

Article

An Encoder-Enhanced Autoformer for Multi-Scale Parking Availability Forecasting

Chao Zeng¹, Changhao Zhao¹, Xiaoting Huang^{2,*} and Changxi Ma² 

¹ College of Traffic and Transportation, Chongqing Jiaotong University, Chongqing 400074, China; zengchao0602@cqjtu.edu.cn (C.Z.); 622250950134@mails.cqjtu.edu.cn (C.Z.)

² School of Traffic and Transportation, Lanzhou Jiaotong University, Lanzhou 730070, China; machangxi@mail.lzjtu.cn

* Correspondence: 13240004@stu.lzjtu.edu.cn

Abstract

Parking space availability forecasting is challenging due to noise contamination, non-stationary dynamics, and complex multi-scale temporal dependencies. This study proposes a wavelet-enhanced Autoformer with dynamic multi-kernel decomposition for parking space availability forecasting. Discrete wavelet transform is used to enhance the input sequence by suppressing high-frequency disturbances while preserving the dominant temporal structure. Within the Autoformer framework, a dynamic multi-kernel decomposition module is developed by introducing multiple moving-average kernels with different scales and an input-dependent dynamic weighting mechanism, enabling the adaptive decomposition and fusion of trend and fluctuation components and thereby improving the representation of complex multi-scale temporal patterns. The proposed framework is evaluated using real parking operation data collected from the Liangshan Jingyuan parking lot in Chongqing, China, including 1248 observations with a 2 h sampling interval. Multi-step forecasting experiments are conducted under short-, medium-, and long-term prediction settings with different label-length configurations. Across different forecasting horizons and label-length settings, the proposed model consistently outperforms the original Autoformer, PatchTST, and a range of conventional and recent forecasting baselines. In particular, under the 24-step forecasting setting, the proposed method reduces MAE, RMSE, and WMAPE by approximately 10% relative to the original Autoformer. Moreover, the results suggest that the dynamic multi-kernel mechanism improves the model's ability to adaptively capture scale-dependent temporal patterns across different forecasting tasks.

Keywords: parking space availability prediction; self-attention mechanism; Autoformer; multi-scale temporal modeling



Academic Editor: Yao Jia

Received: 6 May 2026

Revised: 9 July 2026

Accepted: 20 July 2026

Published: 23 July 2026

Copyright: © 2026 by the authors.

Licensee MDPI, Basel, Switzerland.

This article is an open access article distributed under the terms and

conditions of the [Creative Commons Attribution \(CC BY\)](https://creativecommons.org/licenses/by/4.0/) license.

1. Introduction

Rapid urbanization and motorization have made parking a critical source of urban transport inefficiency. The shortage and uneven distribution of vacant parking spaces not only increase cruising traffic and travel delay, but also reduce network efficiency and aggravate energy consumption and emissions [1–3]. In this context, accurate forecasting of parking space availability is of direct importance for intelligent parking management, real-time parking guidance, and the efficient use of limited urban parking resources.

Existing studies on parking forecasting have evolved from rule-based and statistical approaches to data-driven learning methods [4–6]. Early methods mainly relied on conventional time-series models, such as ARMA, ARIMA, and SARIMA, to characterize temporal

trends and seasonality in parking data [7–10]. Although these models are interpretable and effective under relatively regular conditions, their predictive capability is often constrained by linear assumptions and stationarity requirements. To address nonlinear relationships, machine learning methods such as SVM, RF, and GBDT have been introduced into parking prediction tasks [11–14]. These methods provide greater modeling flexibility, but their performance remains sensitive to feature engineering and data quality, and they are often inadequate for capturing complex sequential dependence and irregular local fluctuations in parking series.

Deep learning has further advanced parking forecasting by enabling more powerful temporal and spatiotemporal representation learning [15–18]. CNN- and LSTM-based models have been widely used to model local patterns and sequential dependencies [19]. Representative studies include GRU–LSTM models for multi-source parking occupancy prediction [20], CNN–LSTM architectures for extracting temporal and spatial features from parking vacancy data [21], GCN–LSTM frameworks for block-level parking forecasting in large-scale networks [22], and decomposition-assisted neural models such as ICEEMDAN–GRU for reducing sequence complexity before prediction [23]. These studies confirm the potential of deep architectures for parking prediction, but also indicate that prediction accuracy remains vulnerable when parking data are noisy, highly non-stationary, and characterized by heterogeneous temporal patterns across multiple scales.

