Decomposed Spatio-Temporal Mamba for Long-Term Traffic Prediction

Sicheng He[1], **Junzhong Ji**[1,2], **Minglong Lei**[1,2*]

[1]College of Computer Science, Beijing University of Technology, Beijing, China
[2]Beijing Institute of Artificial Intelligence, Beijing University of Technology, Beijing, China
sicheng_he@emails.bjut.edu.cn, jjz01@bjut.edu.cn, leiml@bjut.edu.cn

Abstract

Traffic prediction provides vital support for urban traffic management and has received extensive research interest. By virtue of the ability to effectively learn spatial and temporal dependencies from a global view, Transformers have achieved superior performance in long-term traffic prediction. However, existing methods usually underrate the complex spatio-temporal entanglement in long-range sequences. Compared with purely temporal entanglement, spatio-temporal data emphasizes the entangled dynamics under the restrictions of traffic networks, which brings additional difficulties. Moreover, the computational costs of spatio-temporal Transformers scale quadratically as the sequence length grows, limiting their applications on long-range and large-scale scenarios. To address these problems, we propose a decomposed spatio-temporal Mamba (DST-Mamba) for traffic prediction. We aim to apply temporal decomposition to the entangled sequences and obtain the seasonal and trend parts. Shifting from the temporal view to the spatial view, we leverage Mamba, a state space model with near-linear complexity, to capture seasonal variations in a node-centric manner. Meanwhile, multi-scale trend information is extracted and aggregated by simple linear layers. Such combination equips DST-Mamba with superior capability to model long-range spatio-temporal dependencies while remaining efficient compared with Transformers. Experimental results across five real-world datasets demonstrate that DST-Mamba can capture both local fluctuations and global trends within traffic patterns, achieving state-of-the-art performance with favorable efficiency.

Introduction

Traffic prediction plays an essential role in modern intelligent transportation systems, with typical applications covering congestion alleviation (Bai et al. 2020) and route planning (Dai et al. 2020). It aims to leverage observed traffic conditions within real road networks to forecast future traffic conditions. Since prediction accuracy highly depends on capturing spatial and temporal dependencies in traffic data, spatio-temporal neural networks that can jointly learn features from time series and graphs have become the mainstream methods in the traffic prediction domain (Wu et al. 2020; Ji, Yu, and Lei 2023; Li et al. 2024).

Recently, spatio-temporal Transformers (Xu et al. 2020; Chen et al. 2022) have received great attention. Unlike traditional spatio-temporal neural networks that extract spatial and temporal features with different deep learning architectures, e.g., graph neural networks (GNNs) (Cao et al. 2024) and recurrent neural networks (RNNs) (Ounoughi and Ben Yahia 2024), spatio-temporal Transformers attempt to maintain a unified model architecture for both spatial and temporal data with only slight differences that reveal data structures. Technically, self-attention provides a global picture for traffic data, which equips Transformers with superior ability in long-term traffic prediction tasks.

However, the difficulty of traffic prediction increases as the spatio-temporal ranges increase (Luo et al. 2024; Cai, Wang, and Hu 2024). Current spatio-temporal Transformers still face several challenges in long-term traffic prediction tasks. First, long-term traffic data exhibits complex and entangled spatio-temporal dependencies that hinder traditional Transformers from extracting perspicacious patterns. Although variants like Autoformer (Wu et al. 2021) and FEDformer (Zhou et al. 2022) attempt to address this problem through decomposition, they only focus on the temporal entanglement, ignoring the effect of spatial restrictions from road networks. Second, Transformers usually require quadratic complexity, which brings computation burdens particularly when both spatial and temporal Transformers are used in traffic prediction. The advancements in linear models have made progress in balancing the efficiency and accuracy for long-range forecasting (Das et al. 2023; Zeng et al. 2023; Wang et al. 2024). Unfortunately, these methods are more suitable for capturing stable trends while failing to perceive the frequent fluctuations in traffic conditions.

To address these challenges, we propose in this paper a decomposed spatio-temporal Mamba for long-term traffic prediction. The main aim of the proposed method is to disentangle the patterns within large spatio-temporal ranges and leverage efficient feature extraction backbones to reduce the computation complexity. Mamba (Gu and Dao 2024) is a selective structured state space model (SSM) that has gained great popularity in many fields. Our core idea is to first decompose spatio-temporal data into trend and seasonal parts from the temporal perspective where a Mamba backbone and a linear model are leveraged to learn from the seasonal and trend parts. Notice that the Mamba framework is

*Corresponding author

designed to model the seasonal part from the spatial perspective where a view shift happens after the decomposition. The Mamba architecture is known to be more efficient in long-term sequence modeling than Transformers. Meanwhile, the remaining trends can be well captured by linear models. Hence, the proposed model can be effective in long-term prediction tasks while maintaining lower computational costs.

To be concrete, the seasonal part contains local and periodic patterns, where the large-scale spatial dependencies from different locations play the dominant role in long-range prediction tasks. Therefore, our model adopts a bi-directional Mamba block to carefully capture these forward and backward spatial correlations from the node perspective. We leverage the graph structure derived from time sequences to formulate the initial input tokens for Mamba so that the spatial information can be incorporated. Since Mamba mainly focuses on cross-node correlations, we additionally design a learnable node embedding for each node to maintain node-independent information. The aggregated node tokens are combined with learnable embeddings for further feature extraction. In contrast, the trend part contains long-term patterns. We propose to utilize linear predictors with multi-scale sampling and mixing to aggregate different scales of trends. Finally, the outputs of these two parallel modules are combined to generate the final prediction results. Notice that the discrepancy of models used for seasonal and trend parts not only depends on their properties but also relies on the difficulties in extracting information from these parts.

To summarize, our contributions lie in three aspects:

- We uncover the entanglement in spatio-temporal traffic data and propose a spatio-temporal decomposition method for long-term traffic prediction. The trend and seasonal inputs are processed with different modules where effectiveness and efficiency can be well balanced.
- We design a Mamba-based architecture to leverage both cross-node and node-independent spatial information. The bi-directional correlations and graph structures are combined to obtain spatio-temporal features and to reduce the sequential order bias.
- The proposed method consistently achieves SOTA performance across five public datasets with favorable efficiency compared to current representative Transformers, linear models, and Mamba-based models.

