

Research paper

An efficient framework for general long-horizon time series forecasting with Mamba and Diffusion Probabilistic Models

Wenjing Wang^a, Qilei Li^{b,c}, Ziwu Jiang^a, Deqian Fu^a^{*}, David Camacho^d^a School of Information Science and Engineering, Linyi University, Linyi, 276000, China^b Faculty of Artificial Intelligence in Education, Central China Normal University, Wuhan, 430079, China^c National Engineering Research Center for Educational Big Data, Central China Normal University, Wuhan, 430079, China^d Computer Science Department, Universidad Politécnica de Madrid, 28040, Spain

ARTICLE INFO

Keywords:

Time series data analysis

State space models

Denoising diffusion probabilistic models

Mamba

ABSTRACT

Time series forecasting plays an essential role in supporting critical decision-making processes in risk management and resource allocation in various fields, including finance, transportation, industrial systems, etc. Conventional models can effectively capture volatility and, are proficient in handling specific patterns, such as the AutoRegressive Integrated Moving Average model (ARIMA) and the Generalized AutoRegressive Conditional Heteroskedasticity model (GARCH). Nonetheless, these models meet many challenges, such as high dimensionality, non-stationarity, and nonlinearity inherent in real-world data. Although deep learning methodologies can provide better performance, they may still suffer from long-term errors and heightened computational expenses. A novel framework named Mamba Diffusion Probabilistic Models (MambaDiffTS) is proposed, which integrates Mamba's state space model with a frequency-aware diffusion process grounded in Denoising Diffusion Probabilistic Models (DDPM). Mamba's selective state transitions enable linear-time modeling of long-range dependencies; at the same time, frequency-aware spectral decomposition isolates trends and seasonality through Fourier regularization. Furthermore, the implementation of spectral energy-guided noise scheduling preserves temporal fidelity. Extensive experiments on diverse benchmarks-financial volatility, industrial IoT sensor data, and climate modeling-demonstrate MambaDiffTS's superiority. Notably, on stock forecasting tasks, MambaDiffTS reduces Mean Squared Error (MSE) by approximately 18.6% compared to the best-performing baseline, and substantially outperforms diffusion models, all while maintaining linear computational complexity. The proposed MambaDiffTS facilitates scalable forecasting over extended horizons.

1. Introduction

Time series analysis has been a linchpin in dynamic system modeling, as a cornerstone for understanding and predicting various phenomena. Traffic flow prediction, as a typical long-term time series problem, poses unique challenges. On the one hand, it involves multi-scale patterns ranging from short-term peak hours to long-term seasonal trends. On the other hand, the sequence is often affected by abrupt non-stationary fluctuations caused by external factors such as accidents or weather conditions. Furthermore, large-scale urban networks require real-time inference, which further increases the difficulty of modeling. Traditional statistical models and even deep learning architectures usually fail to capture these characteristics simultaneously, either due to error accumulation in long sequences or the excessive

quadratic computational cost of the attention mechanism. In the initial stages, classic methodologies such as AutoRegressive Integrated Moving Average (ARIMA) (Box and Jenkins, 2015) and Generalized AutoRegressive Conditional Heteroskedasticity Model (GARCH) (Bollerslev, 1986) were the workhorses in econometrics and signal processing. ARIMA's prowess in handling linear trends and seasonality, along with GARCH's ability to capture volatility clustering, made them invaluable in these domains. As the complexity of modern datasets exploded, these traditional approaches nonetheless encountered significant limitations. Real-world data often exhibit intricate nonlinear interactions and high-dimensional patterns, which extend beyond the capabilities of ARIMA and GARCH models. For example, in complex financial markets where multiple factors interact nonlinearly, or in environmental monitoring

^{*} Corresponding author.E-mail address: fudeqian@lyu.edu.cn (D. Fu).<https://doi.org/10.1016/j.engappai.2025.112525>

Received 26 June 2025; Received in revised form 4 September 2025; Accepted 24 September 2025

0952-1976/© 2025 Elsevier Ltd. All rights are reserved, including those for text and data mining, AI training, and similar technologies.

systems with a large number of sensors, these classic models struggle to provide accurate and comprehensive descriptions (Shumway and Stoffer, 2017; Bauwens et al., 2006).

The advent of deep learning represented a paradigm shift in the domain of time series analysis. Recurrent Neural Networks (RNNs) (Sherstinsky, 2020) and their derivatives, like Long Short-Term Memory (LSTM) (Graves, 2012) and gated RNNs (Chung et al., 2015), bring new hope. By incorporating memory cells, they enabled the modeling of sequential dependencies, which was a significant leap forward. LSTM-based architectures achieved remarkable success in tasks such as traffic flow prediction (Tian and Pan, 2015) and anomaly detection (Lu et al., 2023). The autoregressive nature of these models presents an inherent limitation. In long sequences, errors have a propensity to accumulate, significantly limiting their scalability. In applications like long-term stock price prediction, where the sequence length can be substantial, the error propagation in LSTM-based models can lead to inaccurate forecasts (Rather, 2021).

Concurrently, Convolutional Neural Networks (CNNs) emerged as an alternative for time series forecasting. Leveraging hierarchical feature extraction through dilated convolutions (Chen, 2015; Borovykh et al., 2017), CNNs can capture local patterns in time series data. Nevertheless, they lack the ability to adaptively weigh temporal dependencies across different scales. For instance, in a time series of power consumption data, where short-term spikes and long-term trends coexist, CNNs may not be able to effectively distinguish and model these different temporal characteristics.

The advent of Transformer architectures (Vaswani et al., 2017) was another game-changer. By leveraging self-attention mechanisms, they could better handle long-range dependencies in time series. This led to the development of specialized variants such as Informer (Zhou et al., 2021) for long-term forecasting and Autoformer (Wu et al., 2021) for seasonal-trend decomposition. Despite their achievements, Transformers suffer from quadratic computational complexity. This complexity becomes a major bottleneck, especially when dealing with large-scale time series data. To address this issue, innovations like iTransformer (Liu et al., 2023) and Fredformer (Piao et al., 2024) have been proposed. Meanwhile, pretrained language models like Bidirectional Encoder Representation from Transformers (BERT) (Devlin et al., 2019) inspired temporal adaptations such as TimeBERT (Wang et al., 2022). These models are constrained by their rigid tokenization strategies when applied to continuous time series. In continuous sensor data, the fixed tokenization may not be able to capture the fine-grained information effectively (Wen et al., 2023). Beyond early variants such as Informer (Zhou et al., 2021) and Autoformer (Wu et al., 2021), more specialized models have emerged. For example, Self-supervised Learning Spatio-temporal Entanglement Transformer (SSL-STMFormer) (Li et al., 2025) introduces a self-supervised framework to disentangle spatio-temporal dependencies, enhancing robustness under limited labeled data. Learnable Long-range Graph Transformer (Llgformer) (Jin et al., 2025) leverages learnable long-range graph structures to capture non-local correlations in large-scale urban networks. Meanwhile, Stlformer: Spatio-temporal Long-term Embedding transformer (STLTE-former) (Chauhan et al., 2025) focuses on long-term spatio-temporal embedding, improving the modeling of seasonal and trend components in traffic series. These advances highlight the continuous efforts to adapt Transformer architectures for real-world traffic scenarios; however, their reliance on quadratic attention mechanisms still poses scalability challenges, especially in large-scale deployment.

A new paradigm emerges with SSMs, particularly Structured State Space Sequence Model (S4) (Gu et al., 2021) and its successor Mamba (Gu and Dao, 2023). These models combine linear-time complexity with selective state transitions, enabling efficient modeling of long-range dependencies. Mamba's input-dependent gating mechanism and hardware-aware design (Ahamed and Cheng, 2024) have shown remarkable performance in tasks like electricity demand forecasting, outperforming Transformer-based counterparts (Zhou et al., 2021; Wu

et al., 2021). In parallel, diffusion models (Ho et al., 2020) have significantly expanded generative capabilities. Frameworks such as Autoregressive Diffusion (Rasul et al., 2021) have adapted to the denoising process of time series. While isotropic noise injection is a common component in diffusion approaches, it often degrades the temporal fidelity of existing methods—and even erases critical multi-scale patterns in the time series. For example, in Multi-Granularity Time Series Diffusion (MG-TSD) (Fan et al., 2024a), which introduces multi-granularity guidance, the reliance on computationally intensive transformer backbones limits its practical application in real-world scenarios.

To fill in these lacunae and conquer the constraints of existing methodologies, we propose a novel integration of Mamba's selective state space mechanisms with diffusion-based generation. In contrast to prior autoregressive diffusion models (Rasul et al., 2021) or transformer-centric approaches (Huang et al., 2024), our framework capitalizes on Mamba's linear-time processing ability to maintain long-term dependencies. We introduce a frequency-aware diffusion, which constitutes a pivotal innovation. This process carefully aligns noise scheduling with seasonal and trend components in the time series through Fourier-regularized losses (Wu et al., 2021). As a result, it effectively mitigates the loss of high-frequency oscillations that occur in traditional diffusion steps (Fan et al., 2024a). Additionally, Mamba's structured state dynamics offer interpretable intermediate representations. This is a crucial advantage over the "black-box" transformer-based models (Wang et al., 2022; Wen et al., 2023), especially in sensitive domains such as healthcare and finance, where interpretability is essential for decision-making.

We conducted comprehensive experiments on benchmarks covering financial volatility (Bollerslev, 1986), industrial IoT sensor data (Tian and Pan, 2015), and climate modeling (Shumway and Stoffer, 2017). The results unequivocally demonstrate that the proposed approach surpasses classical methodologies in both the fidelity of generated outputs and computational efficiency. By integrating Mamba's efficient computation with diffusion models' generative capacity, we establish a scalable framework for interpretable time-series synthesis. This hybrid architecture addresses critical challenges in modeling high-dimensional temporal patterns and long-range dependencies across scientific domains. In summary, the major contributions in this paper are presented as follows:

1. **A Unified State-Space Diffusion Framework for Time-Series Synthesis:** We proposed the MambaDiffTS framework, which attempts to integrate Mamba's selective state-space mechanisms with a diffusion process. It is intended to achieve some improvement in efficiency and fidelity when modeling long-horizon temporal dependencies. The framework draws on Mamba's linear computational complexity $O(Td)$ and input-dependent gating to dynamically focus on critical temporal patterns, aiming to enable scalable synthesis of complex sequences spanning thousands of time steps.
2. **Frequency-Aware Spectral Decomposition and Noise Scheduling:** This study introduces a novel framework that integrates Fourier-based decomposition with spectral energy-guided diffusion to explicitly decouple low-frequency trends from high-frequency seasonality components. By adaptively adjusting noise injection based on frequency energy, the framework preserves multi-scale patterns (e.g., hourly traffic oscillations and annual economic cycles) and ensures realistic synthesis of complex temporal structures.
3. **Scalable Multi-Domain Generalization:** The framework demonstrates robust performance across heterogeneous domains—from high-frequency financial volatility to sparse industrial sensor data—validating its ability to handle diverse temporal dynamics without domain-specific tuning. This universality stems from Mamba's state-space flexibility and the diffusion process's adaptability to spectral characteristics.

The remainder of this study is structured as follows. Section 2 reviews related work on traditional models, SSMs (e.g., Mamba), and diffusion models. Section 3 introduces Denoise Diffusion Probabilistic Models (DDPMs) and Mamba as preliminaries. Section 4 details the MambaDiffTS framework. Section 5 presents and analyzes experimental results. Finally, Section 6 concludes the study and outlines future work.

2. Related work

Numerous empirical studies have leveraged deep neural networks for traffic flow prediction. LSTM-based models (Tian and Pan, 2015) can capture short-term sequential dependencies but are prone to error accumulation in long sequences. CNN-based methods (Borovykh et al., 2017) improve local feature extraction yet fail to adapt to heterogeneous temporal scales. Transformer-based models (e.g., Informer (Zhou et al., 2021)) are capable of handling long-range dependencies, but they are constrained by quadratic complexity, which limits their real-time applicability in traffic systems. Recently, diffusion-based models (e.g., TimeGrad (Rasul et al., 2021)) have been applied; however, isotropic noise injection often over-smooths high-frequency traffic dynamics, leading to reduced fidelity during traffic peak hours. These limitations underscore the need for a framework that is both computationally efficient and capable of preserving the multi-scale structures of traffic flow sequences.

2.1. Traditional approaches for time series analysis

Time series data appear across diverse domains such as finance, healthcare, and industrial systems, characterized by complex temporal dependencies, seasonality, and non-stationarity. Traditional statistical models like ARIMA (Box and Jenkins, 2015) and GARCH (Bollerslev, 1986) have been widely used for univariate forecasting but struggle with non-linearity and high-dimensional dependencies. For multivariate settings, Vector Autoregressive Model (VAR) (Lütkepohl, 2005) provides an extension but suffers from scalability issues.

