Conditional Denoising Meets Polynomial Modeling: A Flexible Decoupled Framework for Time Series Forecasting

Jintao Zhang[1], Mingyue Cheng[1]*, Xiaoyu Tao[1], Zhiding Liu[1], Daoyu Wang[1]
[1]State Key Laboratory of Cognitive Intelligence, University of Science and Technology of China
zjttt@mail.ustc.edu.cn, mycheng@ustc.edu.cn, {txytiny, zhiding, wdy030428}@mail.ustc.edu.cn

Abstract

Time series forecasting models are becoming increasingly prevalent due to their critical role in decision-making across various domains. However, most existing approaches represent the coupled temporal patterns, often neglecting the distinction between their specific components. In particular, fluctuating patterns and smooth trends within time series exhibit distinct characteristics. In this work, to model complicated temporal patterns, we propose a Conditional Denoising Polynomial Modeling (CDPM) framework, where probabilistic diffusion models and deterministic linear models are trained end-to-end. Instead of modeling the coupled time series, CDPM decomposes it into trend and seasonal components for modeling them separately. To capture the fluctuating seasonal component, we employ a probabilistic diffusion model based on statistical properties from the historical window. For the smooth trend component, a module is proposed to enhance linear models by incorporating historical dependencies, thereby preserving underlying trends and mitigating noise distortion. Extensive experiments conducted on six benchmarks demonstrate the effectiveness of our framework, highlighting the potential of combining probabilistic and deterministic models. Our code is available at https://github.com/zjt-gpu/CDPM.

1 Introduction

Time series forecasting is a fundamental task across various domains, playing a crucial role in driving informed decision-making in industries such as finance, healthcare, and manufacturing [Kashpruk *et al.*, 2023]. It involves analyzing historical patterns to model the dynamic evolution of underlying processes and predict future trends [Cheng *et al.*, 2025a]. Accurate time series forecasting is essential for tasks such as demand prediction, resource allocation, enabling organizations to anticipate future needs and optimize operations [Kilcioglu *et al.*, 2017; Liu *et al.*, 2024].

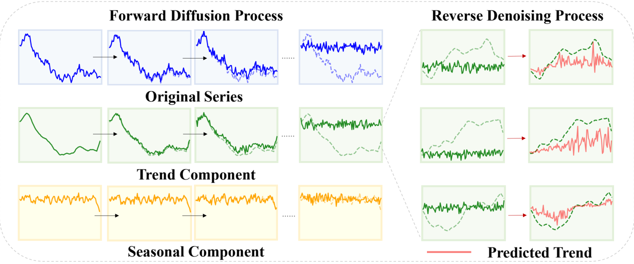

Figure 1: The forward diffusion process iteratively adds noise to the original time series, which causes the trend components to be overshadowed as the noise increasingly dominates.

Over the years, various approaches have been proposed to address this challenge [Cheng *et al.*, 2025b], ranging from classical statistical models, such as Autoregressive Integrated Moving Average (ARIMA) [Asteriou and Hall, 2011], to machine learning techniques [Masini *et al.*, 2023], and advanced deep learning models, including recurrent neural networks (RNNs) [Lin *et al.*, 2023; Lai *et al.*, 2018], convolutional neural networks (CNNs) [Bai *et al.*, 2018; Koprinska *et al.*, 2018], and transformer-based models [Zhou *et al.*, 2021; Wu *et al.*, 2021]. Large language models (LLMs) such as TimeLLM and LLMTIME [Jin *et al.*, 2023; Gruver *et al.*, 2024] have demonstrated the potential in time series forecasting. In addition to their success, probabilistic models have emerged as promising alternatives, offering a principled framework for quantifying uncertainty and modeling complex distributions. Diffusion models [Rasul *et al.*, 2021; Shen and Kwok, 2023], becoming the leading paradigm for probabilistic models originally developed for generation, have proven effective in time series forecasting due to their ability to capture intricate stochastic processes.

Diffusion-based methods [Lee *et al.*, 2024] have shown great promise in time series forecasting. However, we argue that their naive application of noise addition and denoising overlooks the heterogeneous nature of temporal dynamics. In particular, these methods uniformly add noise across the entire sequence and apply denoising without distinguishing between different temporal components. However, time series inherently exhibits diverse dynamics, including trends and seasonal patterns, which require different treatment. Trends patterns, reflecting gradual shifts over extended periods, and

[*] Corresponding author

seasonal patterns, characterized by periodic fluctuations, possess different structural properties. These distinct patterns require differentiated treatment. Such indiscriminate processing can distort essential features, especially intrinsic trends and fluctuations, thereby degrading forecasting performance. As shown in Fig.1, this indiscriminate approach disrupts critical trend information during denoising, hindering trend recovery and reducing accuracy.

Typically, a single model often struggles to capture the subtle and dynamic interplay between trends and variations, which can significantly affect the model's overall forecasting ability. Diffusion models, which excel at modeling complex dynamic processes, are more suitable for handling components characterized by large fluctuations and high-frequency variations in time series. In contrast, deterministic models, such as models based on linear layers, learn features from historical patterns and are particularly effective at identifying regular patterns, making them better suited for modeling components with stable trends. Therefore, by strategically leveraging the strengths of both model types and tailoring them to the distinct characteristics of time series, we can more fully capture the dynamic variations and enhance the final forecasting results.

To overcome these challenges, we propose a novel Conditional Denoising Polynomial Modeling (CDPM) framework that leverages the inductive bias of time series decomposition. By separating the time series into trend and seasonal components, we enable decoupled modeling tailored to the characteristics of each component. Specifically, we employ a carefully designed Conditional Denoising Seasonal Module (CDSM) to capture the high-frequency seasonal component, introducing an innovative denoising strategy conditioned on statistical properties derived from historical window. Currently, we model the intrinsic trend component using the polynomial trend module (PTM), which is adept at capturing gradual changes. Additionally, we reformulate the Improved Evidence Lower Bound (ELBO) to facilitate joint training of the CDSM and PTM. This decoupling method of our framework improves the ability to capture temporal dependencies and complex patterns. The key contributions of this work are:

- We introduce a novel framework that enables the decoupled modeling of trend and seasonal components, providing new insights into time series forecasting.
- We design tailored methods and reformulate ELBO for each component-using CDSM for the seasonal component and PTM for the trend component—to effectively capture their distinct behaviors.
- We conduct comprehensive experiments on benchmark datasets to validate effectiveness of our framework, we also present the rationale for modeling different components with several case studies.