Related Work

Long-Term Traffic Prediction

Traffic prediction is a successive field of multivariate time series forecasting, which further considers spatial restrictions of traffic networks. The powerful ability of Transformer to depict global dependencies attracts researchers to adapt it to time series representation learning, most of which focus on extending the sequence length while maintaining efficiency (Zhou et al. 2021; Liu et al. 2022). However, the typical use of cross-time attention to extract point-wise temporal relations has recently been questioned to be less effective (Zeng et al. 2023; Liu et al. 2024). Another insight is to shift the view by applying attention along the variable dimension to generate more informative representations (Liu et al. 2024). Besides, as an emerging alternative predictor, linear models (Zeng et al. 2023; Das et al. 2023; Wang et al. 2024; Lin et al. 2024) excel at learning trends based on simpler but more efficient point-wise mappings. However, compared to Transformers, linear models often rely on longer historical observations to make accurate predictions and perform worse in detecting cross-node dependencies.

To strike a better balance between accuracy and efficiency, a novel model called Mamba (Gu and Dao 2024) is proposed, which improves the expressive capability of previous state space models (Gu et al. 2020; Gu, Goel, and Ré 2022) by introducing parameterized matrices and a hardware-aware parallel computing algorithm. Recently, the applications of Mamba in diverse fields have drawn great attention (Zhu et al. 2024; Ma, Li, and Wang 2024; Ahamed and Cheng 2024; Liang et al. 2024). Similar to Transformers, Mamba is also utilized to model either temporal (Zeng et al. 2024; Xu et al. 2024) or spatial dependencies (Wang et al. 2025; Lee et al. 2024) in time series forecasting. The appealing properties of Mamba in sequence modeling inspire us to explore its potential to overcome the challenges faced by current methods of long-term traffic prediction.

Temporal Decomposition

Temporal decomposition deconstructs time series into separate components that exhibit more predictable patterns (Wen et al. 2019; Oreshkin et al. 2020). It is widely used as a pre-processing step or an inner block in Transformer-based and linear models. For instance, Autoformer (Wu et al. 2021) first proposes an inner module for progressively decomposing not only the past sequences but also the intermediate results. Based on Autoformer, FEDformer (Zhou et al. 2022) further combines temporal decomposition with Fourier analysis to better extract global information. TimeMixer (Wang et al. 2024) proposes a fully MLP-based forecaster with Past-Decomposable-Mixing and Future-Multipredictor-Mixing blocks to take advantage of disentangled variations and complementary forecasting capabilities from multi-scale series simultaneously. We notice that these methods only focus on decomposing temporal patterns and ignore the influence of spatial restrictions.

Preliminaries

Problem Formulation

Traffic data collected by multiple sensors is a special multivariate time series under the restrictions of road networks. The traffic series can be denoted as $\mathbf{X}$, and its road network can be represented as a graph $\mathcal{G}$ with an adjacency matrix $\mathbf{S} \in \mathbb{R}^{N \times N}$ where N is the total number of nodes.

Given the input historical observations over the previous L time steps $\mathcal{I} = \{\mathbf{X}_1^t, \mathbf{X}_2^t, \cdots, \mathbf{X}_N^t\}_{t=1}^{L}$ and the prediction horizon T, where $\mathbf{X}_i^t$ represents the traffic state of the i-th node at the t-th time step, the model aims to generate the output future prediction $\mathcal{O} = \{\mathbf{X}_1^t, \mathbf{X}_2^t, \cdots, \mathbf{X}_N^t\}_{t=L+1}^{L+T}$ through a parametrized neural network $\mathcal{F}(\cdot)$.

State Space Models

State Space Models (SSMs) originate from the continuous-time linear time-invariant system and describe the evolution of dynamic systems by latent state transition. This process can be expressed using ordinary differential equations:

$$\begin{aligned} \mathbf{h}'(t) &= \mathbf{A}\mathbf{h}(t) + \mathbf{B}\mathbf{x}(t), \\ \mathbf{y}(t) &= \mathbf{C}\mathbf{h}(t), \end{aligned} \tag{1}$$

where $\mathbf{A}$, $\mathbf{B}$, and $\mathbf{C}$ are learnable matrices. The first equation describes how the current state changes over time under the impact of the input, while the second equation describes how the current state translates to the output.

The continuous SSMs can be discretized by zero-order holding as follows:

$$\begin{aligned} \mathbf{h}_t &= \bar{\mathbf{A}}\mathbf{h}_{t-1} + \bar{\mathbf{B}}\mathbf{x}_t, \\ \mathbf{y}_t &= \mathbf{C}\mathbf{h}_t, \end{aligned} \tag{2}$$

where $\bar{\mathbf{A}} = \exp(\Delta\mathbf{A})$ and $\bar{\mathbf{B}} = (\Delta\mathbf{A})^{-1}(\exp(\Delta\mathbf{A}) - \mathbf{I}) \cdot \Delta\mathbf{B}$. Δ is the step size. The discretized SSMs can be trained in a parallel convolutional manner and make predictions in a recurrent neural network manner.

Mamba (Gu and Dao 2024) introduces a data-dependent selection to parameterize matrices $\mathbf{B}$, $\mathbf{C}$, and Δ. Given an input sequence $\mathbf{x} \in \mathbb{R}^D$, where D is the hidden dimension, Mamba first expands D to ϵD with an expansion factor ϵ by linear projection. The expanded representation passes through convolution operations and a SiLU activation to get $\mathbf{x}' \in \mathbb{R}^{\epsilon D}$. The parameter matrices $\mathbf{A}$, $\mathbf{B}$, $\mathbf{C}$, and Δ are generated and discretized following the aforementioned strategies. Then, these matrices together with $\mathbf{x}'$ are processed by the discretized selective SSM to obtain the state representation $\mathbf{y}$. The final output $\mathbf{y} \in \mathbb{R}^D$ is obtained through a linear transformation with the residual connection.

Methodology

Model Overview

The overall structure of the proposed DST-Mamba is illustrated in Figure 1. Generally, the input traffic series $\mathbf{X}$ is first decomposed into separate trend part $\mathbf{X}_{\text{TR}}$ and seasonal part $\mathbf{X}_{\text{SE}}$, which contain the global trend and local dynamics respectively. These two parts are then processed by the multi-scale linear prediction module and the spatio-temporal Mamba encoder according to their properties.