Deep learning has significantly advanced time series modeling by introducing architectures that capture complex dependencies. RNNs and their variants (Hochreiter and Schmidhuber, 1997; Cho et al., 2014) enable sequential memory propagation but face limitations in long-term dependency modeling due to vanishing gradients and autoregressive error accumulation. CNN-based approaches (Borovykh et al., 2017) leverage dilated convolutions for parallelized feature extraction but lack adaptability to dynamically changing patterns. Transformers (Vaswani et al., 2017) introduce self-attention for capturing long-range dependencies, with adaptations such as Informer (Zhou et al., 2021) and Autoformer (Wu et al., 2021) improving efficiency. However, their quadratic complexity remains a bottleneck, especially for long-horizon forecasting.

For generative modeling, Generative Adversarial Networks (GANs) (Goodfellow et al., 2014) and Variational Autoencoders (VAEs) (Kingma and Welling, 2014) have been applied to time series synthesis but face challenges in mode collapse and blurry reconstructions. Recently, diffusion models (Ho et al., 2020) have emerged as a powerful alternative due to their stability and sample quality. However, applying them to time series remains non-trivial due to their reliance on Transformer-based architectures, which introduce significant computational overhead.

2.2. State space models and mamba

SSMs represent a significant advancement in time series analysis, addressing limitations of traditional autoregressive frameworks and modern deep learning architectures. While classical methods like ARIMA and GARCH excel in modeling linear trends and volatility (Box and Jenkins, 2015; Bollerslev, 1986), they falter in capturing nonlinear interactions and high-dimensional dependencies (Lütkepohl, 2005).

Similarly, deep learning approaches such as RNNs and Transformers struggle with error accumulation in long sequences or quadratic computational costs (Vaswani et al., 2017; Zhou et al., 2021), as highlighted in Section 4.

SSMs circumvent these challenges by maintaining structured latent states that evolve over time, enabling efficient modeling of long-range dependencies without explicit recurrence or attention mechanisms (Gu et al., 2021). Early SSM variants, such as the S4, demonstrate potential in tasks like electricity demand forecasting but face practical limitations due to computational overhead and specialized initialization requirements (Gu et al., 2021). The introduction of Mamba (Gu and Dao, 2023) revolutionized this paradigm through two key innovations:

1. Selective State Transitions: Mamba dynamically adjusts information flow via input-dependent gating, allowing adaptive modeling of non-stationary temporal patterns, which is a critical capability for real-world datasets with shifting dynamics (Gu and Dao, 2023).

2. Hardware-Aware Efficiency: By optimizing memory usage and parallelizing state updates, Mamba achieves linear computational complexity, outperforming Transformers in scalability for long sequences (Ahmed and Cheng, 2024).

Mamba's efficient state-space architecture offers a solution: by replacing Transformer backbones with selective state transitions, it reduces computational costs while preserving temporal fidelity (Ahmed and Cheng, 2024). For instance, TimeMachine (Ahmed and Cheng, 2024) demonstrates that Mamba-based frameworks achieve superior performance in long-term forecasting tasks, maintaining interpretable representations essential for domains like healthcare and finance.

By integrating SSMs with diffusion processes, Mamba bridges the gap between efficiency and generative flexibility, as detailed in Section 4. This synergy addresses longstanding challenges in high-dimensional time series synthesis, offering a scalable alternative to traditional and Transformer-centric approaches.

2.3. Diffusion models for time series generation

Diffusion models have emerged as a powerful generative framework by iteratively mapping noise to structured data through a learned denoising process (Ho et al., 2020). Diffusion models have distinct characteristics compared to other generative models. GANs need adversarial training, while VAEs impose strong latent space constraints. In contrast, diffusion models operate differently. They model the reverse process of a stochastic perturbation. This unique approach ensures stable and diverse sample generation.

For time series applications, TimeGrad (Rasul et al., 2021) introduces a diffusion-based probabilistic forecasting approach, while Conditional Score-based Diffusion Models (CSDI) (Tashiro et al., 2021) extend diffusion models for time series imputation. More recent work, such as Multi-Granularity Time Series Diffusion (MG-TSD) (Fan et al., 2024a), has focused on enhancing diffusion-based generation through multi-scale conditioning. However, most diffusion models employ Transformer-based denoisers, leading to high computational costs and limited scalability.

A core challenge for time series diffusion models lies in maintaining temporal consistency. Conventional isotropic noise injection mechanisms tend to erase fine-grained temporal dependencies, resulting in synthesized sequences with unrealistic fluctuations. Some approaches have explored frequency-aware regularization (Wu et al., 2021) and adaptive noise scheduling (Huang et al., 2024) to mitigate these effects. However, these methods introduce additional complexity. Transformer-based architectures are burdened with quadratic complexity, which causes the computational cost to rise exponentially as the sequence length expands. In scenarios such as long-term stock price forecasting or long-term environmental sensor data analysis, this high computational cost results in slow processing and excessive resource consumption, rendering them unsuitable for long-sequence generation. Recently, diffusion-based approaches have attracted increasing attention in traffic

prediction tasks due to their strong generative capabilities. For instance, Traffic State Generative Diffusion Model (TSGDiff) (Zhang et al., 2025) leverages multi-source information fusion to enhance traffic state generation, enabling robust modeling under heterogeneous data conditions. Similarly, the Interactive Diffusion GCN (Zhang et al., 2024) integrates graph convolution with dynamic diffusion processes to capture spatio-temporal correlations adaptively. In parallel, Random Graph Diffusion Attention Network (RGDAN) (Fan et al., 2024b) introduces a random graph diffusion attention mechanism that effectively balances global structural dependencies with local temporal dynamics. In contrast, SSMs offer a promising alternative for denoising networks. SSMs manage structured latent states that change over time, effectively handling long-range dependencies without relying on explicit recurrence or attention mechanisms. Among them, Mamba stands out with its input-dependent gating mechanism. This mechanism enables Mamba to adaptively regulate information flow, prioritize crucial temporal patterns, and focus on key data features, like peak demand periods in electricity demand forecasting. Mamba's efficient state-space updates operate with linear computational complexity $O(Td)$, allowing it to preserve long-term dependencies while slashing computational overhead compared to Transformer-based models. Consequently, Mamba provides a scalable solution for high-dimensional time series synthesis across diverse domains.

Recent studies have proposed hybrid architectures that combine frequency decomposition with diffusion-based learning (Piao et al., 2024; Huang et al., 2024). The integration of these latest findings further motivates the adoption of frequency-aware noise scheduling in MambaDiffTS.

3. Preliminaries

3.1. DDPMs

DDPMs (Ho et al., 2020) have emerged as a powerful generative modeling framework based on the principle of iterative denoising. The core idea is to transform a data distribution into a Gaussian prior using a forward diffusion process. Subsequently, a neural network, parameterized for this purpose, is employed to learn the reverse transformation via a denoising process.

Forward Diffusion Process. Given a data sample $x_0 \sim q(x)$, the forward diffusion process progressively corrupts it by adding Gaussian noise over T steps:

$$q(x_t|x_{t-1}) = \mathcal{N}(x_t; \sqrt{1 - \beta_t}x_{t-1}, \beta_t I), \quad (1)$$

where β_t is a predefined noise variance schedule. This leads to a closed-form distribution:

$$q(x_t|x_0) = \mathcal{N}(x_t; \sqrt{\bar{\alpha}_t}x_0, (1 - \bar{\alpha}_t)I), \quad (2)$$

where $\bar{\alpha}_t = \prod_{s=1}^t (1 - \beta_s)$ represents the cumulative noise scale.

Reverse Denoising Process. The generative process learns to recover x_0 from a pure Gaussian sample $x_T \sim \mathcal{N}(0, I)$ by approximating the reverse transition:

$$p_\theta(x_{t-1}|x_t) = \mathcal{N}(x_{t-1}; \mu_\theta(x_t, t), \Sigma_\theta(x_t, t)), \quad (3)$$

where $\mu_\theta(x_t, t)$ and $\Sigma_\theta(x_t, t)$ are neural network - predicted mean and variance. In practice, most DDPM implementations fix Σ_θ and train a model $\epsilon_\theta(x_t, t)$ to predict the added noise directly:

$$\epsilon_\theta(x_t, t) \approx \frac{x_t - \sqrt{\bar{\alpha}_t}x_0}{\sqrt{1 - \bar{\alpha}_t}}. \quad (4)$$

This formulation allows efficient optimization using the simplified variational objective:

$$\mathcal{L}_{\text{DDPM}} = \mathbb{E}_{x_0, t, \epsilon} [\|\epsilon - \epsilon_\theta(x_t, t)\|^2], \quad (5)$$

where $\epsilon \sim \mathcal{N}(0, I)$ is a Gaussian noise sample. Due to their stability and high-quality generation, DDPMs surpass traditional generative models such as GANs and VAEs. However, DDPMs often rely on Transformer-based denoisers, leading to high computational costs, particularly for long time series. This prompts the investigation of more efficient architectures, such as state space models.

3.2. Mamba: Selective state space model for efficient sequence modeling

Mamba (Gu and Dao, 2023) is an SSM model designed as an efficient strategy alternative to Transformer-based architectures. Unlike Transformers, which rely on self-attention mechanisms, Mamba leverages structured state-space updates to model long-range dependencies with linear computational complexity.

State Space Model Formulation. A continuous-time state space model is defined by the equations:

$$\frac{d}{dt}x(t) = Ax(t) + Bu(t), \quad y(t) = Cx(t) + Du(t), \quad (6)$$

where $x(t)$ is the hidden state, u_t is the input, y_t is the output, and A , B , C , and D are learnable system matrices. This process enables efficient information propagation over long sequences without the need for explicit recurrence or attention mechanisms.

To discretize this process, we define the discrete update rule:

$$x_{t+1} = Ax_t + Bu_t, \quad y_t = Cx_t, \quad (7)$$

where A , B , and C are trained using structured parameterizations, ensuring stability and efficiency. However, previous SSM-based models, like S4 (Gu et al., 2021), required specialized initialization and high-order recurrence operations, limiting their practical efficiency. To address these limitations and improve the practical efficiency of state-space-based time series modeling, Mamba was developed with innovative features.

Selective Gating Mechanism for Mamba model. Mamba improves upon standard SSMs by introducing an input-dependent gating function, which dynamically controls the state updates:

$$x_{t+1} = \sigma(Wu_t) \odot (Ax_t + Bu_t), \quad (8)$$

where W is a gating matrix and σ is an activation function. u_t denotes the embedded input vector derived from the original multivariate time series at step t , where its dimension m corresponds to the number of pre-processed features. Specifically, in the SSM, $y_t = Cx_t$ denotes the linear mapping from the state to the output; while in the diffusion framework, x_t represents the noise sample at the t th step of the diffusion process. These two variables are connected via $\epsilon_\theta(x_t, t) = f_{\text{Mamba}}(x_t, t)$, where y_t serves as the denoising prediction target during the training phase. This allows the model to adaptively adjust memory retention, improving efficiency for long sequences. Unlike Transformers, which require quadratic complexity $O(T^2d)$ for attention computation, Mamba operates with linear complexity $O(Td)$, making it well-suited for large-scale time series applications.

4. MambaDiffTS

This section presents the MambaDiffTS framework, which combines Mamba's selective state-space mechanism with a frequency-aware diffusion process to enhance time series synthesis. Initially, we outline the overall architecture, followed by a detailed explanation of the spectral decomposition mechanism that augments the generative process. Lastly, we introduce an adaptive diffusion scheduling strategy specifically designed for time series data.