2 Related Work

2.1 Deterministic Forecasting

Deterministic models map deterministic inputs to deterministic outputs by learning consistent patterns within time series, making them crucial for capturing predictable temporal relationships. Recurrent neural networks (RNNs), such as LSTNet [Lai *et al.*, 2018] and SegRNN [Lin *et al.*, 2023], became popular for sequential dependencies, while convolutional neural networks (CNNs) [Cheng *et al.*, 2024] like TCN [Bai *et al.*, 2018] and SCINet [Liu *et al.*, 2022] excelled at capturing both local and global temporal features. Graph-based models, such as MTGNN [Wu *et al.*, 2020] and FourierGNN [Yi *et al.*, 2024], improved forecasting by modeling interdependencies between time series. More recently, Transformer-based models like Informer [Zhou *et al.*, 2021] and Autoformer [Wu *et al.*, 2021] have utilized self-attention mechanisms to efficiently capture long-range dependencies, excelling in high-dimensional and long-horizon tasks. Surprisingly, well-tuned linear models such as DLinear [Zeng *et al.*, 2023] and TSMixer [Ekambaram *et al.*, 2023] have demonstrated strong performance by effectively capturing long-term trends, often outperforming more complex architectures and showcasing their efficiency. Despite advances, deterministic models often struggle with volatile fluctuations, limiting their ability to model complex time series dynamics.

2.2 Diffusion-Based Time Series Forecasting

Probabilistic models offer uncertainty estimation and the ability to model temporal distributions. These models, including Variational Autoencoders (VAEs)[Kingma and Welling, 2022], Generative Adversarial Networks (GANs)[Goodfellow *et al.*, 2014], and flow-based models [Rezende and Mohamed, 2016], are designed to capture high-dimensional distribution. Recently, Denoising Diffusion Probabilistic Models (DDPMs) have emerged as a promising paradigm for generative modeling, demonstrating their potential in various time series applications. DDPMs have advanced time series forecasting by capturing complex temporal patterns. Early models like TimeGrad [Rasul *et al.*, 2021] introduced autoregressive denoising with Langevin sampling for multivariate predictions. TSDiff [Kollovieh *et al.*, 2024] improved short-term accuracy through self-guiding mechanisms, while score-based models such as ScoreGrad [Yan *et al.*, 2021] applied stochastic differential equations (SDEs) for continuous-time forecasting, expanding diffusion to fluctuatingly sampled data. Conditional models like TimeDiff [Shen and Kwok, 2023] incorporated external information to guide the diffusion process, outperforming traditional methods. Latent Diffusion Models (LDMs)[Rombach *et al.*, 2022] enhanced efficiency by performing diffusion in lower-dimensional spaces. This approach was further demonstrated by LDCast and Latent Diffusion Transformers (LDT)[Feng *et al.*, 2024], which improved precipitation and time series forecasting. Models like DSPD and CSPD [Bilos *et al.*, 2023] applied diffusion to anomaly detection and interpolation.

Despite the advancements in both deterministic and probabilistic forecasting approaches, existing research typically applies each paradigm separately to represent time series, with limited efforts to integrate them into a unified modeling framework. However, effectively capturing the diverse characteristics of time series, such as the smooth trend and fluctuating seasonal components, requires leveraging both deterministic and probabilistic models.

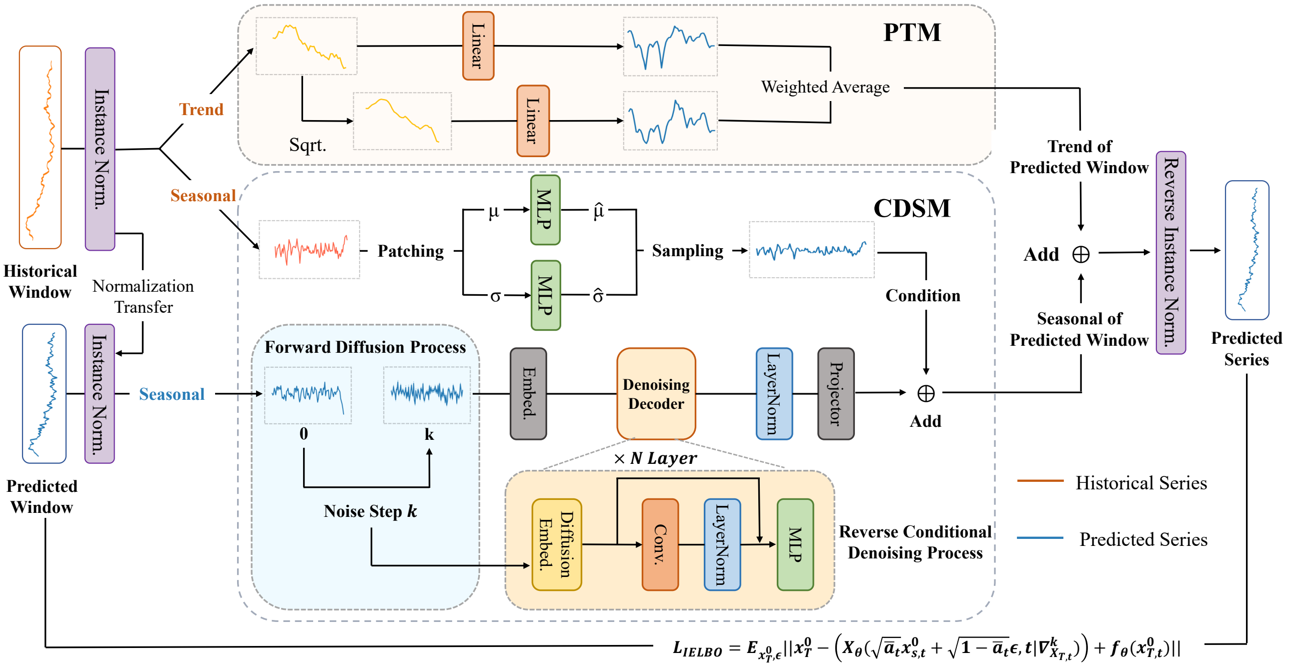

Figure 2: Illustration of the proposed Conditional Denoising Polynomial Modeling (CDPM) framework. The Conditional Denoising Seasonal Module (CDSM) refines the seasonal component through a diffusion process conditioned on historical patterns, while the Polynomial Trend Module (PTM) models the trend component to capture different patterns.

