising robustness. In contrast, AEA maintains stable forecasting accuracy, demonstrating that its differential embedding mechanism effectively suppresses common-mode noise while preserving discriminative high-frequency content. The significant performance gap highlights AEA's superior noise robustness compared to existing amplification approaches.

B.3 More Details of Datasets

Table 4: Dataset detailed descriptions. The dataset size is organized into (Train, Validation, Test).

Datasets	Dim	Prediction Length	Dataset Size	Frequency	Information
ETTh1	7	$\{96, 192, 336, 720\}$	$(8545, 2881, 2881)$	15 min	Electricity
ETTh2	7	$\{96, 192, 336, 720\}$	$(8545, 2881, 2881)$	15 min	Electricity
ETTm1	7	$\{96, 192, 336, 720\}$	$(34465, 11521, 11521)$	15 min	Electricity
ETTm2	7	$\{96, 192, 336, 720\}$	$(34465, 11521, 11521)$	15 min	Electricity
Electricity	321	$\{96, 192, 336, 720\}$	$(18317, 2633, 5261)$	1 hour	Electricity
Exchange	8	$\{96, 192, 336, 720\}$	$(5120, 665, 1422)$	1 day	Exchange rate
Weather	21	$\{96, 192, 336, 720\}$	$(36792, 5271, 10540)$	10 min	Weather
Traffic	862	$\{96, 192, 336, 720\}$	$(12185, 1757, 3509)$	1 hour	Transportation

We evaluate our method on eight established time series benchmarks for long-term forecasting. Dataset statistics are summarized in Table 4, with detailed descriptions provided below:

(1) The **ETT** (Electricity Transformer Temperature) dataset (Zhou et al., 2021) records temperature and load data from power transformers in two Chinese regions between 2016 and 2018. It includes two temporal resolutions: ETTh (hourly) and ETTm (15-minute intervals).

(2) The **Electricity** dataset (Wu et al., 2023) comprises hourly power consumption measurements (kWh) from 321 customers. Collected from the UCL repository and spanning 2012-2014, it captures residential and commercial energy usage patterns.

(3) The **Weather** dataset (Wu et al., 2023) contains 21 meteorological variables recorded at 10-minute intervals throughout 2020 in Germany. Parameters include temperature, humidity, pressure, and visibility, providing comprehensive environmental monitoring.

(4) The **Exchange** dataset (Wu et al., 2023) tracks daily currency values for eight major economies relative to the US dollar over 1990-2016. This 26-year series reflects global financial dynamics and macroeconomic trends.

(5) The **Traffic** dataset (Wu et al., 2023) provides hourly occupancy rates from 862 sensors on San Francisco Bay Area freeways during 2015-2016. It captures urban mobility patterns and congestion dynamics.

B.4 More Details of Experiment

We make our codes publicly available, including implementations of all base models and the proposed AEA framework, to ensure reproducibility. The backbone implementations are adapted from their official GitHub repositories, with reference to the TimesNet codebase (Wu et al., 2023). All experiments were conducted using the following unified settings: batch size of 32, learning rate of 0.0005, random seed fixed at 2021, and Adam optimizer (Kingma & Ba, 2015). Each run was trained for 10 epochs with early stopping (patience = 3) to prevent overfitting.

C More Results

C.1 More Results of Hyperparameter Sensitivity Analysis

We conduct sensitivity analysis on four key hyperparameters using MSE as the evaluation metric, with DLinear as the backbone model under a forecasting horizon of 96.

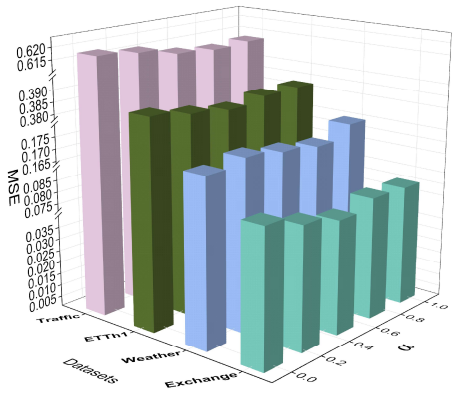


Figure 3: Performance *w.r.t.* different amplitude mixing ratio α with DLinear as backbone under a horizon of 96.