The multi-scale linear prediction module down-samples $\mathbf{X}_{\text{TR}}$ into m scales through average pooling and obtains a set of trend series. Before being fed into the linear predictor, each trend series is sequentially mixed with its coarser trend to remove the noise. This module then ensembles the outputs from multi-scale trends and obtains $\widetilde{\mathbf{Y}}_{\text{TR}}$. As node interactions are more important for long-term prediction in $\mathbf{X}_{\text{SE}}$, we shift the view and embed the entire series as a node token through efficient linearized graph aggregation. The node tokens are then combined with learnable node embeddings to formulate the input of the bi-directional Mamba block so that the correlations among different nodes can be captured.

Finally, the outputs of these two parts are combined to generate final predictions. The following subsections will describe the main components of DST-Mamba.

Temporal Decomposition Module

Due to the intricate spatio-temporal entanglement in traffic data, it is difficult to effectively capture the hidden patterns from entangled inputs. Specifically, traffic conditions can be viewed as a mixture of seasonal and trend parts that correspond to short-term and long-term dynamic patterns separately. The trend part is less relevant to the node interactions and can be well captured by linear mappings. Meanwhile, the spatial correlations make the traffic states of neighbor nodes influence each other frequently, resulting in dynamic fluctuations. Therefore, instead of directly extracting these spatio-temporal correlations with different properties from entangled traffic series, it is beneficial to decompose the traffic series into separate parts, with each part being processed by suitable model architectures.

To be concrete, we apply temporal decomposition to deconstruct the entangled inputs into seasonal and trend parts. For the traffic series $\mathbf{X} \in \mathbb{R}^{L \times N}$, the moving average is used to highlight the global trend and smooth out the periodic fluctuation. It is implemented as average pooling with the padding method to keep the sequence length unchanged, which can be formulated as:

$$\begin{aligned} \mathbf{X}_{\text{TR}} &= \text{AvgPool}(\text{Padding}(\mathbf{X})), \\ \mathbf{X}_{\text{SE}} &= \mathbf{X} - \mathbf{X}_{\text{TR}}, \end{aligned} \tag{3}$$

where $\mathbf{X}_{\text{TR}}$ and $\mathbf{X}_{\text{SE}}$ denote the extracted trend and seasonal parts with the same shape of $\mathbf{X} \in \mathbb{R}^{L \times N}$.

Multi-Scale Linear Prediction Module

Since linear models have demonstrated great potential in extracting long-term trends from time series, we employ point-wise linear mappings as the basic structure to capture the hidden patterns within $\mathbf{X}_{\text{TR}}$. Based on the observations that traffic series exhibit multiple patterns at different scales, we extend the uni-scale trend extraction to the multi-scale situation by applying down-sampling and scale-mixing.

First, $\mathbf{X}_{\text{TR}}$ is down-sampled to obtain the multi-scale trend series $\{\mathbf{X}_{\text{TR}_0}, \cdots, \mathbf{X}_{\text{TR}_{m-1}}\}$, where m is the number of scales and $\mathbf{X}_{\text{TR}_i} \in \mathbb{R}^{\lfloor L/2^i \rfloor \times N}$. As observed from Figure 1, the series lengths at different scales decrease as the process of down-sampling continues. The lowest level of trend series $\mathbf{X}_{\text{TR}_0}$ is the original input containing the finest information while the highest level of trend series $\mathbf{X}_{\text{TR}_{m-1}}$ reflects the most coarse and macroscopic information.

To eliminate the noise caused by detailed variations in trend series, we apply a top-down mixing strategy. The top-level series with the coarser trend is progressively added to the down-level one with the finer trend so that the information interactions among different scales can be achieved. Specifically, for the multi-scale trend series $\{\mathbf{X}_{\text{TR}_0}, \cdots, \mathbf{X}_{\text{TR}_{m-1}}\}$, the top-down mixing process can be formulated as follows:

$$\mathbf{X}'_{\text{TR}_i} = \mathbf{X}_{\text{TR}_i} + \text{ScaleMix}(\mathbf{X}_{\text{TR}_{(i+1)}}), \tag{4}$$

where $i \in \{0, \cdots, m-1\}$ and $\text{ScaleMix}(\cdot)$ is used to match the temporal dimension between $\mathbf{X}_{\text{TR}_i}$ and $\mathbf{X}_{\text{TR}_{(i+1)}}$, which is implemented by an MLP.

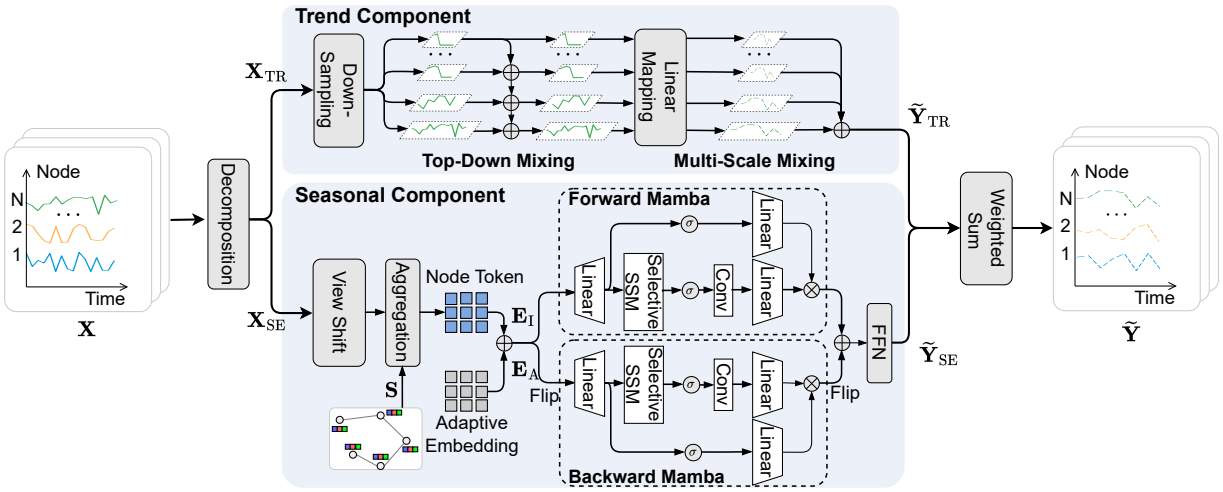

Figure 1: Overall framework of DST-Mamba. After decomposition, the multi-scale linear prediction module aggregates trend information at each scale, while the bi-directional Mamba jointly learns cross-node and node-independent representations from the spatial perspective.