3 Problem Statement

Time series represent numerical sequence ordered chronologically, typically recorded at regular temporal intervals. Formally, consider a multivariate time series $X_{L+T} \in \mathbb{R}^{(L+T)\times d}$, where L and T denotes the length of the historical and the length of the predicted window respectively, with d channels recorded for each time step. The objective of time series forecasting is to train a model G that learns to map the historical series X_L to the predicted series X_T. The model leverages the information embedded in the historical series to predict future series, aiming to capture both temporal dependencies for accurate forecasting.

4 Methodology

In this section, we formalize the time series forecasting problem and present the proposed Conditional Denoising Polynomial Modeling (CDPM) framework, comprising two core components: the Conditional Denoising Seasonal Module (CDSM) and the Polynomial Trend Module (PTM).

4.1 Overall Framework

The CDPM addresses time series forecasting by independently modeling trend and seasonal patterns, thereby capturing distinct temporal behaviors more effectively. As depicted in Fig. 2, CDPM consists of two main modules: CDSM and PTM. The CDSM employs a denoising diffusion process conditioned on statistical properties from the historical window to accurately capture and refine seasonal variation patterns. In parallel, the PTM models trends using polynomial linear layers, providing flexibility in handling both linear and nonlinear trends. By decoupling and specifically modeling these components, CDPM enhances forecasting accuracy, offering an effective solution for capturing the complex temporal dynamics inherent in time series. We also reformulate the Evidence Lower Bound (ELBO) to better jointly train the PTM and CDSM. By handling trend and seasonal components independently, CDPM allows each module to focus on its respective dynamics, leading to enhanced overall performance.

4.2 Temporal Feature Structuring

Instance Normalization. Instance normalization across both historical and predicted windows is essential for maintaining the stability of the CDPM. We normalize the historical window by subtracting the mean and dividing by the standard deviation [Kim *et al.*, 2021], and transfer the same statistics to normalize the predicted window during training. Specifically, x_L is normalized to x_L^0, and x_T is normalized to x_T^0 using the statistics derived from x_L. This ensures alignment between the windows and preserves both the temporal structure and stability in the forecasting process. The subsequent denormalization step restores the predicted values to their original scale, ensuring consistency with the input data distribution.

Decomposition Operation. Decomposition is an important approach to achieving decoupling in time series analysis, as it allows for the separation of trend components and seasonal components [Cleveland *et al.*, 1990], enabling each characteristic to be handled independently. We decompose both historical X_L^0 and predicted X_T^0 windows into trend and seasonal components. The trend component $X_{L,t}^0$ of historical window is estimated using a moving average:

$$X^0_{L,t} = \mathrm{Avg}(X^0_L), \tag{1}$$

where Avg denotes the moving average.

$$X^0_{L,s} = X^0_L - X^0_{L,t}, \tag{2}$$

where the seasonal component $X^0_{L,s}$ is the residual after subtracting the trend. Thus, we obtain the trend component $X^0_{L,t}$ and seasonal component $X^0_{L,s}$ of historical window.

Similarly, we compute $X^0_{T,t}$ and $X^0_{T,s}$. Decomposition operation isolates the distinct characteristics of time series.

4.3 Conditional Denoising Seasonal Module

Modeling fluctuating seasonal components in time series is challenging due to their intricate patterns. We propose CDSM, based on a denoising diffusion model. By incorporating historical statistical properties as conditional inputs, CDSM captures seasonal patterns more effectively.

Forward Diffusion Process. In the forward diffusion process, noise is progressively added to the seasonal component. At each diffusion step k, the noisy seasonal series is:

$$X^k_{T,s} = \sqrt{\bar{\alpha}_k}\, x^0_{T,s} + \sqrt{1-\bar{\alpha}_k}\, \epsilon, \tag{3}$$

where $X^k_{T,s}$ is the seasonal component at step k, $x^0_{T,s}$ is the original seasonal component, $\epsilon \sim \mathcal{N}(0, I)$ is Gaussian noise, and $\bar{\alpha}_k \in [0, 1]$ is the noise schedule. As k increases, the seasonal component becomes more noise-dominated.

Reverse Conditional Denoising Process. The denoising process starts by embedding $X^k_{T,s}$:

$$\mathbf{h} = \mathrm{Emb}(X^k_{T,s}), \tag{4}$$

where $\mathrm{Emb}(\cdot)$ is a convolutional MLP that maps the noisy sequence to a hidden dimension while capturing local temporal features. Sinusoidal positional encodings for diffusion step t are integrated via Adaptive Layer Normalization (AdaLN):

$$\mathrm{AdaLN}(\mathbf{h}) = \alpha_t \cdot \mathrm{LayerNorm}(\mathbf{h}) + \beta_t, \tag{5}$$

where α_t and β_t are learnable parameters projected from the diffusion step encoding.

The decoder composed of convolutional layers and MLPs, captures global dependencies and feature interactions. Convolutional layers extract fine-grained local temporal variations, while MLPs model cross-dimensional relationships. Layer normalization and a projection layer map the output back to the original feature space.