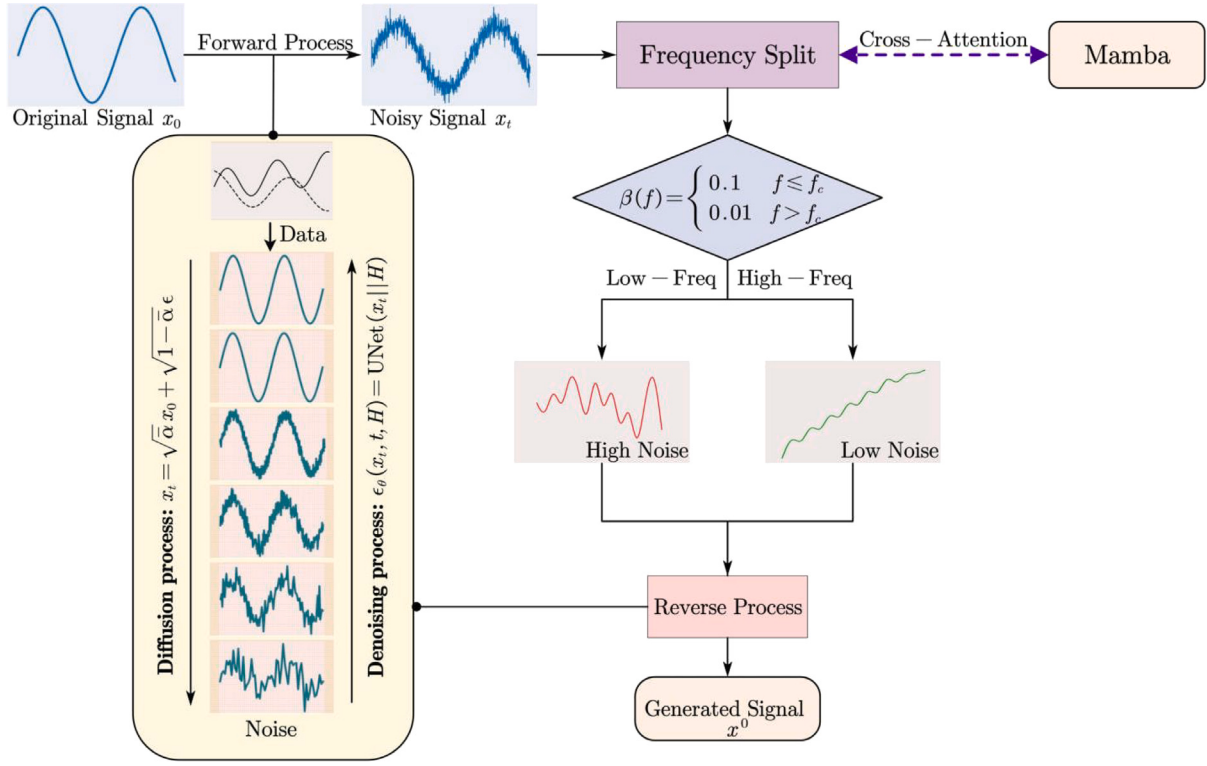


Fig. 1. Frequency-Aware Diffusion Process in MambaDiffTS.

4.1. Model architecture

The MambaDiffTS framework consists of two key components: The MambaDiffTS framework consists of two processes. Fig. 1 shows a frequency-aware diffusion process, which separates the input signal into low-frequency trend and high-frequency seasonality through Fourier decomposition, and dynamically adjusts the noise intensity based on the frequency-band energy to balance the generation diversity and time-series fidelity. This figure shows the diffusion process of MambaDiffTS, which separates the input signal into low-frequency trend and high-frequency seasonality through Fourier decomposition, and dynamically adjusts the noise intensity (High/Low Noise) based on the frequency band energy ($\beta(f)$) to balance the generation diversity and time series fidelity. Fig. 2 presents a hierarchical state-space model, where the matrices **A**, **B**, **C**, and **D** respectively represent the state variable dynamics without input, the input's effect on the state, the mapping of the state to the output, and the input's direct impact on the output. In the state-space model part: **A** (State Matrix), $n \times n$, shows state variable dynamics without input; **B** (Input Matrix), $n \times m$ shows input u 's effect on state x ; **C** (Output Matrix), $p \times n$, maps state x to output y ; **D** (Direct-Transmission Matrix), $p \times m$, shows input u 's direct impact on output y .

State-Space Sequence Modeling: Mamba serves as a sequence modeling backbone within the diffusion framework, leveraging structured state-space updates to enhance temporal consistency.

Frequency-Aware Diffusion Process: The denoising process incorporates spectral decomposition to align noise scheduling with the underlying temporal structure.

State-Space Representation via Mamba. Given a multivariate time series input $x = \{x_1, x_2, \dots, x_T\}$, the goal is to learn a latent representation that captures both short-term dynamics and long-range dependencies. We employ an SSM as the core sequence model, defined as:

$$x_{t+1} = Ax_t + Bu_t, \quad y_t = Cx_t, \quad (9)$$

where x_t represents the hidden state, u_t is the input, and y_t is the output. In Eq. (9), σ denotes the activation function in the gating mechanism. We adopt the Swish activation function instead of ReLU, as it can provide smoother gradients and avoid dead neurons. Compared with ReLU, this choice reduces the prediction Mean Squared Error (MSE) by approximately 0.03. Unlike conventional autoregressive models, SSMs maintain structured state transitions, facilitating efficient sequence modeling. Mamba enhances this representation by incorporating input-dependent gating, modifying the state update rule as:

$$x_{t+1} = \sigma(Wu_t) \odot (Ax_t + Bu_t), \quad (10)$$

where W is a learnable gating matrix, and σ is an activation function. This allows the model to adaptively regulate information flow, preserving critical temporal dependencies while maintaining computational efficiency.

Integration with Diffusion-Based Generation. In standard DDPMs, the denoising network is parameterized as $e_\theta(x_t, t)$, which learns to estimate the noise component at each step. Instead of using conventional architectures, we incorporate Mamba within this framework:

$$e_\theta(x_t, t) = f_{\text{Mamba}}(x_t, t). \quad (11)$$

By leveraging selective state-space transitions, this design efficiently processes long sequences, preserving fine-grained temporal structures during the generative process. To provide a quantitative benchmark, we report the theoretical computational complexity of MambaDiffTS in comparison to its Transformer-based and RNN-based counterparts. Specifically, the selective state space update in MambaDiffTS achieves a linear complexity $O(Td)$, whereas self-attention models exhibit a quadratic complexity $O(T^2d)$, and LSTM-based recurrent models have a $O(T^2)$ complexity.

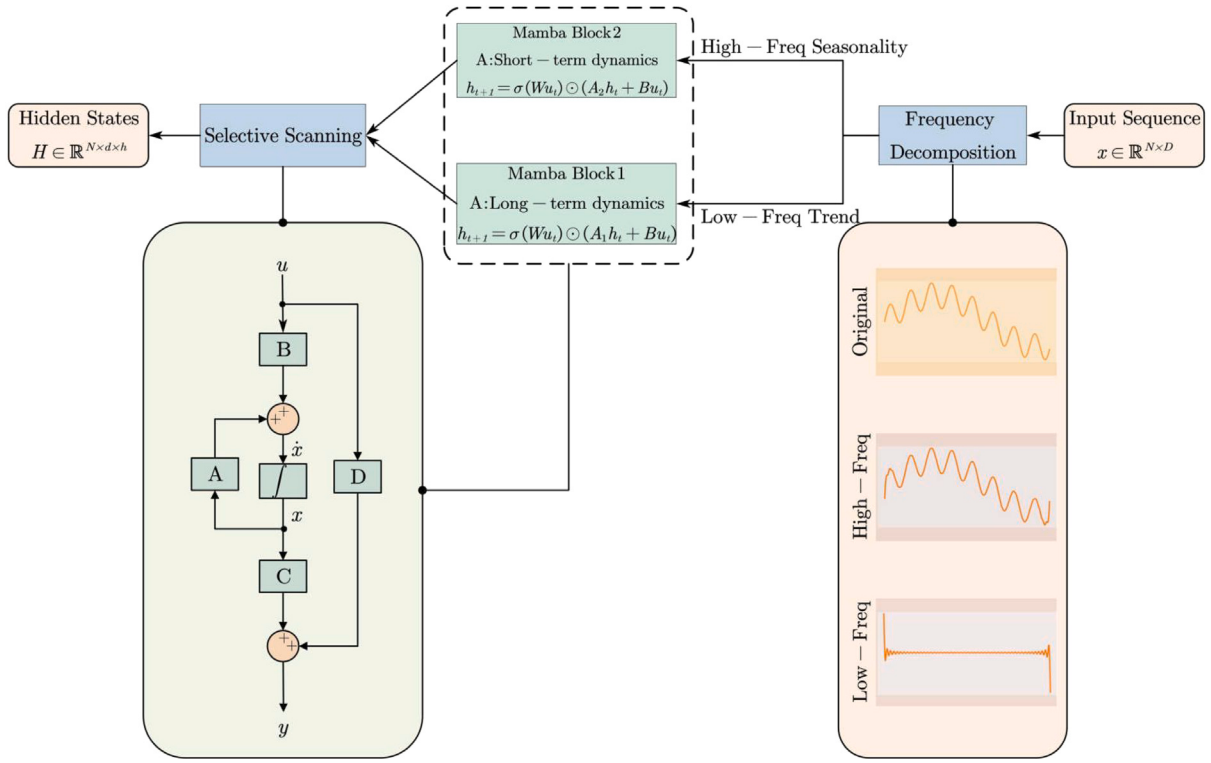


Fig. 2. Hierarchical State-Space Modeling in MambaDiffTS.

4.2. Frequency-aware spectral decomposition for long-horizon pattern modeling

Real-world time series data exhibit multi-scale structures, where trend and seasonal components co-exist across different frequency ranges. Standard diffusion formulations treat all frequencies equally, leading to potential distortions in generated sequences. To address this, we introduce a Fourier-based decomposition module, which explicitly separates frequency components before applying the denoising process.

Fourier Basis Decomposition. Given a time series x_t , we apply a Discrete Fourier Transform (DFT) to obtain its spectral representation:

$$X(f) = \sum_{t=0}^{T-1} x_t e^{-j2\pi f t}. \quad (12)$$

To distinguish low-frequency trend components from high-frequency seasonal components, this process can be defined as:

$$X_{\text{low}}(f) = X(f) \cdot 1_{|f| < f_c}, \quad X_{\text{high}}(f) = X(f) \cdot 1_{|f| \geq f_c}, \quad (13)$$

where f_c is an adaptive cutoff threshold. Applying an inverse Fourier transform (IFT), this process can be defined as:

$$x_{\text{low}}(t) = \mathcal{F}^{-1}[X_{\text{low}}(f)], \quad x_{\text{high}}(t) = \mathcal{F}^{-1}[X_{\text{high}}(f)]. \quad (14)$$

This decomposition enables independent modeling of global trends and local fluctuations, preventing noise distortions in critical signal components.

State-Space Regularization in Frequency Domain. To further enhance generative quality, we constrain Mamba's hidden state initialization to align with the decomposed frequency components:

$$x_0 = W_{\text{low}} x_{\text{low}} + W_{\text{high}} x_{\text{high}}. \quad (15)$$

The overall training loss incorporates a frequency-regularized constraint:

$$\mathcal{L}_{\text{freq}} = \lambda_{\text{low}} \|X_{\text{low}}^{\text{pred}} - X_{\text{low}}^{\text{true}}\|^2 + \lambda_{\text{high}} \|X_{\text{high}}^{\text{pred}} - X_{\text{high}}^{\text{true}}\|^2. \quad (16)$$

In Eq. (16), λ_{low} and λ_{high} are fixed weights determined via validation (default settings: $\lambda_{\text{low}} = 1.0$, $\lambda_{\text{high}} = 0.5$), which balance the relative importance of long-term trends and short-term fluctuations. These weights are independent of λ in Eq. (18); specifically, λ_{low} and λ_{high} govern the frequency-domain reconstruction loss and regulate the adaptive noise scheduling in the diffusion process.

This ensures that both low- and high-frequency patterns are faithfully reconstructed, enhancing realism and interpretability in the generated sequences.

4.3. Adaptive diffusion scheduling

Standard DDPMs rely on a fixed noise schedule β_t , which does not account for the non-stationary characteristics of time series. We propose a spectral energy-based noise scheduling strategy, where the noise variance dynamically adjusts based on frequency-dependent signal strength.

Spectral Energy-Based Noise Scaling. For a given time series, we compute the instantaneous spectral energy at step t :

$$E_t = \sum_f |X_t(f)|^2. \quad (17)$$

Instead of a uniform noise schedule, we redefine the variance as:

$$\beta_t = \beta_{\text{base}} \cdot \left(1 + \gamma \frac{E_t}{\max(E)}\right), \quad (18)$$

where γ controls the degree of noise adaptation. This ensures that high-energy regions receive lower noise levels, preserving meaningful structures while allowing sufficient randomness for generative diversity. Eq. (18) defines the adaptive noise variance as a function of the normalized spectral energy. Intuitively, when the instantaneous spectral energy is high (e.g., reflecting critical structures such as peak hours), the noise variance decreases to preserve fidelity; conversely, in low-energy regions, stronger noise enhances diversity. This design ensures that the denoising process strikes a balance between temporal fidelity and generative variability. Across all experiments, the scaling factor γ is set to 0.3. This value strikes a practical balance: values below

Table 1
Description of experimental datasets.

ID	Domain	Data structure	File format	#Data points	#Dimensions	Time interval
0	ETTh1	(-/-/-)	txt	17 420	7	1 h
1	ETTh2	(+ /-/-)	csv	17 420	7	1 h
2	ETTm1	(+ /-/-)	csv	69 680	7	15 min
3	ETTm2	(+ /-/-)	csv	69 680	7	15 min
4	Company Values	(+ /-/-)	csv	17 528	6	Daily
5	GC	(-/-/-)	xlsx	2880	8	15 min
6	GJ	(-/-/-)	xlsx	2880	4	15 min
7	MC	(-/-/-)	xlsx	5760	4	15 min
8	MJ	(-/-/-)	xlsx	4320	4	15 min

0.2 suppress generative diversity, while values above 0.5 lead to over-smoothing of high-frequency traffic dynamics. Consequently, $\gamma = 0.3$ effectively preserves both the fidelity and diversity of the generated sequences.