After top-down mixing, scale-wise linear predictors map the mixed trends into corresponding m scales of predictions. Then, these outputs are aggregated to generate overall trend predictions $\widetilde{\mathbf{Y}}_{\text{TR}}$, which empowers DST-Mamba to capture long-range variations within historical traffic series.

Spatio-Temporal Mamba Block

To learn informative spatio-temporal representations from the seasonal part in a node-centric manner, we propose a bi-directional spatio-temporal Mamba block.

Tokenization The seasonal series contains high-frequency information where the spatial proximity among nodes plays a dominant role. Unlike conventional Transformers that capture seasonal patterns from the temporal perspective, we shift our view to the spatial side and adopt the spatial tokenization method to formulate initial inputs. The shift from temporal tokens to spatial tokens empowers DST-Mamba to aggregate global temporal representations within the entire series and explicitly capture spatial dependencies. Concretely, node tokens are obtained through graph aggregation:

$$\mathbf{E}_{\text{I}} = \mathbf{S}\mathbf{X}_{\text{SE}}, \tag{5}$$

where $\mathbf{S}$ is the adjacency matrix and $\mathbf{E}_{\text{I}} \in \mathbb{R}^{N \times D_{\text{I}}}$ is the obtained node tokens, D_{I} is the hidden dimension of the node tokens. The adjacency matrix can be obtained from a predefined graph or by the results of Dynamic Time Warping (DTW) distances. Such aggregation integrates graph structures into tokens to reveal spatial dependencies.

Except for the cross-node interactions, the unique information for each node also impacts traffic patterns. The traffic conditions from distinct regions tend to behave differently, while nearby nodes should be more similar in variations. Thus, we introduce an adaptive spatial embedding $\mathbf{E}_{\text{A}} \in \mathbb{R}^{N \times D_{\text{A}}}$ to jointly consider node-specific patterns. The spatial embeddings are learnable parameters that can be adjusted along with the training process.

Finally, the hidden representation $\mathbf{E} \in \mathbb{R}^{N \times D}$ is obtained by concatenating the token/embedding above, where the dimension D equals to $D_{\text{I}} + D_{\text{A}}$.

Bi-directional Mamba Encoder This module consists of a bi-directional Mamba for modeling node correlations within seasonal variations and a feed-forward network for learning temporal patterns on each node.

The selective mechanism of Mamba is designed to deal with uni-directional sequences with inherent orders. However, the index of spatial nodes does not reflect real orders and the cross-node correlations should be bi-directional. Therefore, directly applying uni-directional Mamba inevitably leads to unsatisfactory results. To mitigate this gap, we extend the uni-directional architecture to a bi-directional one. It can be seen from Figure 1 that the two Mamba pipelines, namely the forward Mamba and backward Mamba, are identical except that they receive nodes from two distinct directions.

For the seasonal series $\mathbf{X}_{\text{SE}} \in \mathbb{R}^{N \times D}$, the sequential order is defined as the forward direction. The reversed input is obtained by flipping its node dimension. Two sequences with different directions are then passed through forward and backward Mamba to generate a global view of spatial correlations. Specifically, each Mamba block contains two branches, where one branch contains the core selective SSM module, serving as the information filter, and another branch is the gate. The process is described as follows:

$$\begin{aligned} \mathbf{H}_{\text{FW}} &= \text{ForwardMamba}(\mathbf{X}_{\text{SE}}), \\ \mathbf{H}_{\text{BW}} &= \text{BackwardMamba}(\text{Flip}(\mathbf{X}_{\text{SE}})), \\ \mathbf{H}_{\text{SE}} &= \mathbf{H}_{\text{FW}} + \text{Flip}(\mathbf{H}_{\text{BW}}), \end{aligned} \tag{6}$$

where $\mathbf{H}_{\text{FW}}$ and $\mathbf{H}_{\text{BW}}$ are features extracted by two Mamba pipelines and $\mathbf{H}_{\text{SE}}$ denotes the learned seasonal patterns.

Feed-Forward Network The encoded seasonal representations processed by Mamba are then fed into a feed-forward network (FFN) to obtain the predictions $\widetilde{\mathbf{Y}}_{\text{SE}}$. For traditional temporal embedding methods, FFN may fail to extract meaningful representations as temporal tokens typically fuse multiple nodes at the same time step and are too localized to provide long-range temporal patterns. However, as each node token inherently contains the entire time series, FFN can implicitly encode temporal dependencies by keeping the dynamic sequential relationships.

Traffic Prediction

After obtaining the outputs from the two components, DST-Mamba generates the final predictions $\widetilde{\mathbf{Y}}$ via

$$\widetilde{\mathbf{Y}} = \widetilde{\mathbf{Y}}_{\text{SE}} + \lambda \widetilde{\mathbf{Y}}_{\text{TR}}, \tag{7}$$

where λ is the weight of trend information. Then, we can construct the training objective for traffic prediction through Mean Squared Error (MSE) loss:

$$\mathcal{L} = |\widetilde{\mathbf{Y}} - \mathbf{Y}|_2^2, \tag{8}$$

where $\mathbf{Y}$ is the ground truth.

Experiments

Datasets and Baselines

To evaluate the performance of DST-Mamba, we carry out experiments on five real-world traffic datasets (Wang et al. 2025), including Traffic and the PEMS datasets. Table 1 gives the detailed statistics of these datasets. Z-Score normalization is applied to each time series to stabilize the training process and accelerate convergence. We adopt the same data processing and dataset split setting in S-Mamba, which strictly follows the chronological order to avoid the data leakage issue.

DST-Mamba is compared with 8 baselines from three categories to demonstrate its effectiveness: (1) Transformers: iTransformer (Liu et al. 2024), Crossformer (Zhang and Yan 2023), PatchTST (Nie et al. 2023), FEDformer (Zhou et al. 2022), and Autoformer (Wu et al. 2021); (2) Linear models: DLinear (Zeng et al. 2023); (3) Mamba models: S-Mamba (Wang et al. 2025) and SOR-Mamba (Lee et al. 2024).