To enhance denoising process, we incorporate conditional information from the historical seasonal component $X^0_{L,s}$.

$$\mu_{L,i} = \frac{1}{H} \sum_{h=1}^{H} x_{L,s,i,h}, \tag{6}$$

$$\sigma^2_{L,i} = \frac{1}{H} \sum_{h=1}^{H} \left(x_{L,s,i,h} - \mu_{L,i}\right)^2, \tag{7}$$

where, for each patch $P_{L,i}$, $\mu_{L,i}$ represents the local mean and $\sigma^2_{L,i}$ denotes the variance. The historical data is divided into patches P of length H.

These statistics capture seasonal variability. The predicted mean $\hat{\mu}_{T,i}$ and variance $\hat{\sigma}^2_{T,i}$ for the target window $P_{T,i}$ are computed using two MLPs:

$$\hat{\mu}_{T,i} = \mathrm{MLP}_\mu(\mu_{L,i}), \tag{8}$$

$$\hat{\sigma}^2_{T,i} = \mathrm{MLP}_\sigma(\sigma^2_{L,i}), \tag{9}$$

where $\mathrm{MLP}_\mu(\cdot)$ and $\mathrm{MLP}_\sigma(\cdot)$ capture the mean and variance of the historical window features, respectively. These predicted statistics are subsequently used in the denoising process via Gaussian sampling:

$$\nabla^{k-1}_{\mathbf{x}_{T,s}} = \hat{\mu}_{T,i} + \hat{\sigma}_{T,i} \cdot z, \tag{10}$$

where $z \sim \mathcal{N}(0, I)$. This enables the model to incorporate stochastic variability while aligning with historical patterns.

Finally, the denoised output $\hat{x}^0_{T,s}$ is combined with the conditional information through a weighted summation:

$$X_\theta\left(X^k_{T,s}, t \mid \nabla^{k-1}_{\mathbf{x}_{T,s}}\right) = \rho_1 \hat{x}^0_{T,s} + \rho_2 \nabla^{k-1}_{\mathbf{x}_{T,s}}, \tag{11}$$

where ρ_1 and ρ_2 are learnable parameters that balance the denoised series and conditional information, ensuring effective seasonal modeling while maintaining statistical properties.

4.4 Polynomial Trend Module

Different trends follow distinct trajectories, including phases of growth and decline, highlighting the need to capture diverse trend behaviors. By leveraging Cover's theorem [Cover, 1965], we project smooth time series patterns into a polynomial space, facilitating the extraction of both linear and nonlinear trends. Based on this, polynomial modeling [Liu *et al.*, 2021] is employed to better capture these diverse trend variations. To fully harness polynomial modeling, we introduce a square root transformation to smooth series and enhance the framework's flexibility in capturing diverse trend patterns.

As a deterministic model, PTM consists of two complementary pathways. The first models stable trends with a linear layer applied directly to the historical trend series:

$$T_{\text{origin}}(x^0_{T,t}) = \mathrm{Linear}_{\text{origin}}(x^0_{T,t}), \tag{12}$$

where $\mathrm{Linear}_{\text{origin}}(\cdot)$ captures the general trend.

The second pathway enhances trend modeling by applying square root transformation, thereby mitigating the impact of extreme values and stabilizing trend predictions:

$$T_{\text{sqrt}}(x^0_{T,t}) = \mathrm{Linear}_{\text{root}}(\sqrt{x^0_{T,t}}), \tag{13}$$

where $\mathrm{Linear}_{\text{sqrt}}(\cdot)$ captures the transformed trend.

The outputs from both pathways are combined via weighted summation to capture a wider range of trends:

$$f_\theta(x_{T,t}^0) = \lambda_1 T_{\text{origin}}(x_{T,t}^0) + \lambda_2 T_{\text{sqrt}}(x_{T,t}^0), \tag{14}$$

where λ_1 and λ_2 are learnable parameters that balance the contributions of the linear and root-based components. This mechanism allows the model to adapt to both linear and sublinear growth patterns, enhancing its ability to forecast complex time series with diverse dynamics.

4.5 Optimization

Integrating trend and seasonal components improves forecasting accuracy by combining their strengths for more comprehensive predictions. Finally, the derived loss function enables joint optimization, allowing both components to be learned concurrently and effectively.

Component Reconstruction. The final prediction $\hat{x}_T^0$ is calculated by integrating the trend $\hat{x}_{T,t}^0$ and seasonal components $\hat{x}_{T,s}^0$ as follows:

$$\hat{x}_T^0 = X_\theta\left(\sqrt{\bar{\alpha}_t}x_{s,t}^0 + \sqrt{1-\bar{\alpha}_t}\epsilon, t \mid \nabla^k_{\mathbf{x}_{T,s}}\right) + f_\theta(x_{T,t}^0), \tag{15}$$

where $X_\theta(\cdot)$ represents the CDSM and $f_\theta(\cdot)$ corresponds to the PTM, respectively.

Improved ELBO Optimization. The final prediction $\hat{x}_T^0$ is subsequently denormalized to $\hat{x}_T$ using the mean and variance derived from the historical window, ensuring that the forecast is consistent with the original distribution of the time series. The loss function in CDPM is derived from the Evidence Lower Bound (ELBO). Furthermore, we extend the traditional ELBO to accommodate the separate modeling of the trend and seasonal components. This decoupled modeling of the trend and seasonal components ensures that both contribute independently to the final forecast. Through derivation, we find that by modeling the seasonal component with CDSM and the trend component with PTM, the results can be jointly optimized and trained to improve the overall performance of our framework, leading to more accurate predictions. The improved ELBO is given by:

$$L_{\text{IELBO}} = \mathbb{E}_{x_T^0,\epsilon}\left\|x_T^0 - \hat{x}_T^0\right\|^2, \tag{16}$$

where x_T^0 and $\hat{x}_T^0$ represent the true and predicted values of the time series, respectively. This formulation combines the strengths of both probabilistic diffusion models and deterministic linear models, leading to more accurate forecasts.

Dataset	ETTh1	ETTm1	Wind	Exchange	Weather	Electricity
Dimension	7	7	7	8	21	321
Frequency	1 hour	15 mins	15 mins	1 day	10 mins	1 hour
T (Steps)	168	192	192	14	672	168

Table 1: Overview of the dataset characteristics.