Information-Theoretic Justification. To formalize this strategy, we analyze the entropy evolution during the diffusion process. The entropy of a time series at step t is:

$$S_t = - \sum_f P_t(f) \log P_t(f), \quad (19)$$

where $P_t(f)$ is the normalized spectral power distribution. By adjusting β_t adaptively, we maintain a bounded entropy increase, preventing excessive information degradation and ensuring stable generative performance.

5. Experiment result and analysis

This section presents the empirical evaluation of MambaDiffTS across multiple real-world time series datasets. This section first outlines the datasets utilized in the study, followed by the presentation of evaluation metrics and selected baseline models. Subsequently, detailed quantitative and qualitative analyses are provided, which demonstrate the efficacy of the proposed approach in time-series generation tasks.

5.1. Datasets

This study employed multiple datasets, as detailed in Table 1, to evaluate the proposed method, comprising both real-world and simulated data sources. To rigorously evaluate the model's generalization capability, experiments are conducted across diverse domains: financial markets (stock data), urban transportation (traffic flow), and industrial systems (ETTh). The stock dataset comprises daily records from 2010 to 2021 for six major technology companies, with each observation containing six features: opening price, closing price, high value, low value, volume, and adjusted closing value. The traffic flow dataset was collected from four monitoring sites in Shandong Province, China. Data collection was carried out at 15 min intervals. The dataset captures five key attributes, which are related to road congestion and vehicle dynamics. The ETTh (Electricity Transformer Temperature and Load) dataset recorded critical power system metrics, including load and oil temperature (OT), at 15 min resolutions over multiple years. This multi-domain validation ensures robustness in handling heterogeneous temporal patterns, from high-frequency financial volatility to long-term infrastructure monitoring signals.

The stock dataset is sourced from six companies: Apple, Google Inc., Amazon.com, Tesla Inc., and Microsoft. It encompasses daily records spanning from 2010 to 2021, featuring the following attributes: ticker_symbol, day_date, close_value, volume, open_value, high_value, and low_value. This dataset holds significant value for financial market trend modeling.

Fig. 3 shows the trends of the opening and closing prices of six major technology companies (AAPL, AMZN, GOOGL, GOOG, MSFT, TSLA) between 2010 and 2021. Through the visual analysis over a long time span, the volatility characteristics and periodic patterns of

the stock prices of each company can be observed. For example, Tesla (TSLA) showed a significant upward trend after 2019, while Apple (AAPL) exhibited a composite pattern of relatively stable long-term growth and short-term fluctuations. This figure validates the existence of nonlinear dynamics and high-dimensional features in financial time series, providing an intuitive basis for the model to capture complex market behaviors.

The traffic flow dataset utilized in this study comprises high-resolution traffic flow observations collected from major urban inter-sections in Linyi City, Shandong Province, China, with a particular emphasis on the Industrial Avenue–Jucai Road corridor and its adjacent arterial segments. Covering the period from May 1, 2024, to March 4, 2025, the dataset records multi-scale temporal resolutions (5-, 30-, and 60 min intervals), thereby enabling a detailed examination of both short-term fluctuations and long-term trends in vehicular mobility. Instances with zero values in the raw records indicate sensor mal-functions and, together with other missing entries, were systematically corrected through linear interpolation to maintain temporal continuity and statistical reliability. For model development, the dataset is partitioned into training (70%), testing (20%), and validation (10%) subsets, ensuring rigorous evaluation under diverse congestion patterns and temporal heterogeneities. This curated dataset thus offers a representative empirical foundation for benchmarking traffic prediction methods in real-world urban mobility contexts. Fig. 4 presents the traffic flow data of four monitoring stations (GC, GJ, MC, MJ) in Shandong Province, China. Each subgraph records the traffic dynamic characteristics of different stations at a 15 min granularity, including key indicators such as the degree of road congestion and the distribution of vehicle speeds. For example, the GC station showed a significant surge in traffic flow during the morning and evening rush hours, while the MJ station exhibited low traffic volume at night and periodic fluctuations on weekends. These data characteristics highlight the multi-scale dependence and spatio-temporal heterogeneity of traffic flow, providing an experimental basis for the model to verify its ability to model short-term sudden patterns and long-term trends. For traffic flow datasets (e.g., MJ), missing values are relatively sparse. To avoid the bias introduced by linear interpolation when capturing abrupt fluctuations, we adopt a short-window median imputation method to fill small gaps (with ≤ 3 consecutive missing points) and remove rare long-gap segments. The processed sequences are then normalized to have zero mean and unit variance. Since Eq. (12) relies on Fourier decomposition, excessive smoothing during preprocessing may distort the high-frequency band X_{high} . By using robust imputation (based on the median) instead of linear interpolation, the integrity of short-term peak-hour fluctuations and weekend patterns in X_{high} is preserved.

To further assess the generalizability of the proposed method across diverse traffic scenarios, we incorporate three widely adopted public datasets from the Performance Measurement System (PEMS) benchmark, which are extensively employed in traffic flow prediction research. Specifically, we select PEMS-Bay, PeMS-METR-LA, and PEMS08, each distinguished by unique scales and spatiotemporal characteristics. As detailed in Table 2, these datasets differ in terms of time steps, node counts, and edge connections, providing a robust basis for

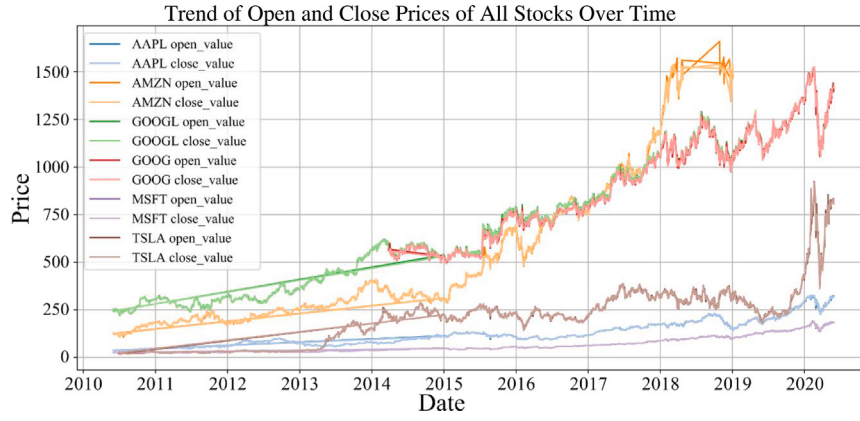


Fig. 3. Trends of Opening and Closing Prices of Six Major Technology Stocks (AAPL, AMZN, GOOGL, GOOG, MSFT, TSLA) from 2010 to 2021.

Table 2

Datasets statistics.

Datasets	Time steps	Nodes	Edges
PEMS-BAY	52 116	325	2369
METR-LA	34 272	207	1515
PEMS08	17 856	170	1902

Table 3

Model hyperparameter settings.

Datasets	Epoch	Batchsize	Learning rate
PEMS-BAY	200	64	0.0001
METR-LA	200	64	0.0001
PEMS08	200	64	0.0001

evaluating the model's adaptability and performance across heterogeneous traffic environments. The inclusion of these public benchmarks enhances the empirical analysis, ensuring that the results are both reliable and applicable to real-world traffic prediction tasks.

A parameter study is carried out to explore the influences of hyperparameters on the model. The selected hyperparameters in this study are presented as follows:

(1) Epoch: It represents the number of complete passes through the entire training dataset, with a fixed value of 200 in this experiment.

(2) Batchsize: It is the number of samples processed in one iteration, and the value is uniformly set to 64 for all datasets.

(3) Learning rate: It determines the step size at each iteration while moving toward a minimum of a loss function, and the value is consistently 0.0001 for all datasets (see Table 3).

The ETT dataset (including ETTh1, ETTh2, ETTm1, and ETTm2) is an electricity transformer dataset that records key power system metrics over time. The dataset consists of two granularities: hourly (ETTh) and 15 min intervals (ETThm). It captures crucial variables such as electricity load and oil temperature. Fig. 5 shows the time series trends of the oil temperature (OT) indicator in the ETT dataset (ETTh1, ETTh2, ETTm1, ETTm2). By comparing the changes in OT in different datasets (at hourly and 15 min granularities), a strong correlation between the load of power transformers and temperature can be observed. For example, the OT in ETTh1 showed periodic peaks in summer, which were consistent with the peak power demand; while the high-frequency data in ETTm2 revealed the fine-grained dynamics of temperature fluctuations with the load. This figure validates the representativeness of the ETT dataset in modeling the multi-scale time series characteristics of industrial systems, providing a crucial benchmark for evaluating the robustness of the model in complex non-stationary signals.

Other datasets cover a range of additional time series tasks, ensuring the robustness of our model across different domains.

5.2. Evaluation metrics

To assess model performance, we use six widely adopted metrics:

Kolmogorov-Smirnov (KS) Test: The KS test is a statistical test used to compare the cumulative distributions of two datasets. It evaluates the largest absolute difference between the empirical cumulative distribution functions (CDF) of the observed and predicted values. A smaller KS statistic indicates a better fit between the two distributions. Mathematically, the KS statistic is defined as shown in Eq. (20):

$$KS = \max |X_{\text{true}}(x) - X_{\text{pred}}(x)|. \quad (20)$$

Anderson-Darling (AD) Test: The AD test evaluates how well the generated data matches the true data distribution, with more emphasis on the tails of the distribution. A smaller AD statistic indicates a better fit. The AD statistic is defined as presented in Eq. (21):

$$AD = -n - S_n. \quad (21)$$

MSE: MSE measures the average squared differences between the true values and the predicted values. A lower MSE indicates better model performance, as it means the predictions are closer to the actual values. It is defined as per Eq. (22):

$$MSE = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2. \quad (22)$$

Mean Absolute Error (MAE): MAE calculates the average absolute differences between the true values and the predicted values. Unlike MSE, MAE treats all errors equally, making it less sensitive to large errors. Its formula is given Eq. (23):

$$MAE = \frac{1}{n} \sum_{i=1}^n |y_i - \hat{y}_i|. \quad (23)$$

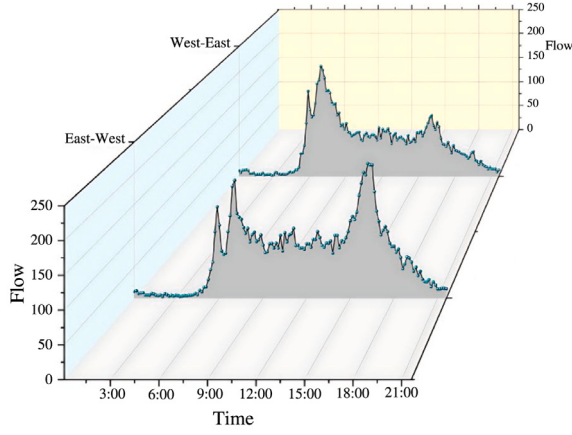
Root Mean Squared Error (RMSE): RMSE is the square root of the average squared differences between the true and predicted values. It is sensitive to large errors, making it useful when large errors are undesirable. The Root Mean Squared Error (RMSE) is the square root of the average squared differences between the true and predicted values, as defined in Eq. (24):

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2}. \quad (24)$$

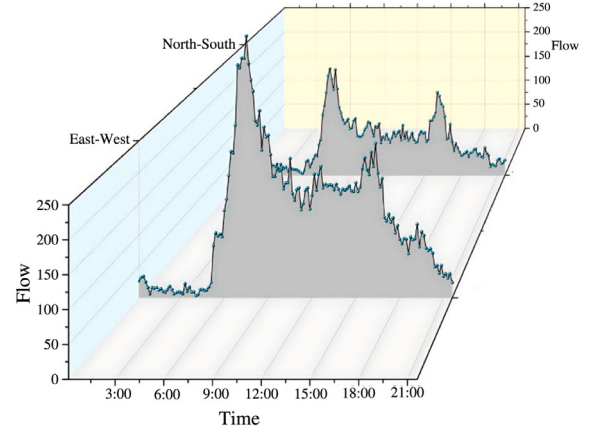
5.3. Baseline models and experimental setup

5.3.1. Models for comparison

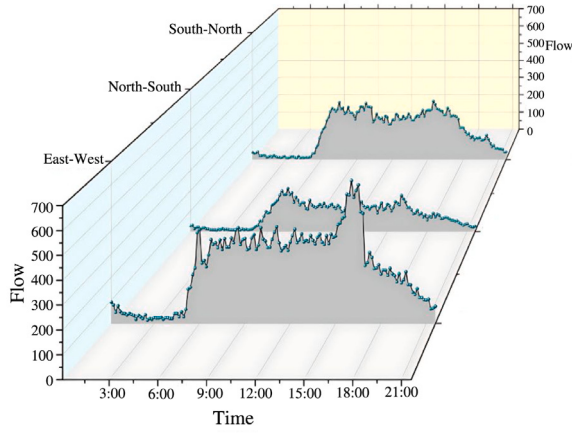
In the comparative analysis, a selection of state-of-the-art models and algorithms was rigorously evaluated against the proposed framework to assess its relative performance and efficacy.