The experiments are conducted on a single NVIDIA GeForce RTX 4090 with 24 GB memory. We use MSE as the loss function and ADAM as the optimizer with an initial learning rate of 10^{-3}. The batch size is set to 32, and the training process is early stopped within 10 epochs. DST-Mamba adopts an encoder-only architecture where the number of encoder layers (bi-directional Mamba block) varies from $\{1, 2, 3, 4\}$. The weight of the trend predictions is selected from the range 0.5 to 1. Our code is available at https://github.com/Anle-He/DST-Mamba.

Datasets	Nodes	Timesteps	Interval
Traffic	862	17,544	1hour
PEMS03	358	26,209	5min
PEMS04	307	16,992	5min
PEMS07	883	28,224	5min
PEMS08	170	17,856	5min

Table 1: The statistics of datasets.

Long-Term Traffic Prediction Results

Following the experimental setting in previous works, we set the input series length to 96 for all datasets and evaluated the models with different prediction horizons. MSE and MAE are selected as the evaluation metrics. The performances of our method and baselines are listed in Table 2.

From the results across different prediction horizons, we can observe that these models that explicitly capture cross-node dependencies (DST-Mamba, SOR-Mamba, S-Mamba, and iTransformer) achieve superior performance compared with traditional cross-time Transformers and linear models. This phenomenon implies that effectively extracting spatial dependencies concealed in traffic data can obtain better results than methods that directly extract temporal dependencies. The results verify the important roles of spatial information in traffic prediction. Compared with Crossformer which jointly captures the cross-time and cross-node dependencies with a two-stage attention module, our model obtains better results, which demonstrates that the backbone to extract spatial dependencies is also important for high prediction accuracy. We also observe that the prediction deviations of linear models increase sharply as the output length grows. Although they require low computational costs, they cannot obtain satisfactory results when facing complex spatio-temporal patterns. This indicates that linear models are not capable of fitting the non-linearity within dynamic spatial correlations.

Compared with other Mamba-based models and iTransformer which extract features from a similar spatial perspective, DST-Mamba achieves better performance on most datasets. According to our analysis, the series decomposition makes the separate sub-series exhibit more obvious and predictable patterns, which improves the effectiveness of cross-node dependency modeling in the bi-directional Mamba block. Meanwhile, the adaptive spatial embedding and the extra trend patterns aggregated from multiple scales can further enhance the learned representations.

Notice that the computational cost of DST-Mamba mainly lies in the bi-directional Mamba module to extract cross-node dependencies. Since DST-Mamba adopts a spatial perspective, the computation complexity is related to the number of nodes instead of the sequence length. Therefore, based on the efficient Mamba block, the complexity is near $\mathcal{O}(N)$, which allows our model to achieve effective long-term traffic prediction with favorable efficiency.

Ablation Analysis

To evaluate the effectiveness of different components in DST-Mamba, we perform detailed ablation studies based on the PEMS08 dataset. Specifically, the following seven variants are included: **w/o AE** removes the adaptive node embeddings. **w/o Dec.** removes the decomposition method in DST-Mamba. **w/o Bi-D** replaces the bi-directional Mamba with a uni-directional Mamba. **w/o Tre.** removes the trend component and keeps the seasonal component. In contrast,

Models		**DST-Mamba (Ours)**		SOR-Mamba (2024)		S-Mamba (2024)		iTransformer (2024)		PatchTST (2023)		DLinear (2023)		Crossformer (2023)		FEDformer (2022)		Autoformer (2021)	
Metric		MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE
PEMS03	12	**0.061**	**0.162**	0.066	0.170	0.065	0.169	0.071	0.174	0.099	0.216	0.122	0.243	0.090	0.203	0.126	0.251	0.272	0.385
	24	**0.077**	**0.184**	0.088	0.197	0.087	0.196	0.093	0.201	0.142	0.259	0.201	0.317	0.121	0.240	0.149	0.275	0.334	0.440
	48	**0.105**	**0.217**	0.134	0.245	0.133	0.243	0.125	0.236	0.211	0.319	0.333	0.425	0.202	0.317	0.227	0.348	1.032	0.782
	96	**0.160**	**0.273**	0.193	0.297	0.201	0.305	0.164	0.275	0.269	0.370	0.457	0.515	0.262	0.367	0.348	0.434	1.031	0.796
	Avg	**0.101**	**0.209**	0.121	0.227	0.122	0.228	0.113	0.221	0.180	0.291	0.278	0.375	0.169	0.281	0.213	0.327	0.667	0.601
PEMS04	12	**0.071**	**0.172**	0.074	0.175	0.076	0.180	0.078	0.183	0.105	0.224	0.148	0.272	0.098	0.218	0.138	0.262	0.424	0.491
	24	**0.083**	**0.188**	0.086	0.192	0.084	0.193	0.095	0.205	0.153	0.275	0.224	0.340	0.131	0.256	0.177	0.293	0.459	0.509
	48	**0.105**	0.218	0.106	**0.214**	0.115	0.224	0.120	0.233	0.229	0.339	0.355	0.437	0.205	0.326	0.270	0.368	0.646	0.610
	96	0.140	0.258	**0.129**	**0.233**	0.137	0.248	0.150	0.262	0.291	0.389	0.452	0.504	0.402	0.457	0.341	0.427	0.912	0.748
	Avg	0.100	0.209	**0.099**	**0.203**	0.103	0.211	0.111	0.221	0.195	0.307	0.295	0.388	0.209	0.314	0.231	0.337	0.610	0.590
PEMS07	12	**0.055**	**0.148**	0.059	0.155	0.063	0.159	0.067	0.165	0.095	0.207	0.115	0.242	0.094	0.200	0.109	0.225	0.199	0.336
	24	**0.070**	**0.165**	0.076	0.174	0.081	0.183	0.088	0.190	0.150	0.262	0.210	0.329	0.139	0.247	0.125	0.244	0.323	0.420
	48	**0.093**	**0.189**	0.098	0.199	**0.093**	0.192	0.110	0.215	0.253	0.340	0.398	0.458	0.311	0.369	0.165	0.288	0.390	0.470
	96	0.126	0.225	**0.117**	0.218	**0.117**	**0.217**	0.139	0.245	0.346	0.404	0.594	0.553	0.396	0.442	0.262	0.376	0.554	0.578
	Avg	**0.086**	**0.182**	0.088	0.186	0.089	0.188	0.101	0.204	0.211	0.303	0.329	0.395	0.235	0.315	0.165	0.283	0.367	0.451
PEMS08	12	**0.069**	**0.167**	0.078	0.178	0.076	0.178	0.079	0.182	0.168	0.232	0.154	0.276	0.165	0.214	0.173	0.273	0.436	0.485
	24	**0.088**	**0.185**	0.103	0.205	0.104	0.209	0.115	0.219	0.224	0.281	0.248	0.353	0.215	0.260	0.210	0.301	0.467	0.502
	48	**0.125**	**0.221**	0.159	0.250	0.167	0.228	0.186	0.235	0.321	0.354	0.440	0.470	0.315	0.355	0.320	0.394	0.966	0.733
	96	**0.194**	**0.263**	0.229	0.295	0.245	0.280	0.221	0.267	0.408	0.417	0.674	0.565	0.377	0.397	0.442	0.465	1.385	0.915
	Avg	**0.119**	**0.209**	0.142	0.232	0.148	0.224	0.150	0.226	0.280	0.321	0.379	0.416	0.268	0.307	0.286	0.358	0.814	0.659
Traffic	96	**0.372**	**0.253**	0.378	0.261	0.382	0.261	0.395	0.268	0.462	0.295	0.650	0.396	0.522	0.290	0.587	0.366	0.613	0.388
	192	**0.388**	**0.259**	0.393	0.269	0.396	0.267	0.417	0.276	0.466	0.296	0.598	0.370	0.530	0.293	0.604	0.373	0.616	0.382
	336	0.401	0.276	**0.399**	**0.272**	0.417	0.276	0.433	0.283	0.482	0.304	0.605	0.373	0.558	0.305	0.621	0.383	0.622	0.337
	720	0.446	0.291	**0.437**	**0.290**	0.460	0.300	0.467	0.302	0.514	0.322	0.645	0.394	0.589	0.328	0.626	0.382	0.660	0.408
	Avg	**0.402**	**0.270**	**0.402**	0.273	0.414	0.276	0.428	0.282	0.481	0.304	0.625	0.383	0.550	0.304	0.610	0.376	0.628	0.379