5 Experiments

5.1 Experimental Settings

Datasets. We evaluate the effectiveness of our proposed framework on multiple real-world time series datasets from various domains. The datasets include ETTh and ETTm [Zhou *et al.*, 2021], which represent electricity transformer temperature data; Exchange, which contains exchange rates from eight countries; and Weather[1], consisting of 21 meteorological indicators sampled at 10-minute intervals. Additionally, we use the Electricity dataset[2], which records electricity consumption from 321 clients, and Wind [Li *et al.*, 2022], comprising wind power measurements sampled every 15 minutes from 2020 to 2021.

To ensure consistency in chronological data splitting, we adopt a 6:2:2 split for the ETTh and ETTm datasets, and a 7:1:2 split for the Wind, Weather, Electricity, and Exchange datasets. This setup enables comprehensive testing of the framework's generalization across different temporal segments. Consistent with TimeDiff [Shen and Kwok, 2023], we evaluate performance using historical window lengths of $96, 192, 720, 1440$ to cover varying forecasting horizons.

Baselines. We compare our proposed Conditional Denoising Polynomial Modeling (CDPM) framework against a diverse set of baseline models.

- **Diffusion models:** CSDI [Tashiro *et al.*, 2021], Diffusion-TS [Yuan and Qiao, 2024].
- **Transformer-based models:** iTransformer [Liu *et al.*, 2023], PatchTST [Nie *et al.*, 2023].
- **Linear-based models:** DLinear [Zeng *et al.*, 2023], TSMixer [Ekambaram *et al.*, 2023].
- **Hybrid model:** PatchMixer [Gong *et al.*, 2023], D3VAE [Li *et al.*, 2022].

Implementation Details. Our proposed CDPM is implemented using PyTorch and trained with Exponential Learning Rate Scheduler [Paszke *et al.*, 2019], starting with an initial learning rate of 1×10^{-3} and a batch size of 16. Early stopping with a patience of 10 epochs is employed to prevent overfitting. The diffusion process utilizes $K = 50$ steps with a cosine variance schedule [Rasul *et al.*, 2021] from $\beta_1 = 10^{-4}$ to $\beta_K = 0.5$. We adopt the DDIM sampler [Song *et al.*, 2021]. All experiments are conducted on a workstation equipped with an NVIDIA GeForce RTX 4090 GPU.

5.2 Experiment Results

Main Results. The experimental results in Table 2 validate the effectiveness of the CDPM for time series forecasting. CDPM's decoupling strategy, which separates trend and seasonal components, allows for more accurate modeling of temporal patterns. It performs well on challenging datasets like ETTh1 and ETTh2, where traditional models such as

[1] https://www.bgc-jena.mpg.de/wetter/

[2] https://archive.ics.uci.edu/ml/datasets/ElectricityLoadDiagrams 20112014

Dataset	T	CDPM		iTransformer		Patchmixer		TSMixer		PatchTST		DLinear		Diffusion-TS		D3VAE		CSDI	
		MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE
ETTh1	168	**0.4364**	**0.4363**	0.4433	0.4429	0.4610	0.4560	0.4418	0.4401	0.4401	0.4463	0.4372	0.4420	1.6559	1.0400	0.9523	0.8618	1.1090	0.8009
ETTh2	168	**0.3393**	**0.3826**	0.4009	0.4156	0.3731	0.4055	0.3654	0.3981	0.3805	0.4036	0.3976	0.4233	3.2225	1.4736	5.3165	1.9796	2.0700	1.0748
ETTm1	192	**0.3532**	**0.3781**	0.3685	0.3916	0.3731	0.3983	0.3673	0.3924	0.3626	0.3890	0.3706	0.3968	1.6769	1.0326	0.6341	0.6833	1.0770	0.7788
ETTm2	192	**0.2303**	**0.3009**	0.2535	0.3184	0.2643	0.3226	0.2495	0.3108	0.2472	0.3129	0.2498	0.3224	2.9380	1.3732	4.1001	1.5884	1.6051	0.9397
Electricity	168	0.1606	0.2526	**0.1531**	**0.2461**	0.1559	0.2505	0.1563	0.2497	0.1537	0.2494	0.1628	0.2581	1.4888	0.9935	1.7061	0.9273	——	——
Exchange	14	**0.0204**	0.0979	0.0719	0.1896	0.0430	0.1150	0.0209	**0.0955**	0.0293	0.1178	0.0412	0.1320	2.7192	1.3573	4.8311	1.9942	1.3023	0.8719
Weather	672	**0.3292**	**0.3373**	0.3676	0.3709	0.3295	0.3411	0.3316	0.3430	0.3299	0.3400	0.3398	0.3801	——	——	——	——	0.4842	0.4498
Wind	192	1.1262	0.7534	**1.0653**	**0.7227**	1.1497	0.7582	1.2407	0.7845	1.1411	0.7607	1.0826	0.7378	2.0706	1.1185	3.6837	1.4195	1.4913	0.8942

Table 2: The forecasting results on eight datasets, with the best results highlighted in bold and the second-best underlined. CSDI encounters out-of-memory issues on the Electricity and Diffusion-TS datasets, while D3VAE runs out of memory on the Weather dataset.

Diffusion-TS and D3VAE struggle with varying time resolutions and high noise. CDPM's strength lies in its Polynomial Trend Module (PTM) and Conditional Denoising Seasonal Module (CDSM), which capture complex trends and seasonal dynamics. Unlike CSDI and Diffusion-TS, which struggle with long forecast horizons, CDPM handles long-term dependencies effectively. On volatile datasets like Wind, CDPM captures sharp fluctuations and achieves competitive results, though further refinement could improve performance on highly noisy data. Overall, CDPM shows strong generalization across various datasets.

Method	ETTh2		ETTm2	
	MSE	MAE	MSE	MAE
CDPM	**0.3393**	**0.3826**	**0.2303**	**0.3009**
CDPM[1]	0.3404	0.3781	0.2320	0.3023
CDPM[2]	0.3433	0.3826	0.2450	0.3121
CDPM[3]	0.3853	0.4215	0.2459	0.3272
CDPM[4]	0.3442	0.3807	0.2335	0.3015

Table 3: Ablation study results on ETTh2 and ETTm2.