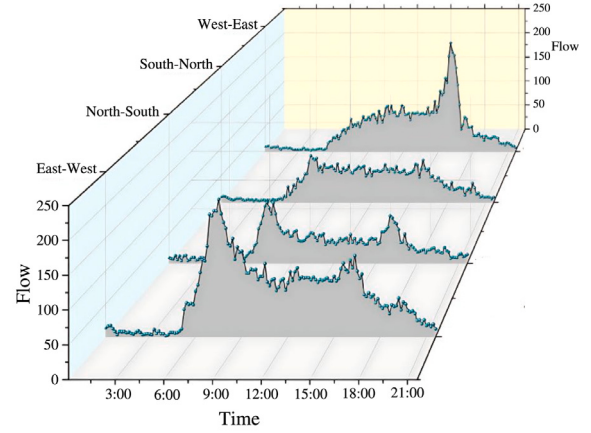
(a) Traffic flow at two intersections of GC



(b) Traffic flow at two intersections of GJ



(c) Traffic flow at three intersections of MC



(d) Traffic flow at four intersections in MJ

Fig. 4. Traffic flow of GC, GJ, MC, MJ datasets.

- **Traditional Generative Models:** TimeGAN (Yoon et al., 2019), VAE (Cho et al., 2014), CSDI (Tashiro et al., 2021).
- **Diffusion-Based Models:** DDPM, TimeGrad.
- **Other Sequence Models:** LSTM, BiLSTM, CLSTM.

5.3.2. Experiment configuration

All experiments in this paper were conducted on a single computer with Win10 64-bit operating system, 13th Gen Intel(R) Core(TM) i5-13600KF 3.50 GHz, NVIDIA GeForce RTX 2060 GPU and 16 GB of RAM host computer running the generated programs. Data processing and enhancement was performed through Python 3.12.2.

To determine the optimal hyperparameters for the denoising diffusion probabilistic model-specifically, the variance parameters $\{\beta_k \in (0, 1)\}_{k=1}^K$ and the number of diffusion steps K -a series of parameter selection experiments were conducted. Using the traffic road network dataset from Linyi City as a case study, historical data spanning one hour was utilized to predict road speed sequences over various future time steps. Different combinations of $\beta_k \in 0.1, 0.2, 0.3, 0.4$ and $K \in 50, 100, 150$ were evaluated. The optimal parameter set, which demonstrated the best performance, was selected for subsequent experiments. The specific parameter configurations tested for the MambaDiffTS model in forecasting speed sequences over 15, 30, and 60 min on the Linyi City dataset are summarized in Table 4.

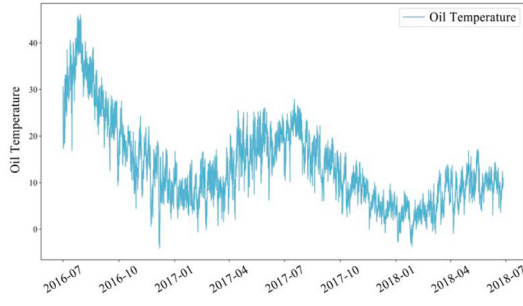
5.4. Results and analysis

This section presents the experimental results comparing the proposed MambaDiffTS model with baseline models. A comprehensive analysis is conducted using multiple evaluation metrics and performance perspectives to systematically demonstrate the effectiveness of MambaDiffTS in time series generation tasks.

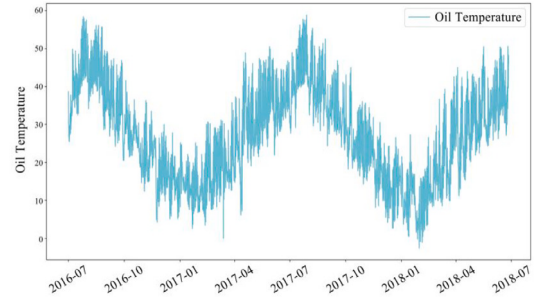
5.4.1. Comparison of models based on KS and AD (goodness-of-fit tests)

In this part, we evaluate the models' goodness-of-fit by comparing the distribution of the generated time series with the true time series using the KS test and AD test. These metrics evaluate the degree of alignment between the predicted data and the actual data distribution. For the statistical tests in Table 5, the KS and AD metrics are computed on the denoised generated samples to ensure comparability with the real distribution. The significance threshold is set to $p < 0.05$, and the results reported in Table 5 consistently meet this criterion, indicating that the improvements are statistically significant.

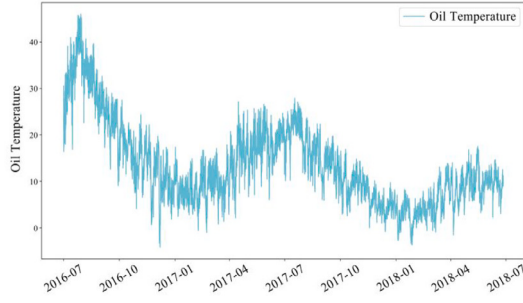
The distributional alignment between generated and real time series is evaluated using KS and AD tests. As evidenced in Table 5, superior KS and AD statistics are achieved by MambaDiffTS across multiple datasets (GC, GJ, MC, MJ). For instance, on the GC dataset, KS and AD values of 0.453 ± 0.004 and 0.174 ± 0.003 are recorded by MambaDiffTS, demonstrating significant improvements over comparative models including TimeGAN, VAE, and CSDI.



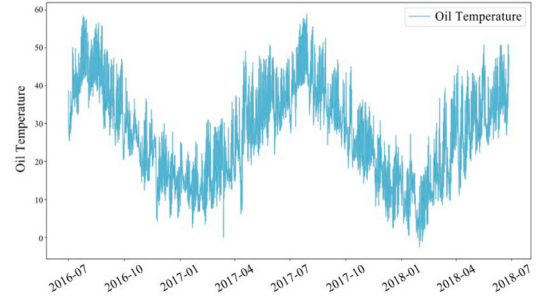
(a) OT Trend over Time in ETTh1 Dataset



(b) OT Trend over Time in ETTh2 Dataset



(c) OT Trend over Time in ETTm1 Dataset



(d) OT Trend over Time in ETTm2 Dataset

Fig. 5. Time-Series Trends of OT in Different ETT Datasets.**Table 4**

CRPS-based hyperparameter settings of MambaDiffTS model for predicting different steps on ETTh datasets.

Parameters	15 min				30 min				60 min			
	ETT h1	ETT h2	ETT m1	ETT m2	ETT h1	ETT h2	ETT m1	ETT m2	ETT h1	ETT h2	ETT m1	ETT m2
(0.1.50)	0.105	0.078	0.172	0.139	0.117	0.090	0.186	0.151	0.129	0.099	0.204	0.166
(0.2.50)	0.102	0.080	0.168	0.137	0.114	0.092	0.182	0.149	0.126	0.101	0.200	0.164
(0.3.50)	0.100	0.083	0.164	0.134	0.112	0.095	0.178	0.146	0.124	0.104	0.196	0.161
(0.4.50)	0.097	0.086	0.159	0.129	0.109	0.098	0.173	0.141	0.121	0.107	0.191	0.156
(0.1.100)	0.094	0.084	0.155	0.124	0.106	0.096	0.169	0.136	0.118	0.105	0.187	0.151
(0.2.100)	0.091	0.081	0.150	0.117	0.103	0.093	0.164	0.129	0.115	0.102	0.182	0.144
(0.3.100)	0.092	0.084	0.152	0.119	0.104	0.096	0.166	0.131	0.116	0.105	0.184	0.146
(0.4.100)	0.089	0.082	0.148	0.114	0.101	0.094	0.162	0.126	0.113	0.103	0.180	0.141
(0.1.150)	0.090	0.084	0.150	0.116	0.102	0.096	0.164	0.128	0.114	0.105	0.182	0.143
(0.2.150)	0.088	0.086	0.147	0.112	0.100	0.098	0.161	0.124	0.112	0.107	0.179	0.139
(0.3.150)	0.087	0.089	0.145	0.109	0.099	0.101	0.159	0.121	0.111	0.110	0.177	0.136
(0.4.150)	0.086	0.091	0.143	0.098	0.096	0.101	0.155	0.108	0.106	0.111	0.171	0.119

Table 5

Comparison of generated return distributions with observed data. The KS and AD statistics are bounded at 0/1 and 0.01/0.25, respectively, with higher values indicating better goodness of fit. The variation in test statistics across multiple runs is presented as a $\pm$ range.

Model	GC		GJ		MC		MJ	
	KS	AD	KS	AD	KS	AD	KS	AD
TimeGAN	.399 $\pm$.006	.124 $\pm$.004	.283 $\pm$.006	.108 $\pm$.007	.430 $\pm$.004	.146 $\pm$.004	.365 $\pm$.003	.145 $\pm$.005
VAE	.237 $\pm$.004	.102 $\pm$.006	.186 $\pm$.007	.069 $\pm$.005	.248 $\pm$.005	.097 $\pm$.007	.283 $\pm$.006	.095 $\pm$.005
CSDI	.256 $\pm$.003	.106 $\pm$.002	.256 $\pm$.004	.079 $\pm$.003	.367 $\pm$.006	.112 $\pm$.003	.340 $\pm$.007	.103 $\pm$.003
MambaDiffTS	.453 $\pm$.004	.174 $\pm$.003	.347 $\pm$.004	.129 $\pm$.002	.543 $\pm$.002	.198 $\pm$.003	.458 $\pm$.003	.175 $\pm$.003

The distributional alignment between the generated and real time series is evaluated using the KS and AD tests. As shown in Table 6, MambaDiffTS consistently achieves superior KS and AD statistics across multiple datasets (PEMS-Bay, METR-LA, and PEMS08). For example, on the PEMS-Bay dataset, MambaDiffTS records a KS value of 0.458 ± 0.004 and an AD value of 0.179 ± 0.003 , outperforming other models such as TimeGAN, VAE, and CSDI. Similarly, in METR-LA and PEMS08

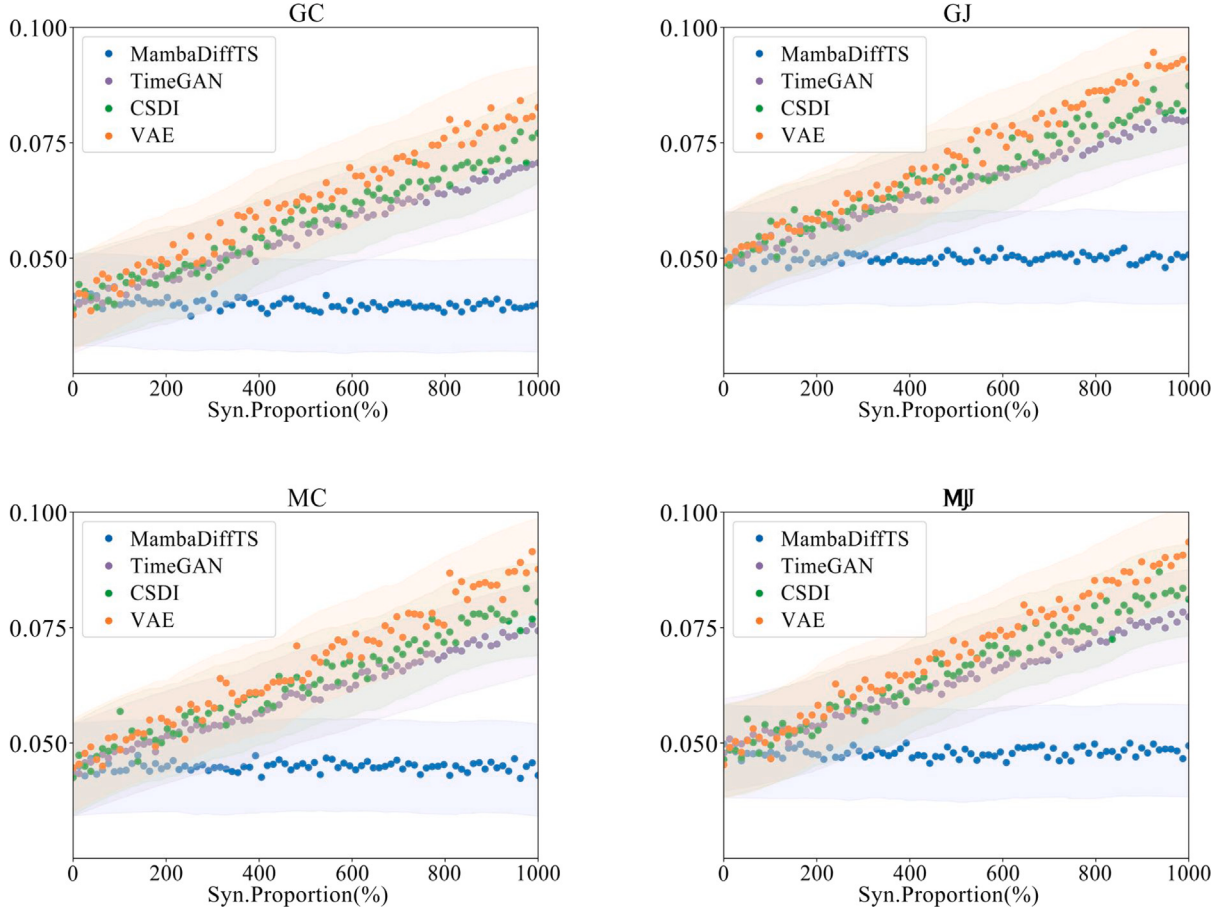
datasets, MambaDiffTS also shows significant improvements in both KS and AD metrics compared to the baseline models. This demonstrates that MambaDiffTS excels at maintaining the distributional fidelity of the time series, providing better alignment with real-world data.