Table 2: Results of DST-Mamba and baselines on traffic-related datasets. The input length is fixed to 96 and the output lengths are set to 96, 192, 336, and 720 for Traffic and vary from 12 to 96 for PEMS. The best results are marked in bold and the second-best results are marked with underlines.

w / o Sea. removes the seasonal component while keeping the trend component. **w/o Dec.+L.** is the model without decomposition and feeds the original inputs into the spatio-temporal Mamba module. **w/o Dec.+M.** removes the decomposition method and feeds the original inputs into the multi-scale linear prediction module.

Model		DST-Mamba		w/o Dec.		w/o Tre.		w/o Dec.+L.	
Metric		MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE
PEMS08	12	0.069	0.167	0.076	0.176	0.075	0.175	0.077	0.174
	24	0.088	0.185	0.094	0.193	0.097	0.193	0.098	0.198
	48	0.125	0.221	0.141	0.233	0.135	0.225	0.135	0.229
	96	0.194	0.263	0.246	0.284	0.227	0.289	0.256	0.289
	Avg	0.119	0.209	0.139	0.222	0.134	0.221	0.142	0.223
Model		w/o AE		w/o Bi-D		w/o Sea.		w/o Dec.+M.	
Metric		MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE
PEMS08	12	0.073	0.173	0.070	0.169	0.125	0.236	0.124	0.235
	24	0.095	0.196	0.091	0.190	0.238	0.331	0.238	0.331
	48	0.139	0.237	0.131	0.224	0.550	0.531	0.551	0.530
	96	0.213	0.297	0.220	0.275	1.137	0.795	1.130	0.792
	Avg	0.130	0.226	0.128	0.215	0.513	0.473	0.551	0.472

Table 3: Ablation results of seven variants for DST-Mamba on the PEMS08 dataset.

The experimental results of the ablation studies are presented in Table 3. First of all, it can be observed that the decomposition step is essential for DST-Mamba. In the cases where the spatio-temporal patterns are entangled without introducing the decomposition method, the overall performance is generally decreasing. This phenomenon indicates that the entangled traffic series increases the difficulty of learning informative representations despite the fact that Mamba and linear models are kept. In addition, removing the trend or seasonal modeling module will also degrade the overall performance. Removing the seasonal part generates a larger performance degradation compared to removing the trend one. The phenomenon corresponds to our analysis that effectively extracting information from the seasonal part is helpful for long-term traffic prediction. A uni-directional Mamba setting would result in the loss of half semantic information, making it less effective than bi-directional Mamba in capturing global information. The adaptive embeddings are used to maintain node-independent information in addition to the cross-node information captured by Mamba and the graph aggregation process. The results also indicate that inserting node attributes into methods that focus on modeling correlations among traffic series (e.g., Transformers and Mamba) can improve the representation ability.

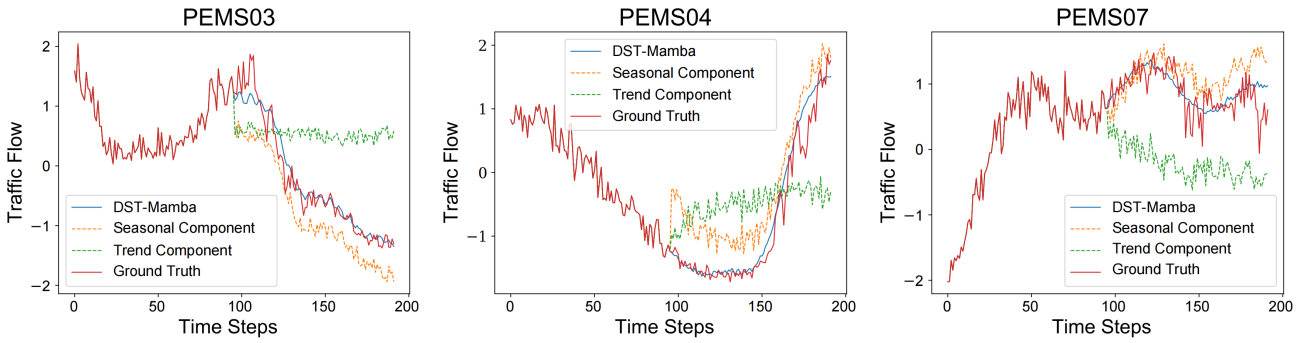

Figure 2: The prediction showcases on the PEMS03, PEMS04 and PEMS07 datasets with output length set to 96. Note that the feature values are normalized during training.