Ablation Study. The ablation experiments highlight the importance of each component in the CDPM framework, especially for handling complex temporal dynamics in datasets with varying granularities like ETTh2 and ETTm2. We present the results of ablation experiments in Table 3 to evaluate the contributions of individual components within the CDPM framework.

- **CDPM[1]**: Removes the conditional information.
- **CDPM[2]**: Replaces the PTM with a simple linear layer.
- **CDPM[3]**: Removes instance normalization alignment.
- **CDPM[4]**: Removes the conditional information and replaces PTM with a linear layer.

Removing the condition information injection module reduces performance, with a larger impact on ETTm2, where external factors influence intricate temporal patterns. Replacing the PTM with a linear layer causes a significant performance drop, which is crucial for capturing long-term trends and periodic behaviors. Eliminating instance normalization leads to the largest performance decline, highlighting its role in stabilizing training for high-resolution datasets. Removing both the condition information injection module and PTM further degrades performance, emphasizing the necessity of both components.

Method	ETTh2		ETTm2	
	MSE	MAE	MSE	MAE
CDPM	**0.3393**	**0.3826**	**0.2303**	**0.3009**
CDPM[5]	0.3839	0.4106	0.2518	0.3157
CDPM[6]	0.3541	0.3832	0.2452	0.3122

Table 4: Decoupling validation study results on ETTh2 and ETTm2.

Evaluation of the Decoupling Strategy. The Decoupling Validation Study emphasizes the importance of using appropriate models to represent the distinct components of the time series, while also providing evidence for the effectiveness and validity of the decoupling strategy. We present the results of ablation experiments in Table 4 to evaluate the validation of our decoupling framework.

- **CDPM[5]**: This variant directly applies the CDSM to the original time series.
- **CDPM[6]**: In this variant, PTM and CDSM are reversed, with PTM predicting the seasonal component and CDSM predicting the trend component.

The decoupling strategy that separates trend and seasonal patterns markedly improves performance. Applying our framework directly underperforms due to its inability to separate trend and seasonal components, while reversing PTM and CDSM shows slight improvements but still falls short of our framework, highlighting the importance of proper module assignment. These results emphasize the need to tailor modules for distinct patterns, with decoupling especially beneficial for datasets with clear temporal structure.

PTM and CDSM Modeling Analysis. To evaluate the effectiveness of PTM and CDSM in modeling different components of the ETTh1 dataset, we conducted experiments with an input length of 192 and a prediction horizon of 168. The results highlight the effectiveness of tailored modeling strategies for each component. As shown in Figure 4 (a), PTM excels in capturing intrinsic and gradual trends due to its linear structure, accurately modeling patterns and tracking the overall direction of the time series. In contrast, applying CDSM to the trend component introduces unnecessary noise and overfitting to minor variations, diminishing trend quality.

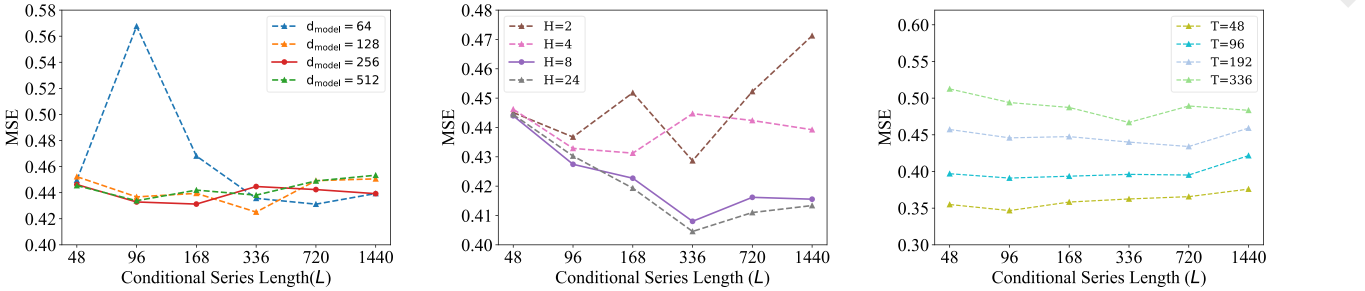

Figure 3: Parameter sensitivity analysis of CDPM: hidden dimension (left), patch length (middle), and predicted length (right).

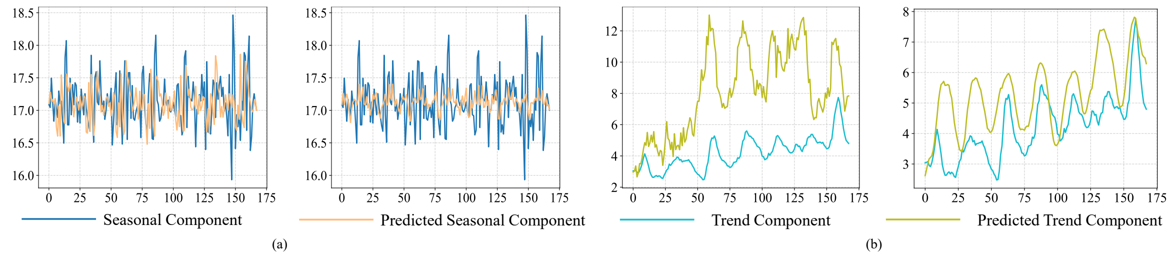

Figure 4: (a) Visual comparison of CDSM (left) and PTM (right) in modeling seasonal components. (b) Visual comparison of CDSM (left) and PTM (right) in modeling trend components.

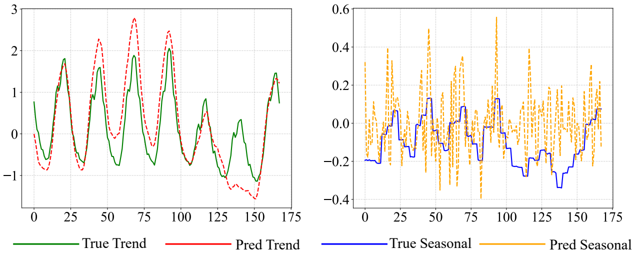

Figure 5: Visualization of trend (left) and seasonall (right) components of joint modeling for the electricity dataset.