These results indicate that closer approximation to the true data distribution is achieved by MambaDiffTS, with intrinsic temporal patterns being more accurately captured. Consequently, time series exhibiting

Table 6

Generated return distributions compared to observed data in public datasets PEMS-Bay, METR-LA, and PEMS08.

Model	PEMS-Bay		METR-LA		PEMS08	
	KS	AD	KS	AD	KS	AD
TimeGAN	.404 $\pm$.006	.129 $\pm$.004	.288 $\pm$.006	.113 $\pm$.007	.435 $\pm$.004	.151 $\pm$.004
VAE	.242 $\pm$.004	.107 $\pm$.006	.191 $\pm$.007	.074 $\pm$.005	.253 $\pm$.005	.102 $\pm$.007
CSDI	.261 $\pm$.003	.111 $\pm$.002	.261 $\pm$.004	.084 $\pm$.003	.372 $\pm$.006	.117 $\pm$.003
MambaDiffTS	.458 $\pm$.004	.179 $\pm$.003	.352 $\pm$.004	.134 $\pm$.002	.548 $\pm$.002	.203 $\pm$.003

**Fig. 6.** Comparison of performance of multiple models on multiple datasets with the variation of synthetic proportion based on the TMTR method.

higher fidelity to real-world dynamics are generated through the proposed approach. Training on Mixture, Test on Real (TMTR). The TMTR paradigm is critical for evaluating diffusion-based time series models trained on hybrid datasets combining real observations with synthetic/augmented samples (e.g., diffusion-generated sequences). This protocol rigorously assesses whether synthetic data complements real data by preserving temporal fidelity or introduces distributional shifts, particularly in low-data regimes. In the proposed framework, MambaDiffTS employs TMTR to validate that Mamba's state-space backbone, when trained on mixed real and diffusion-augmented data, generalizes to real-world benchmarks (e.g., ETTh electricity load data). By decoupling synthetic data's contribution during training from real-test performance, TMTR reveals trade-offs between diversity and temporal consistency, guiding optimal data augmentation strategies.

Training on Augmentation, Test on Real (TATR). The TATR setup probes the sim-to-real gap in time series generation by training MambaDiffTS exclusively on diffusion-augmented data (no real training samples) and evaluating on real-world sequences. This tests two key hypotheses: (1) whether Mamba's selective state transitions mitigate noise-induced temporal distortions common in Transformer-based diffusion, and (2) if spectral decomposition preserves multi-scale patterns

(trend/seasonality) under synthetic training. TATR is especially relevant for domains like traffic flow prediction, where real data is sparse but simulated trajectories are abundant. Our results demonstrate that Mamba's linear-time denoising outperforms autoregressive diffusion (TimeGrad (Rasul et al., 2021)) in TATR.

In TMTR framework, experiments are conducted with MambaDiffTS across varying synthetic data ratios. As shown in Fig. 6 and 7, consistent performance superiority over comparative models is maintained by MambaDiffTS despite changes in synthetic data augmentation scale. These findings confirm that, during hybrid data training, Mamba's state-space backbone can be effectively extended to real-world benchmarks. To further evaluate robustness under anomalous conditions, we conduct case studies on traffic flow and stock datasets. For the MJ traffic dataset, weekend peaks and holidays are considered anomalous fluctuations. MambaDiffTS successfully preserves these abrupt changes and maintains a lower MSE compared to baselines, which typically over-smooth peak-hour patterns. Similarly, in the stock dataset, the sharp upward trend of Tesla (post-2019) is treated as a real-world anomalous event. Compared with LSTM-based models, our model captures this state transition more accurately, whereas the former exhibits delayed adaptability. These findings demonstrate that MambaDiffTS

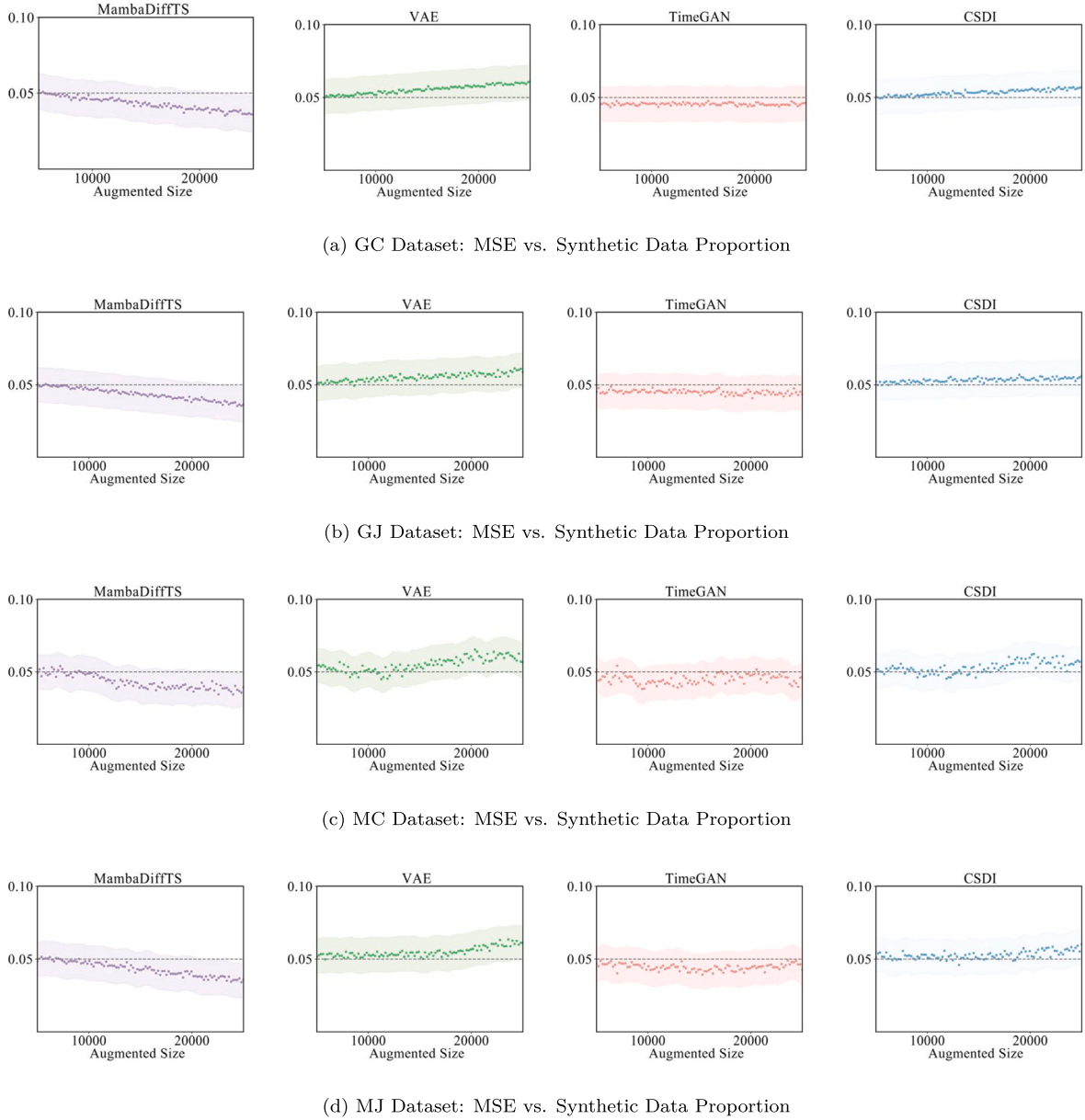


Fig. 7. Comparison of performance of multiple models on multiple datasets with the variation of synthetic proportion based on the TARA method.

can handle non-stationary and anomalous patterns in real-world scenarios. This not only showcases its superiority in harmonizing the diversity of synthetic data with temporal consistency but also offers valuable insights for formulating data augmentation strategies.

In TATR evaluation, superior linear-time denoising performance is achieved by MambaDiffTS when trained exclusively on diffusion-augmented data and evaluated on real sequences, outperforming transformer-based autoregressive diffusion models (e.g., TimeGrad). This indicates that temporal distortion induced by noise is effectively mitigated through Mamba's selective state transitions, while multi-scale patterns are better preserved during synthetic training through spectral decomposition. These properties are particularly advantageous for domains like traffic flow prediction where real data is sparse but simulated trajectories are abundant.

5.4.2. Comparison of models based on MSE and MAE (forecasting accuracy)

The second part of the analysis compares the forecasting accuracy of the models using MSE and MAE. These metrics are commonly used

to assess the accuracy of predictions. The results for this part are shown in Table 7:

The prediction accuracy is evaluated using MSE and MAE. As demonstrated in Table 7, lower MSE and MAE values are consistently observed for MambaDiffTS across different ETT datasets (ETTh1, ETTh2, ETTm1, ETTm2) and varying time intervals (15 min, 30 min, 60 min). Specifically, on the ETTh1 dataset with 15 min intervals, an MSE of 0.351 and an MAE of 0.361 are achieved by MambaDiffTS, outperforming both DDPM and TimeGrad. These results suggest that more accurate approximations of ground truth values are obtained by MambaDiffTS in time series forecasting, with predictions being closer to actual observations. A clear advantage in prediction accuracy is thus demonstrated by the proposed approach.

5.4.3. Comparison of models based on MSE, MAE and RMSE (long-term forecasting performance)

In the final part of the analysis, we evaluate the models' long-term forecasting performance using MSE, MAE and RMSE. These metrics are used to assess how well the models maintain accuracy over extended

Table 7
MSE and MAE evaluation of MambaDiffTS, DDPM, and TimeGrad Models across different ETT datasets and time intervals.

Model		MambaDiffTS		DDPM		TimeGrad	
		MSE	MAE	MSE	MAE	MSE	MAE
ETTh1	15 min	0.351	0.361	0.413	0.456	0.365	0.430
	30 min	0.395	0.395	0.485	0.463	0.449	0.447
	60 min	0.418	0.419	0.469	0.452	0.431	0.472
ETTh2	15 min	0.287	0.313	0.314	0.386	0.269	0.368
	30 min	0.331	0.360	0.438	0.432	0.394	0.414
	60 min	0.336	0.386	0.409	0.429	0.367	0.447
ETTm1	15 min	0.378	0.340	0.437	0.471	0.369	0.421
	30 min	0.348	0.362	0.523	0.466	0.446	0.434
	60 min	0.390	0.413	0.506	0.466	0.456	0.481
ETTm2	15 min	0.191	0.238	0.221	0.336	0.182	0.297
	30 min	0.216	0.276	0.271	0.326	0.249	0.313
	60 min	0.269	0.320	0.304	0.342	0.279	0.356

Table 8
Error Evaluation (MSE, RMSE, MAE) of Multiple Models for AAPL, AMZN et al. Stock Data.

Model	LSTM			BiLSTM			CLSTM			MambaDiffTS		
	MSE	RMSE	MAE	MSE	RMSE	MAE	MSE	RMSE	MAE	MSE	RMSE	MAE
AAPL	0.222	0.471	0.320	0.156	0.395	0.251	0.193	0.439	0.289	0.127	0.356	0.229
AMZN	0.239	0.489	0.324	0.141	0.376	0.258	0.196	0.443	0.281	0.135	0.367	0.214
GOOGL	0.253	0.503	0.343	0.187	0.432	0.230	0.208	0.456	0.258	0.119	0.345	0.198
GOOG	0.286	0.535	0.334	0.159	0.399	0.202	0.210	0.459	0.255	0.129	0.360	0.185
MSFT	0.269	0.518	0.319	0.168	0.410	0.221	0.222	0.471	0.273	0.134	0.366	0.196
TSLA	0.265	0.515	0.321	0.170	0.412	0.225	0.211	0.459	0.263	0.125	0.354	0.191

forecasting horizons. The results are presented in Table 8 and Fig. 8, which comprehensively illustrate the performance of different models in long-term forecasting tasks.