Parameter Sensitivity

In this subsection, we present the influence of four hyper-parameters on the model performance: the weight for the trend output ranging from 0.5 to 1; the down-sampling window size to generate different scales of trend series with the range of $\{2, 3, 4\}$; the number of Mamba blocks with a range from 1 to 4, and the dimension of the adaptive spatial embedding from $\{16, 32, 64, 128\}$. The results on the PEMS08 dataset have been reported in Figure 3.

It is observed that the performance of DST-Mamba with different trend weights is relatively stable, which corresponds to the fact that traffic dynamics rely more on local fluctuations. The results conform to our model design where the seasonal component is modeled by a complex sequential model while the trend component is captured by simple linear mappings. We should also notice that explicitly perceiving trend information provides a global overview and can also improve prediction performance. The down-sampling window size is directly related to the magnitude of the trend information at each scale. When it is too large or too small, the down-sampled sub-series contains more homogeneous patterns, leading to information loss. The spatial embedding dimension reflects the degree of node-specific features. A moderate dimension (i.e., 32) is more effective since it can balance the use of node-independent and node-dependent information. The number of Mamba blocks reveals the capacity of the model, where a small or large number of blocks can impact the prediction accuracy.

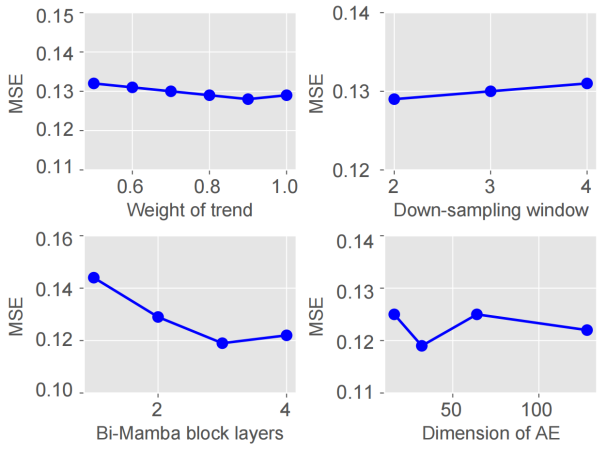

Figure 3: The parameter sensitivity of DST-Mamba on the PEMS08 dataset.

Visualization of Prediction Results

To provide an intuitive demonstration of model performance and the effectiveness of decomposition, we present a set of prediction showcases on the PEMS03, PEMS04, and PEMS07 datasets. As illustrated in Figure 2, DST-Mamba can make precise long-range predictions across various traffic scenarios. Besides, the seasonal outputs made by the spatio-temporal Mamba block can already capture the frequent fluctuations and preserve detailed variations, while the trend outputs supplement the model with stable patterns to adjust the global ranges of final predictions. These observations also correspond to the fact that, compared to other time series data, traffic series exhibit flexible patterns that are closely related to spatial interactions.

Conclusion

In this paper, we developed a spatio-temporal traffic prediction model based on temporal decomposition and Mamba. The proposed model aimed to deconstruct original inputs into trend and seasonal parts, and leverage efficient feature extractors to capture the hidden patterns respectively. In the trend part, linear mappings were leveraged to extract multi-scale information. In the seasonal part, we shifted the view and used a bi-directional Mamba to capture the spatial dependencies. Since Mamba requires less computational burdens in long-term prediction, its combination with linear models exhibited effective and efficient properties. The experimental results indicated that DST-Mamba had low computational costs and achieved leading performance compared with advanced Transformers. Particularly, we designed a bi-directional Mamba model with the help of spatial aggregation and node-independent embedding to effectively handle sequences under spatial networks. Our analyses also demonstrated the validity of these modules and the rationality of spatial modeling in the seasonal part.

Acknowledgments

This work was supported by Beijing Natural Science Foundation under Grant No. 4222020.

References

Ahamed, M. A.; and Cheng, Q. 2024. TimeMachine: A Time Series is Worth 4 Mambas for Long-Term Forecasting. In *European Conference on Artificial Intelligence*.

Bai, L.; Yao, L.; Li, C.; Wang, X.; and Wang, C. 2020. Adaptive Graph Convolutional Recurrent Network for Traffic Forecasting. In *Advances in Neural Information Processing*.

Cai, J.; Wang, C.-H.; and Hu, K. 2024. LCDFormer: Long-Term Correlations Dual-Graph Transformer for Traffic Forecasting. *Expert Systems with Applications*, 249: 123721.

Cao, S.; Wu, L.; Zhang, R.; Wu, D.; Cui, J.; and Chang, Y. 2024. A Spatiotemporal Multiscale Graph Convolutional Network for Traffic Flow Prediction. *IEEE Transactions on Intelligent Transportation Systems*, 25(8): 8705–8718.

Chen, C.; Liu, Y.; Chen, L.; and Zhang, C. 2022. Bidirectional Spatial-Temporal Adaptive Transformer for Urban Traffic Flow Forecasting. *IEEE Transactions on Neural Networks and Learning Systems*, 34(10): 6913–6925.

Dai, R.; Xu, S.; Gu, Q.; Ji, C.; and Liu, K. 2020. Hybrid Spatio-Temporal Graph Convolutional Network: Improving Traffic Prediction with Navigation Data. In *ACM SIGKDD Conference on Knowledge Discovery and Data Mining*.

Das, A.; Kong, W.; Leach, A.; Mathur, S.; Sen, R.; and Yu, R. 2023. Long-Term Forecasting with TiDE: Time-series Dense Encoder. *Transactions on Machine Learning Research*.

Gu, A.; and Dao, T. 2024. Mamba: Linear-Time Sequence Modeling with Selective State Spaces. In *Conference on Language Modeling*.