More recently, self-attention-based forecasting models, including Transformer, Reformer, and Informer, have demonstrated strong capability in long-range dependency modeling [24–26]. Their application has also extended to parking prediction. For example, CNN–Transformer hybrids have been used to integrate local observations, parking intention, and environmental information [27], while Transformer-based spatiotemporal frameworks have been developed to fuse multi-source parking demand data [28]. Recent studies have further extended parking prediction toward graph-enhanced and spatiotemporal Transformer architectures. Zhou et al. proposed a multi-graph convolutional Transformer for multi-step parking demand prediction, in which graph-based spatial feature extraction is combined with Transformer-based temporal modeling [15]. Li et al. developed a semantics-aware spatial–temporal dynamic graph Transformer for on-street parking occupancy prediction, highlighting the importance of semantic correlations and dynamic spatial interactions among parking locations [29]. Chen et al. further proposed a multi-view spatial–temporal graph framework for parking availability prediction, showing that multi-view graph representations can improve long-term occupancy forecasting [30]. Although these approaches significantly improve network-level parking prediction, they are mainly designed for multi-location or graph-structured scenarios with explicit spatial dependencies or multi-source external inputs.

At the same time, decomposition-based forecasting models have provided important insights for modeling complex temporal structures. For instance, FEDformer enhances long-term forecasting through frequency-domain decomposition [31], while DLinear adopts a simple moving-average decomposition strategy to separate trend and seasonal components [32]. These methods improve interpretability and forecasting stability to some extent, but their decomposition operations remain fixed or predefined, making it difficult to adapt to input-dependent temporal variations. In addition, recent Transformer variants such as PatchTST [33] and iTransformer [34] further improve sequence representation through patch-level tokenization and variable-wise attention mechanisms. However, these models mainly strengthen feature representation or attention structure, rather than explicitly modeling adaptive multi-scale decomposition. As a result, their ability to characterize heterogeneous temporal structures under highly non-stationary conditions remains limited.

To better position the present study within the recent literature, Table 1 presents a comparative review of representative studies on parking forecasting as well as decomposition-based and Transformer-based time-series forecasting methods.

Table 1. Comparative review of representative recent studies and the present study.

Reference	Model	Main Contribution	Research Gap
Zhou et al. [15]	MGC-Transformer	Multi-graph spatial–temporal parking prediction	Requires graph-structured multi-location data
Li et al. [29]	SaSFormer	Semantic-aware dynamic spatial modeling	Focuses mainly on spatial interaction learning
Chen et al. [30]	MV-STGCN	Multi-view spatial–temporal graph modeling	Depends on external spatial views
Zhou et al. [31]	FEDformer	Frequency-enhanced decomposition forecasting	Not designed for parking-specific adaptive temporal-scale modeling
Zeng et al. [32]	DLinear	Simple and effective decomposition forecasting	Uses fixed decomposition structure
Nie et al. [33]	PatchTST	Patch-based long-sequence representation	Does not explicitly address denoising or adaptive decomposition
Liu et al. [34]	iTransformer	Inverted Transformer for series modeling	Lacks explicit adaptive temporal decomposition
This study	Wavelet-enhanced Autoformer with dynamic multi-kernel decomposition	DWT denoising and adaptive multi-scale temporal modeling	Addresses noisy inputs and heterogeneous temporal scales in single-site parking forecasting

As shown in Table 1, recent studies have made notable progress in parking forecasting and time-series prediction. Nevertheless, several challenges remain insufficiently addressed in parking space availability forecasting. In particular, raw parking series are often affected by noise and short-term disturbances; parking availability exhibits non-stationary temporal dynamics across different operating conditions and prediction horizons, and the series typically contain heterogeneous multi-scale patterns in which long-term trends and short-term fluctuations interact. Existing methods, including decomposition-based and Transformer-based models, still have limited ability to jointly address input noise suppression and adaptive multi-scale temporal representation within a unified forecasting framework.