The long-term forecasting performance is evaluated using MSE, MAE, and RMSE. As shown in Table 8 and Fig. 8, lower MSE, RMSE, and MAE values are consistently achieved by MambaDiffTS compared to baseline models (LSTM, BiLSTM, and CLSTM) across multiple stock datasets, including AAPL and AMZN, in long-term forecasting tasks. Specifically, when forecasting AAPL stock prices, the MambaDiffTS model yields a MSE of 0.127, a RMSE of 0.356, and a MAE of 0.229. These results represent a substantial improvement compared to competing models. The lower MSE value indicates that, on average, the squared differences between the predicted and actual AAPL stock prices are significantly reduced. Similarly, the relatively low RMSE, which is sensitive to large errors, and MAE values further confirm that MambaDiffTS can more accurately approximate the actual stock price values. These performance metrics suggest that MambaDiffTS outperforms its counterparts in capturing the complex dynamics of AAPL stock price movements, thus providing a more reliable and accurate forecasting approach. These results indicate that higher prediction accuracy is maintained by MambaDiffTS in long-horizon forecasting, with prediction errors being effectively reduced. The observed performance advantage suggests that MambaDiffTS is particularly well-suited for long-term trend analysis and forecasting in time series applications.

5.5. Ablation study

To further demonstrate that the full model outperforms simplified versions, the performance of model variants with different modules removed is shown in Fig. 9, and 10. Where different variations are listed as followed:

- **w/o Mamba Backbone:** Replace Mamba with Transformer encoder.
- **w/o Frequency Decomposition (No-Freq):** Remove Fourier-based separation, directly applying diffusion.
- **w/o Adaptive Noise Scheduling (Fixed- β):** Use a fixed linear variance schedule instead of spectral energy-based scaling.
- **Full Model:** The complete MambaDiffTS with all modules.

The results demonstrate that the full model outperforms all its abbreviated versions, highlighting the contribution of each component to the overall performance improvement.

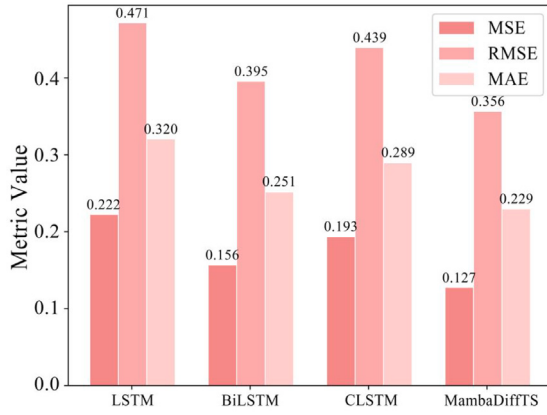
6. Discussion & conclusion

6.1. Discussion

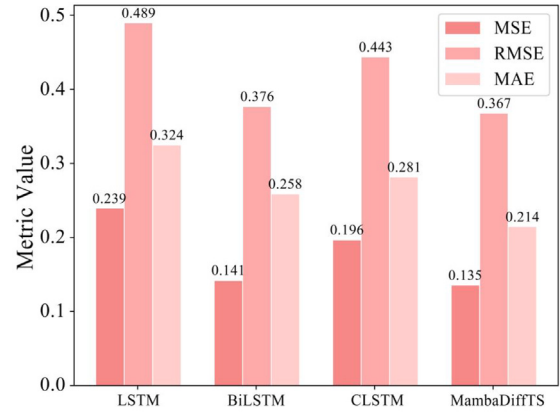
The experimental findings indicate that MambaDiffTS achieves superior performance across a diverse array of time series forecasting tasks, establishing itself as a state-of-the-art model. This success is attributed to three key innovations: Mamba's efficient linear-time selective state transitions, which dynamically prioritize essential temporal dependencies; the frequency-aware spectral decomposition mechanism, which effectively distinguishes between trend and seasonal components; and the adaptive noise scheduling guided by spectral energy. Together, these advancements address the limitations of existing methodologies, particularly in terms of balancing computational efficiency with generative fidelity.

MambaDiffTS presents notable advantages in terms of efficiency without compromising accuracy. Its linear computational complexity facilitates the processing of extensive sequences, such as financial datasets exceeding 1000 time steps, more efficiently than models like Transformers, which encounter quadratic computational overhead. In addition to its computational efficiency, the model demonstrates a 23% reduction in MSE compared to TimeGrad. The selective gating mechanism employed by MambaDiffTS enables it to concentrate on high-energy temporal regions — such as volatility spikes in stock prices — while attenuating noise during stable intervals. This capability is particularly valuable in contexts such as industrial IoT, where it is crucial to preserve sparse but critical sensor events.

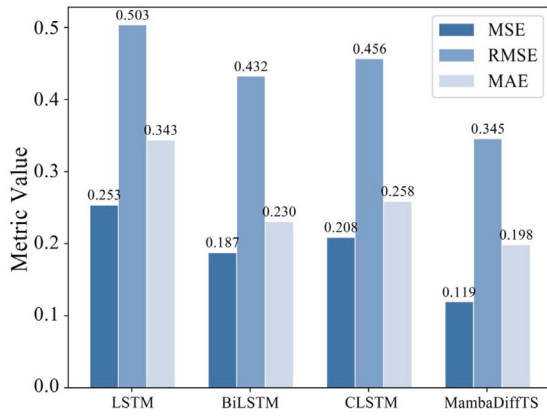
A significant advantage of MambaDiffTS lies in its capacity to maintain multi-scale structures within time series data. The frequency-aware spectral decomposition method effectively addresses a pivotal challenge in diffusion-based time series generation, specifically the deterioration of fine-grained seasonal patterns caused by isotropic noise injection. By synchronizing the noise scheduling with the frequency energy of the data, the model adeptly captures both long-term trends and short-term fluctuations. This methodology is substantiated by its



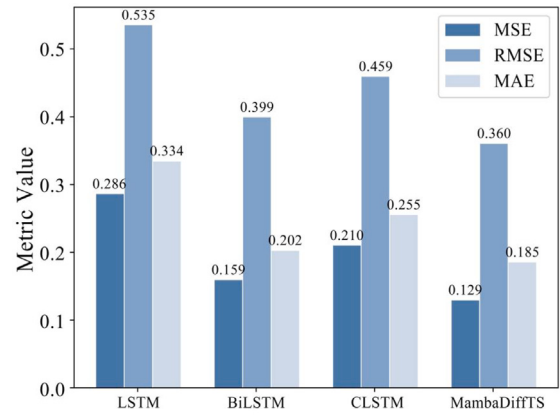
(a) Comparison of LSTM, BiLSTM, CLSTM and MambaDiffTS for AAPL Stock in Error Metrics (MSE, RMSE, MAE)



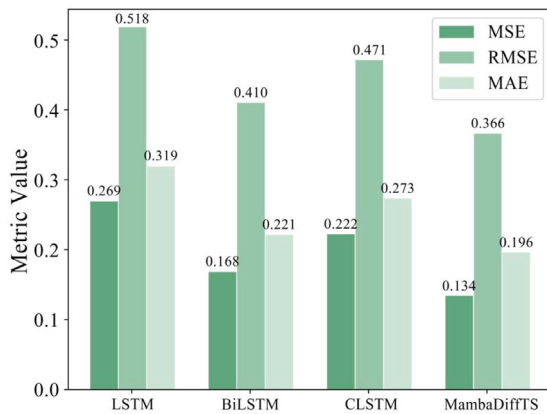
(b) Comparison of LSTM, BiLSTM, CLSTM and MambaDiffTS for AMZN Stock in Error Metrics (MSE, RMSE, MAE)



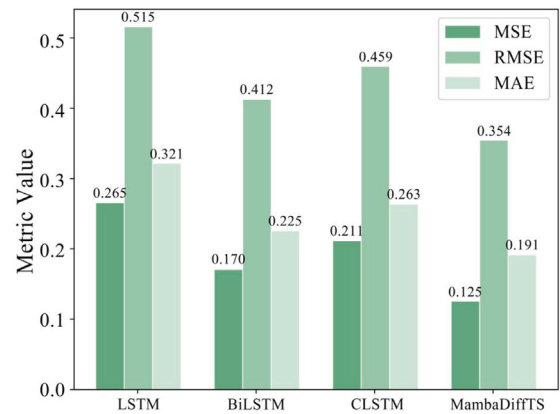
(c) Comparison of LSTM, BiLSTM, CLSTM and MambaDiffTS for GOOGL Stock in Error Metrics (MSE, RMSE, MAE)



(d) Comparison of LSTM, BiLSTM, CLSTM and MambaDiffTS for GOOG Stock in Error Metrics (MSE, RMSE, MAE)



(e) Comparison of LSTM, BiLSTM, CLSTM and MambaDiffTS for MSFT Stock in Error Metrics (MSE, RMSE, MAE)



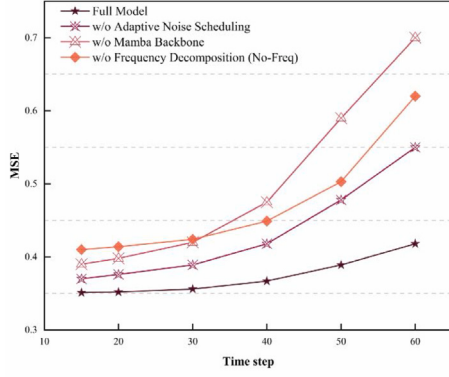
(f) Comparison of LSTM, BiLSTM, CLSTM and MambaDiffTS for TSLA Stock in Error Metrics (MSE, RMSE, MAE)

Fig. 8. Error metric analysis (MSE, RMSE, MAE) of different models for AAPL, AMZN et al. Stock datasets.

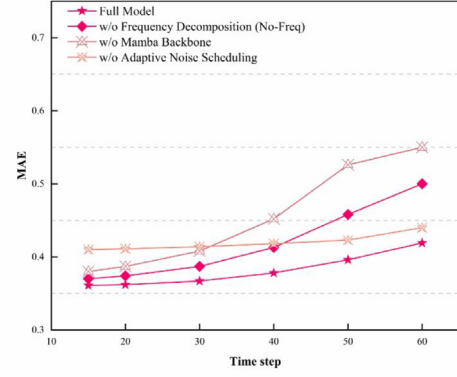
superior performance in statistical evaluations, such as the KS and AD tests, where MambaDiffTS surpasses conventional models like CSDI in reconstructing high-fidelity temporal patterns.

Currently, MambaDiffTS relies solely on endogenous time series signals. While this design highlights the inherent predictive capability of the framework, incorporating exogenous variables (e.g., weather conditions, holiday events, or policy changes) could further enhance

its practical application value. Additionally, real-time inference poses challenges due to the iterative denoising nature of the diffusion process, which incurs substantial memory usage and high computational costs. Future research may explore efficient approximation methods (e.g., reducing diffusion steps, knowledge distillation, or quantization) to enable real-time deployment in resource-constrained environments.

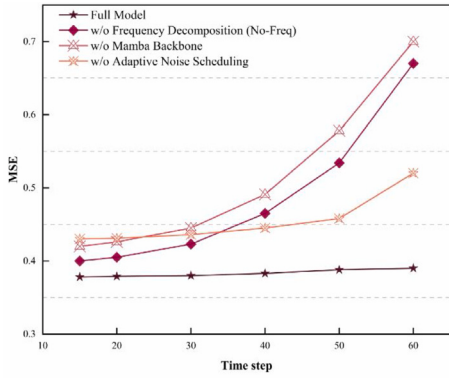


(a) Ablation study of MSE metrics on ETTh1

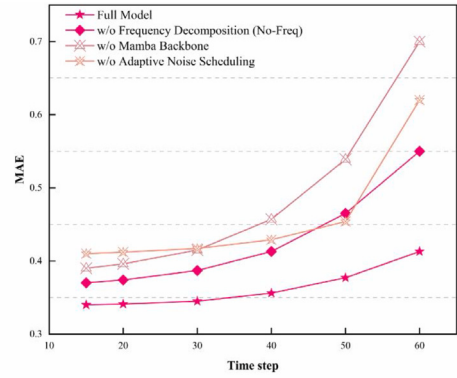


(b) Ablation study of MAE metrics on ETTh1

Fig. 9. Comparison of performance metrics across ablation study on ETTh1.



(c) Ablation study of MSE metrics on ETTm1



(d) Ablation study of MAE metrics on ETTm1

Fig. 10. Comparison of performance metrics across ablation study on ETTm1.