Gu, A.; Dao, T.; Ermon, S.; Rudra, A.; and Ré, C. 2020. Hippo: Recurrent Memory with Optimal Polynomial Projections. In *Advances in Neural Information Processing*.

Gu, A.; Goel, K.; and Ré, C. 2022. Efficiently Modeling Long Sequences with Structured State Spaces. In *International Conference on Learning Representations*.

Ji, J.; Yu, F.; and Lei, M. 2023. Self-Supervised Spatiotemporal Graph Neural Networks with Self-Distillation for Traffic Prediction. *IEEE Transactions on Intelligent Transportation Systems*, 24(2): 1580–1593.

Lee, S.; Hong, J.; Lee, K.; and Park, T. 2024. Sequential Order-Robust Mamba for Time Series Forecasting. arXiv:2410.23356.

Li, H.; Liu, J.; Han, S.; Zhou, J.; Zhang, T.; and Chen, C. P. 2024. STFGCN: Spatial-Temporal Fusion Graph Convolutional Network for Traffic Prediction. *Expert Systems with Applications*, 255: 124648.

Liang, A.; Jiang, X.; Sun, Y.; and Lu, C. 2024. Bi-Mamba+: Bidirectional Mamba for Time Series Forecasting. arXiv:2404.15772.

Lin, S.; Lin, W.; Wu, W.; Chen, H.; and Yang, J. 2024. SparseTSF: Modeling Long-term Time Series Forecasting with 1k Parameters. In *International Conference on Machine Learning*.

Liu, S.; Yu, H.; Liao, C.; Li, J.; Lin, W.; Liu, A. X.; and Dustdar, S. 2022. Pyraformer: Low-Complexity Pyramidal Attention for Long-Range Time Series Modeling and Forecasting. In *International Conference on Learning Representations*.

Liu, Y.; Hu, T.; Zhang, H.; Wu, H.; Wang, S.; Ma, L.; and Long, M. 2024. iTransformer: Inverted Transformers Are Effective for Time Series Forecasting. In *International Conference on Learning Representations*.

Luo, Q.; He, S.; Han, X.; Wang, Y.; and Li, H. 2024. LSTTN: A Long-Short Term Transformer-based Spatiotemporal Neural Network for Traffic Flow Forecasting. *Knowledge-Based Systems*, 293: 111637.

Ma, J.; Li, F.; and Wang, B. 2024. U-Mamba: Enhancing Long-Range Dependency for Biomedical Image Segmentation. arXiv:2401.04722.

Nie, Y.; Nguyen, N. H.; Sinthong, P.; and Kalagnanam, J. 2023. A Time Series is Worth 64 Words: Long-Term Forecasting with Transformers. In *International Conference on Learning Representations*.

Oreshkin, B. N.; Carpov, D.; Chapados, N.; and Bengio, Y. 2020. N-BEATS: Neural Basis Expansion Analysis for Interpretable Time Series Forecasting. In *International Conference on Learning Representations*.

Ounoughi, C.; and Ben Yahia, S. 2024. Sequence to sequence hybrid Bi-LSTM model for traffic speed prediction. *Expert Systems with Applications*, 236: 121325.

Wang, S.; Wu, H.; Shi, X.; Hu, T.; Luo, H.; Ma, L.; Zhang, J. Y.; and Zhou, J. 2024. TimeMixer: Decomposable Multiscale Mixing for Time Series Forecasting. In *International Conference on Learning Representations*.

Wang, Z.; Kong, F.; Feng, S.; Wang, M.; Yang, X.; Zhao, H.; Wang, D.; and Zhang, Y. 2025. Is Mamba Effective for Time Series Forecasting? *Neurocomputing*, 619: 129178.

Wen, Q.; Gao, J.; Song, X.; Sun, L.; Xu, H.; and Zhu, S. 2019. RobustSTL: A Robust Seasonal-Trend Decomposition Algorithm for Long Time Series. In *AAAI Conference on Artificial Intelligence*.

Wu, H.; Xu, J.; Wang, J.; and Long, M. 2021. Autoformer: Decomposition Transformers with Auto-Correlation for Long-Term Series Forecasting. In *Advances in Neural Information Processing*.

Wu, Z.; Pan, S.; Long, G.; Jiang, J.; Chang, X.; and Zhang, C. 2020. Connecting the Dots: Multivariate Time Series Forecasting with Graph Neural Networks. In *ACM SIGKDD Conference on Knowledge Discovery and Data Mining*.

Xu, M.; Dai, W.; Liu, C.; Gao, X.; Lin, W.; Qi, G.-J.; and Xiong, H. 2020. Spatial-Temporal Transformer Networks for Traffic Flow Forecasting. arXiv:2001.02908.

Xu, X.; Chen, C.; Liang, Y.; Huang, B.; Bai, G.; Zhao, L.; and Shu, K. 2024. SST: Multi-Scale Hybrid Mamba-Transformer Experts for Long-Short Range Time Series Forecasting. arXiv:2404.14757.

Zeng, A.; Chen, M.; Zhang, L.; and Xu, Q. 2023. Are Transformers Effective for Time Series Forecasting? In *AAAI Conference on Artificial Intelligence*.

Zeng, C.; Liu, Z.; Zheng, G.; and Kong, L. 2024. CMamba: Channel Correlation Enhanced State Space Models for Multivariate Time Series Forecasting. arXiv:2406.05316.

Zhang, Y.; and Yan, J. 2023. Crossformer: Transformer Utilizing Cross-Dimension Dependency for Multivariate Time Series Forecasting. In *International Conference on Learning Representations*.

Zhou, H.; Zhang, S.; Peng, J.; Zhang, S.; Li, J.; Xiong, H.; and Zhang, W. 2021. Informer: Beyond Efficient Transformer for Long Sequence Time-Series Forecasting. In *AAAI Conference on Artificial Intelligence*.

Zhou, T.; Ma, Z.; Wen, Q.; Wang, X.; Sun, L.; and Jin, R. 2022. FEDformer: Frequency Enhanced Decomposed Transformer for Long-Term Series Forecasting. In *International Conference on Machine Learning*.

Zhu, L.; Liao, B.; Zhang, Q.; Wang, X.; Liu, W.; and Wang, X. 2024. Vision Mamba: Efficient Visual Representation Learning with Bidirectional State Space Model. In *International Conference on Machine Learning*.