For the seasonal component, which exhibits complex fluctuations, the seasonal module demonstrates superior performance, as shown in Figure 4 (b). These results validate each module's modeling capabilities for its designated component.

Hyper-Parameter Sensitivity Analysis. To investigate the impact of key design choices in CDPM, such as hidden dimension size (d_{model}), patch length (H), and prediction horizon on model performance, we conducted a hyperparameter sensitivity analysis using the ETTh1 dataset. The results show that increasing d_{model} initially improves performance, peaking at $d_{\text{model}} = 256$, while larger values (e.g., $d_{\text{model}} = 512$) offer diminishing returns and risk overfitting, highlighting the trade-off between model capacity and efficiency. Similarly, H significantly affects performance, with $H = 8$ capturing both local dependencies and global patterns, whereas larger values introduce redundancy and reduce efficiency. Finally, our framework performs consistently across prediction horizons.

Combined Modeling Capability Analysis. To assess the interpretability and predictive performance of the framework, we conducted a joint analysis of the PTM and CDSM modules on the Electricity dataset, as shown in Figure 5. The framework decouples multidimensional time series, capturing diverse trends and cyclic patterns, demonstrating its capacity to model complex temporal dynamics. In Figure 5 (left), applied to the Electricity dataset with clear periodic and trend structures, the trend module captures the underlying patterns, with predictions closely matching actual values. Figure 5 (right) illustrates the seasonal module's ability to model periodic variations, where predicted components align well with the observed sequences, compensating for deviations in trend estimation and improving accuracy. The integration of deterministic and diffusion-based modules within the framework enhances both interpretability and predictive accuracy across datasets with combined components.

6 Conclusion

In this paper, we proposed the Conditional Denoising Polynomial Modeling (CDPM), a novel time series forecasting framework that separately models trend and seasonal components using specialized modules, trained in an end-to-end manner. The Polynomial Trend Module (PTM) captured intrinsic trends, while the Conditional Denoising Seasonal Module (CDSM) addressed complex seasonal fluctuations. By reformulating the Evidence Lower Bound, we were able to better integrate the training of both modules. This decoupling paradigm enhanced the modeling of temporal dynamics, resulting in improved forecasting accuracy. Extensive experiments on real-world datasets validated the effectiveness of the proposed CDPM.

Acknowledgements

This research was supported by grants from the grants of Provincial Natural Science Foundation of Anhui Province (No.2408085QF193), the National Natural Science Foundation of China (62337001), the Key Technologies R&D Program of Anhui Province (No. 202423k09020039), USTC Research Funds of the Double First-Class Initiative (No. YD2150002501), and the Fundamental Research Funds for the Central Universities (No. WK2150110032).

References

[Asteriou and Hall, 2011] Dimitros Asteriou and Stephen G Hall. Arima models and the box–jenkins methodology. *Applied Econometrics*, 2(2):265–286, 2011.

[Bai *et al.*, 2018] 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.

[Bilos *et al.*, 2023] Marin Bilos, Kashif Rasul, Anderson Schneider, Yuriy Nevmyvaka, and Stephan Günnemann. Modeling temporal data as continuous functions with stochastic process diffusion. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202, pages 2452–2470. PMLR, 2023.

[Cheng *et al.*, 2024] Mingyue Cheng, Jiqian Yang, Tingyue Pan, Qi Liu, and Zhi Li. Convtimenet: A deep hierarchical fully convolutional model for multivariate time series analysis. *arXiv preprint arXiv:2403.01493*, 2024.

[Cheng *et al.*, 2025a] Mingyue Cheng, Yiheng Chen, Qi Liu, Zhiding Liu, Yucong Luo, and Enhong Chen. Instructime: Advancing time series classification with multimodal language modeling. In *Proceedings of the Eighteenth ACM International Conference on Web Search and Data Mining*, pages 792–800, 2025.

[Cheng *et al.*, 2025b] Mingyue Cheng, Zhiding Liu, Xiaoyu Tao, Qi Liu, Jintao Zhang, Tingyue Pan, Shilong Zhang, Panjing He, Xiaohan Zhang, Daoyu Wang, et al. A comprehensive survey of time series forecasting: Concepts, challenges, and future directions. *Authorea Preprints*, 2025.

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

[Cover, 1965] Thomas M Cover. Geometrical and statistical properties of systems of linear inequalities with applications in pattern recognition. *IEEE transactions on electronic computers*, (3):326–334, 1965.

[Ekambaram *et al.*, 2023] Vijay Ekambaram, Arindam Jati, Nam Nguyen, Phanwadee Sinthong, and Jayant Kalagnanam. Tsmixer: Lightweight mlp-mixer model for multivariate time series forecasting. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 459–469, 2023.

[Feng *et al.*, 2024] Shibo Feng, Chunyan Miao, Zhong Zhang, and Peilin Zhao. Latent diffusion transformer for probabilistic time series forecasting. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 11979–11987, 2024.

[Gong *et al.*, 2023] Zeying Gong, Yujin Tang, and Junwei Liang. Patchmixer: A patch-mixing architecture for long-term time series forecasting. *arXiv preprint arXiv:2310.00655*, 2023.

[Goodfellow *et al.*, 2014] Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks, 2014.

[Gruver *et al.*, 2024] Nate Gruver, Marc Finzi, Shikai Qiu, and Andrew G Wilson. Large language models are zero-shot time series forecasters. *Advances in Neural Information Processing Systems*, 36, 2024.

[Jin *et al.*, 2023] Ming Jin, Shiyu Wang, Lintao Ma, Zhixuan Chu, James Y Zhang, Xiaoming Shi, Pin-Yu Chen, Yuxuan Liang, Yuan-Fang Li, Shirui Pan, et al. Time-llm: Time series forecasting by reprogramming large language models. *arXiv preprint arXiv:2310.01728*, 2023.