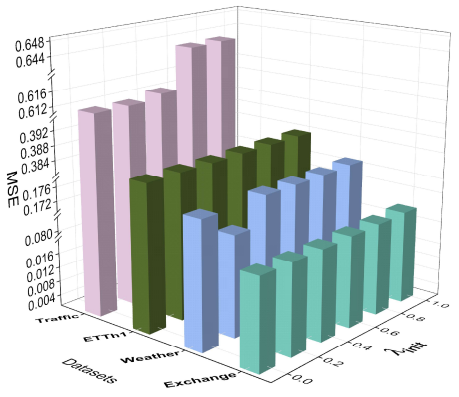


Figure 4: Performance *w.r.t.* different differential scaling initialization λ_{init} with DLinear as backbone under a horizon of 96.

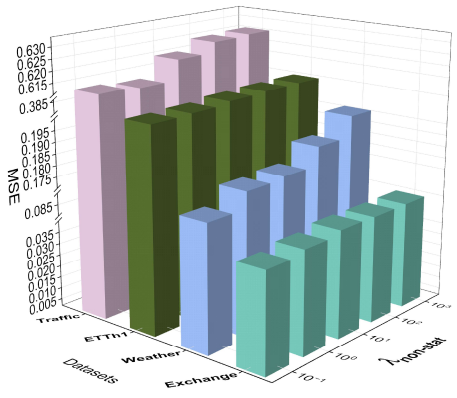


Figure 5: Performance *w.r.t.* different non-stationarity weight $\lambda_{non-stat}$ with DLinear as backbone under a horizon of 96.

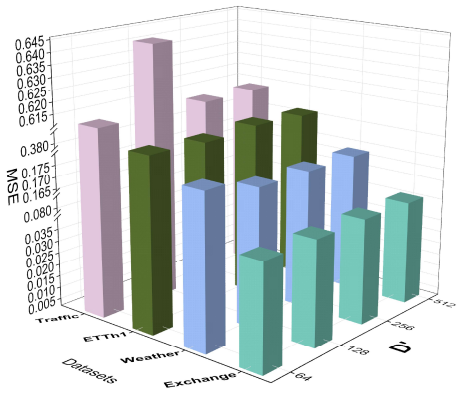


Figure 6: Performance *w.r.t.* different energy predictor dimension D' with DLinear as backbone under a horizon of 96.

Sensitivity of Amplitude Mixing Ratio α. We investigate the impact of the amplitude mixing ratio α in Spectral Mirroring, which controls the balance between original and mirrored spectra amplitudes. As shown in Figure 4, performance remains stable across $\alpha \in [0, 1]$, with $\alpha = 0.4$ achieving optimal or near-optimal results on all four datasets. This suggests that equal weighting offers the optimal balance between signal enhancement and distortion avoidance. The minimal performance variation ($< 1\%$ MSE difference across values) demonstrates the robustness of our amplitude mixing strategy to this hyperparameter.

Sensitivity of Differential Scaling Initialization λ_{init}. The initialization of differential scaling parameters λ_1 and λ_2 is crucial for stable training. As shown in Figure 5, performance is largely insensitive to $\lambda_{init} \in [0, 1]$, with fluctuations within 5% across datasets. The Softplus constraint ensures that positive values are maintained throughout the optimization process, while the learning mechanism allows for adaptation to dataset-specific noise characteristics. We use $\lambda_{init} = 0.2$ as the default for consistent convergence.

Sensitivity of Non-stationarity Weight $\lambda_{non-stat}$. The regularization weight $\lambda_{non-stat}$ balances forecasting accuracy with representation stability. As shown in Figure 6, extreme values (≥ 100) cause noticeable degradation, while moderate settings ($0.1 - 1.0$) maintain stable performance. This confirms the importance of the non-stationarity loss for robust learning, while demonstrating that a wide range of values provides effective regularization. We set $\lambda_{non-stat} = 0.1$ as the default balanced configuration.

Sensitivity of Energy Predictor Dimension D'. As shown in Figure 3, the embedding dimension D' in the Energy Predictor shows minimal impact on performance, with differences $< 5\%$ across $D' \in [64, 512]$. This indicates that even compact representations ($D' = 64$) effectively capture the spectral mapping between enhanced and original distributions. The consistency across dimensions confirms the efficiency of our frequency-domain alignment approach.