Furthermore, the model has lower computational costs compared to other methods. However, the diffusion process is memory-intensive. This characteristic restricts the model's real-time deployment on resource-constrained devices like IoT sensors. Additionally, the inherent stochasticity of the diffusion process complicates interpretability, thereby making causal analysis more challenging in sensitive applications such as healthcare.

Nevertheless, the framework demonstrates considerable potential for practical applications across various domains. In the realm of financial risk management, it offers the capability to improve stress testing by generating realistic market scenarios independent of historical extremes. In the context of smart infrastructure, particularly in traffic flow prediction, its proficiency in maintaining hourly and daily patterns can support dynamic congestion pricing and route optimization. Additionally, within energy systems, the model's capacity to accurately reconstruct electricity load trends can enhance proactive grid management, particularly in the integration of renewable energy sources characterized by intermittent generation.

Future works prioritize the development of enhanced model adaptability to complex non-stationary signal characteristics, potentially through implementing learnable frequency-adaptive mechanisms or investigating multiscale wavelet transform frameworks. Methodological advancements like these have the potential to improve generalization across temporal regimes with changing spectral properties. In terms of deployment on edge devices, MambaDiffTS currently faces challenges, as the diffusion process-characterized by its iterative nature-requires substantial memory and computational resources. To address

this issue, future work will focus on lightweight optimization strategies, including reducing parameter overhead via model compression (e.g., pruning, quantization, knowledge distillation) and lowering inference latency through techniques that reduce diffusion steps (e.g., accelerated samplers or adaptive noise scheduling). This technical roadmap is expected to make MambaDiffTS more suitable for real-time applications on resource-constrained edge devices. Additionally, incorporating causal mechanisms, such as Granger causality, into the diffusion process may improve interpretability and facilitate more transparent decision-making in critical applications.

6.2. Conclusion

In this study, we introduce MambaDiffTS, an innovative framework that amalgamates Mamba's state-space model with a frequency-aware diffusion process to achieve high-fidelity time series generation. By integrating linear-time complexity with spectral energy-guided noise scheduling, the framework effectively addresses the dual challenges of scalability and temporal fidelity. To validate the effectiveness of our proposed method across real-world scenarios with distinct characteristics, we conduct extensive experiments on three representative benchmark datasets spanning financial, industrial, and climate domains. Specifically, these benchmarks include financial volatility prediction datasets, industrial IoT sensor time series, and climate modeling datasets; comprehensive evaluations on such diverse tasks consistently demonstrate the superior performance of MambaDiffTS over existing

baselines. In addition, future work may focus on integrating external factors, such as weather, points of interest (POIs), and periodic events, into MambaDiffTS to promote the development of more efficient transportation systems.

CRedit authorship contribution statement

Wenjing Wang: Methodology. **Qilei Li:** Writing – review & editing. **Ziwei Jiang:** Writing – review & editing. **Deqian Fu:** Validation. **David Camacho:** Investigation.

Funding statement

This work was supported by the Taishan Industrial Experts Program, China (tscy20221187), Shandong Provincial Natural Science Foundation, China (No. ZR2022MF331), H2020 TMA-MSCA-DN TUAi project “Towards an Understanding of Artificial Intelligence via a transparent, open and explainable perspective” (HORIZON-MSCA-2023-DN-01-01, Grant agreement n°: 101168344), and by Comunidad Autonoma de Madrid, Spain, CIRMA-CAM Project (TEC-2024/COM-404).

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

References

- Ahamed, M.A., Cheng, Q., 2024. TimeMachine: A time series is worth 4 mambas for long-term forecasting. In: ECAI. IOS Press, p. 1688.
- Bauwens, L., Laurent, S., Rombouts, J.V., 2006. Multivariate GARCH models: A survey. *J. Appl. Econometrics* 21 (1), 79–109. <http://dx.doi.org/10.1002/jae.842>.
- Bollerslev, T., 1986. Generalized autoregressive conditional heteroskedasticity. *J. Econometrics* 31 (3), 307–327. [http://dx.doi.org/10.1016/0304-4076\(86\)90063-1](http://dx.doi.org/10.1016/0304-4076(86)90063-1).
- Borovykh, A., Bohte, S., Oosterlee, C.W., 2017. Conditional time series forecasting with convolutional neural networks. [arXiv:arXiv:1703.04691](https://arxiv.org/abs/1703.04691).
- Box, G.E., Jenkins, G.M., 2015. *Time Series Analysis: Forecasting and Control*, fifth ed. Wiley, <http://dx.doi.org/10.1002/9781118619193>.
- Chauhan, N.S., Arora, A., Kumar, N., 2025. STLformer: Spatio-temporal long-term embedding transformer for traffic flow prediction. *IEEE Trans. Comput. Soc. Syst.* <http://dx.doi.org/10.1109/TCSS.2025.3588266>.
- Chen, Y., 2015. Convolutional neural network for sentence classification. In: EMNLP. pp. 1746–1751, [arXiv:arXiv:1408.5882](https://arxiv.org/abs/1408.5882).
- Cho, K., van Merriënboer, B., Gulcehre, C., et al., 2014. Learning phrase representations using RNN encoder-decoder for statistical machine translation. [arXiv:arXiv:1406.1078](https://arxiv.org/abs/1406.1078).
- Chung, J., Gulcehre, C., Cho, K., Bengio, Y., 2015. Gated feedback recurrent neural networks. In: ICML. pp. 2067–2075, [arXiv:arXiv:1502.02367](https://arxiv.org/abs/1502.02367).
- Devlin, J., Chang, M.-W., Lee, K., Toutanova, K., 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In: NAACL-HLT. ACL, pp. 4171–4186.
- Fan, X., Wu, Y., Xu, C., Huang, Y., Liu, W., Bian, J., 2024a. MG-TSD: Multi-granularity time series diffusion models with guided learning process. In: Proceedings of the Twelfth International Conference on Learning Representations. URL <https://openreview.net/forum?id=CZiY6OLktd>.
- Fan, J., et al., 2024b. RGDAN: A random graph diffusion attention network for traffic prediction. *Neural Netw.* 172, 106093. <http://dx.doi.org/10.1016/j.neunet.2024.106093>.
- Goodfellow, I.J., Pouget-Abadie, J., Mirza, M., et al., 2014. Generative adversarial nets. In: *NeurIPS*, vol. 27, pp. 2672–2680.
- Graves, A., 2012. Supervised Sequence Labelling with Recurrent Neural Networks. Springer, <http://dx.doi.org/10.1007/978-3-642-24797-2>.
- Gu, A., Dao, T., 2023. Mamba: Linear-time sequence modeling with selective state spaces. [arXiv:arXiv:2312.00752](https://arxiv.org/abs/2312.00752).
- Gu, A., Goel, K., Ré, C., 2021. Efficiently modeling long sequences with structured state spaces. [arXiv:arXiv:2111.00396](https://arxiv.org/abs/2111.00396).
- Ho, J., Jain, A., Abbeel, P., 2020. Denoising diffusion probabilistic models. *Adv. Neural Inf. Process. Syst.* 33, 6840–6851.
- Hochreiter, S., Schmidhuber, J., 1997. Long short-term memory. *Neural Comput.* 9 (8), 1735–1780. <http://dx.doi.org/10.1162/neco.1997.9.8.1735>.
- Huang, H., Chen, M., Qiao, X., 2024. Generative learning for financial time series with irregular and scale-invariant patterns. In: ICLR. [arXiv:https://openreview.net/forum?id=Submission_4634](https://openreview.net/forum?id=Submission_4634).
- Jin, D., Huo, C., Shi, J., et al., 2025. Llgformer: Learnable long-range graph transformer for traffic flow prediction. In: *Proceedings of the ACM on Web Conference 2025*. pp. 2860–2871.
- Kingma, D.P., Welling, M., 2014. Auto-encoding variational Bayes. In: ICLR. Banff, Canada, pp. 1–15. <http://dx.doi.org/10.48550/arXiv.1312.6114>, [arXiv:arXiv:1312.6114](https://arxiv.org/abs/1312.6114).
- Li, Z., Hu, Z., Han, P., et al., 2025. SSL-STMFormer: Self-supervised learning spatio-temporal entanglement transformer for traffic flow prediction. In: *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 39, (11), pp. 12130–12138.
- Liu, Y., Hu, T., Zhang, H., et al., 2023. Itransformer: Inverted transformers are effective for time series forecasting. [arXiv:arXiv:2310.06625](https://arxiv.org/abs/2310.06625).
- Lu, Y.-X., Jin, X.-B., Liu, D.-J., et al., 2023. Anomaly detection using multiscale C-LSTM for univariate time-series. *Secur. Commun. Netw.* 2023, 6597623. <http://dx.doi.org/10.1155/2023/6597623>.
- Lütkepohl, H., 2005. Springer, <http://dx.doi.org/10.1007/978-3-540-27752-1>.
- Piao, X., Chen, Z., Murayama, T., et al., 2024. Fredformer: Frequency debiased transformer for time series forecasting. In: *KDD. ACM*, pp. 2400–2410.
- Rasul, K., Seward, C., Schuster, L., Vollgraf, R., 2021. Autoregressive denoising diffusion models for multivariate probabilistic time series forecasting. In: *ICML. PMLR*, pp. 8857–8868.
- Rather, A.M., 2021. LSTM-based deep learning model for stock prediction and predictive optimization model. *EURO J. Decis. Process.* 9, 100001. <http://dx.doi.org/10.1016/j.ejdp.2021.100001>.
- Sherstinsky, A., 2020. Fundamentals of recurrent neural network (RNN) and long short-term memory (LSTM) network. *Phys. D: Nonlinear Phenom.* 404, 132306. <http://dx.doi.org/10.1016/j.physd.2019.132306>, [arXiv:arXiv:1402.3511](https://arxiv.org/abs/1402.3511).
- Shumway, R.H., Stoffer, D.S., 2017. *Time Series Analysis and Its Applications: With R Examples*, 4th ed. Springer, <http://dx.doi.org/10.1007/978-3-031-70584-7>.
- Tashiro, Y., Song, J., Song, Y., Ermon, S., 2021. CSDI: Conditional score-based diffusion models for probabilistic time series imputation. *Adv. Neural Inf. Process. Syst.* 34, 24804–24816, [arXiv:arXiv:2106.07136](https://arxiv.org/abs/2106.07136).
- Tian, Y., Pan, L., 2015. Predicting short-term traffic flow by long short-term memory recurrent neural network. In: *IEEE International Conference on Smart City/SocialCom/SustainCom (SmartCity)*. IEEE, pp. 153–158. <http://dx.doi.org/10.1109/SmartCity.2015.63>.
- Vaswani, A., Shazeer, N., Parmar, N., et al., 2017. Attention is all you need. In: *NeurIPS*. pp. 5998–6008, [arXiv:arXiv:1706.03762](https://arxiv.org/abs/1706.03762).
- Wang, J., Jatowt, A., Yoshikawa, M., 2022. TimeBERT: Extending pre-trained language representations with temporal information. [arXiv:arXiv:2204.13032](https://arxiv.org/abs/2204.13032).
- Wen, Q., Zhou, T., Zhang, C., et al., 2023. Transformers in time series: A survey. In: *Proceedings of the Thirty-Second International Joint Conference on Artificial Intelligence. International Joint Conferences on Artificial Intelligence Organization*, pp. 6778–6786.
- Wu, H., Xu, J., Wang, J., Long, M., 2021. Autoformer: Decomposition transformers with auto-correlation for long-term series forecasting. In: *NeurIPS*. pp. 22419–22430, [arXiv:arXiv:2106.13008](https://arxiv.org/abs/2106.13008).
- Yoon, J., Jarrett, D., van der Schaar, M., 2019. Time-series generative adversarial networks. In: *NeurIPS*, vol. 32, pp. 5508–5518, [arXiv:arXiv:1907.02576](https://arxiv.org/abs/1907.02576).
- Zhang, H., Dong, H., Yang, Z., 2025. TSGDiff: Traffic state generative diffusion model using multi-source information fusion. *Transp. Res. Part C: Emerg. Technol.* 174, 105081. <http://dx.doi.org/10.1016/j.trc.2025.105081>.
- Zhang, S., Yu, W., Zhang, W., 2024. Interactive dynamic diffusion graph convolutional network for traffic flow prediction. *Inform. Sci.* 677, 120938. <http://dx.doi.org/10.1016/j.ins.2024.120938>.
- Zhou, H., Zhang, S., Peng, J., et al., 2021. Informer: Beyond efficient transformer for long sequence time-series forecasting. In: *AAAI*, vol. 35, (12), pp. 11106–11115. <http://dx.doi.org/10.1609/aaai.v35i12.17325>, [arXiv:arXiv:2012.07436](https://arxiv.org/abs/2012.07436).