[Kashpruk *et al.*, 2023] Nataliia Kashpruk, Cezary Piskor-Ignatowicz, and Jerzy Baranowski. Time series prediction in industry 4.0: A comprehensive review and prospects for future advancements. *Applied Sciences*, 13(22):12374, 2023.

[Kilcioglu *et al.*, 2017] Cinar Kilcioglu, Justin M Rao, Aadharsh Kannan, and R Preston McAfee. Usage patterns and the economics of the public cloud. In *Proceedings of the 26th International Conference on World Wide Web*, pages 83–91, 2017.

[Kim *et al.*, 2021] 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.

[Kingma and Welling, 2022] Diederik P Kingma and Max Welling. Auto-encoding variational bayes, 2022.

[Kollovieh *et al.*, 2024] Marcel Kollovieh, Abdul Fatir Ansari, Michael Bohlke-Schneider, Jasper Zschiegner, Hao Wang, and Yuyang Bernie Wang. Predict, refine, synthesize: Self-guiding diffusion models for probabilistic time series forecasting. *Advances in Neural Information Processing Systems*, 36, 2024.

[Koprinska *et al.*, 2018] Irena Koprinska, Dengsong Wu, and Zheng Wang. Convolutional neural networks for energy time series forecasting. In *2018 international joint conference on neural networks (IJCNN)*, pages 1–8. IEEE, 2018.

[Lai *et al.*, 2018] Guokun Lai, Wei-Cheng Chang, Yiming Yang, and Hanxiao Liu. Modeling long-and short-term temporal patterns with deep neural networks. In *The 41st international ACM SIGIR conference on research & development in information retrieval*, pages 95–104, 2018.

[Lee *et al.*, 2024] Seunghan Lee, Kibok Lee, and Taeyoung Park. Ant: Adaptive noise schedule for time series diffusion models. *Advances in Neural Information Processing Systems*, 37:121099–121130, 2024.

[Li *et al.*, 2022] Yan Li, Xinjiang Lu, Yaqing Wang, and Dejing Dou. Generative time series forecasting with diffusion, denoise, and disentanglement. *Advances in Neural Information Processing Systems*, 35:23009–23022, 2022.

[Lin *et al.*, 2023] Shengsheng Lin, Weiwei Lin, Wentai Wu, Feiyu Zhao, Ruichao Mo, and Haotong Zhang. Segrnn: Segment recurrent neural network for long-term time series forecasting. *arXiv preprint arXiv:2308.11200*, 2023.

[Liu *et al.*, 2021] Zhenyu Liu, Zhengtong Zhu, Jing Gao, and Cheng Xu. Forecast methods for time series data: A survey. *IEEE Access*, 9:91896–91912, 2021.

[Liu *et al.*, 2022] 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.

[Liu *et al.*, 2023] Yong Liu, Tengge Hu, Haoran Zhang, Haixu Wu, Shiyu Wang, Lintao Ma, and Mingsheng Long. itransformer: Inverted transformers are effective for time series forecasting. *arXiv preprint arXiv:2310.06625*, 2023.

[Liu *et al.*, 2024] Zhiding Liu, Jiqian Yang, Mingyue Cheng, Yucong Luo, and Zhi Li. Generative pretrained hierarchical transformer for time series forecasting. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 2003–2013, 2024.

[Masini *et al.*, 2023] Ricardo P Masini, Marcelo C Medeiros, and Eduardo F Mendes. Machine learning advances for time series forecasting. *Journal of economic surveys*, 37(1):76–111, 2023.

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

[Paszke *et al.*, 2019] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32, 2019.

[Rasul *et al.*, 2021] Kashif Rasul, Calvin Seward, Ingmar Schuster, and Roland Vollgraf. Autoregressive denoising diffusion models for multivariate probabilistic time series forecasting. In *International Conference on Machine Learning*, pages 8857–8868. PMLR, 2021.

[Rezende and Mohamed, 2016] Danilo Jimenez Rezende and Shakir Mohamed. Variational inference with normalizing flows, 2016.

[Rombach *et al.*, 2022] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10684–10695, 2022.

[Shen and Kwok, 2023] Lifeng Shen and James T. Kwok. Non-autoregressive conditional diffusion models for time series prediction. In *Proceedings of the 40th International Conference on Machine Learning (ICML)*, pages 31016–31029, 2023.

[Song *et al.*, 2021] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. In *Proceedings of the 9th International Conference on Learning Representations*, 2021.

[Tashiro *et al.*, 2021] Yusuke Tashiro, Jiaming Song, Yang Song, and Stefano Ermon. Csdi: Conditional score-based diffusion models for probabilistic time series imputation. *Advances in Neural Information Processing Systems*, 34:24804–24816, 2021.

[Wu *et al.*, 2020] 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*, pages 753–763, 2020.

[Wu *et al.*, 2021] 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.

[Yan *et al.*, 2021] Tijin Yan, Hongwei Zhang, Tong Zhou, Yufeng Zhan, and Yuanqing Xia. Scoregrad: Multivariate probabilistic time series forecasting with continuous energy-based generative models. *arXiv preprint arXiv:2106.10121*, 2021.

[Yi *et al.*, 2024] Kun Yi, Qi Zhang, Wei Fan, Hui He, Liang Hu, Pengyang Wang, Ning An, Longbing Cao, and Zhendong Niu. Fouriergnn: Rethinking multivariate time series forecasting from a pure graph perspective. *Advances in Neural Information Processing Systems*, 36, 2024.

[Yuan and Qiao, 2024] Xinyu Yuan and Yan Qiao. Diffusion-ts: Interpretable diffusion for general time series generation. In *Proceedings of the Twelfth International Conference on Learning Representations*, 2024.

[Zeng *et al.*, 2023] Ailing Zeng, Muxi Chen, Lei Zhang, and Qiang Xu. Are transformers effective for time series forecasting? In *Proceedings of the AAAI conference on artificial intelligence*, volume 37, pages 11121–11128, 2023.

[Zhou *et al.*, 2021] 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. In *Proceedings of the AAAI conference on artificial intelligence*, volume 35, pages 11106–11115, 2021.