To address these issues, this study proposes an improved Autoformer framework for parking space availability forecasting. DWT-based denoising is used to improve input signal quality by suppressing high-frequency disturbances while preserving the dominant temporal structure, and the fixed single-kernel decomposition in the encoder of Autoformer is replaced by a dynamic multi-kernel decomposition mechanism with adaptive scale weighting and multi-scale feature fusion. In this way, the proposed framework differs from existing decomposition-enhanced Transformer architectures by jointly considering input-level noise reduction and encoder-side adaptive multi-scale decomposition for parking forecasting. Extensive experiments show that the proposed method achieves superior predictive performance under different forecasting horizons and label-length settings, demonstrating its potential for intelligent parking management applications.

2. Methods

2.1. Wavelet Transform

Parking availability series are inherently nonlinear, non-stationary, and multi-scale, as they are jointly affected by periodic travel demand, parking turnover, random disturbances, and sensing noise. Directly modeling the raw sequence may therefore obscure its intrinsic temporal structure and weaken forecasting performance. To address this issue, this study introduces the discrete wavelet transform (DWT) to decompose the original series into sub-signals with distinct frequency characteristics.

Unlike Fourier-based analysis, which mainly provides global frequency information, wavelet transform enables localized time-frequency analysis and is therefore more suitable for non-stationary sequence modeling [35]. By projecting the signal onto wavelet bases at different scales, DWT separates smooth low-frequency structures from high-frequency transient variations, thereby providing a more informative representation of parking dynamics.

The continuous wavelet transform of a signal $f(t)$ is defined as

$$W_{\varphi}f(a, b) = \frac{1}{\sqrt{|a|}} \int_{\mathbb{R}} f(t) \overline{\psi\left(\frac{t-b}{a}\right)} dt \quad (1)$$

where $\bar{\psi}$ is the complex conjugate of the mother wavelet ψ , a denotes the scale parameter, and b denotes the translation parameter. The wavelet basis satisfies $\int_{-\infty}^{+\infty} \psi(t) dt = 0$. For discrete time-series applications, wavelet analysis is implemented in discretized form. Let $a = a_0^j$ and $b = kb_0a_0^j$, where $a_0 > 1$, $b_0 \in \mathbb{R}$, and j, k are integers. The DWT can then be expressed as

$$W_{\varphi}f(j, k) = 2^{-j/2} \int_{\mathbb{R}} f(t) \overline{\psi(2^{-j}t - k)} dt \quad (2)$$

In this study, the Daubechies 4 (db4) wavelet is selected as the mother wavelet, and the decomposition level is set to four. The db4 wavelet is adopted because it provides a good balance between localization ability and smooth reconstruction, which is suitable for parking availability series containing both gradual structural evolution and short-term fluctuations. Compared with simpler wavelets such as db1, db4 is smoother and better suited to representing continuous temporal variation, whereas wavelets with higher order may lead to stronger smoothing effects and reduced sensitivity to local changes. In addition, the four-level decomposition is used to achieve a practical balance between multi-scale representation and information preservation, so that short-term fluctuations, intermediate variations, and long-term trend information can be separated more effectively.

Applying DWT to the parking availability sequence therefore yields low-frequency and high-frequency components at multiple scales. The low-frequency component mainly retains the dominant trend and structural evolution of the series, whereas the high-frequency components capture short-term fluctuations, abrupt variations, and noise-like disturbances. Such multi-resolution decomposition provides a more robust basis for subsequent denoising and forecasting.

2.2. Mallat Wavelet Decomposition and Reconstruction

In this study, DWT is implemented via the Mallat algorithm, which provides an efficient multi-resolution decomposition through cascaded low-pass and high-pass filtering. Owing to its recursive structure and low computational cost, the Mallat algorithm is well suited to time-series decomposition tasks.

At each decomposition level, the input sequence is filtered and down-sampled to generate approximation and detail coefficients. The decomposition process is given by

$$\begin{cases} c_k^j = \sum_n h(n-2k)c_n^{j-1} \\ d_k^j = \sum_n g(n-2k)c_n^{j-1} \end{cases} \quad (3)$$

where $h(n)$ and $g(n)$ denote the low-pass and high-pass filters, respectively, and c_0 is the original sequence. Here, c_k^j represents the approximation coefficient at level j , which characterizes the low-frequency trend, whereas d_k^j represents the detail coefficient at level j , which reflects high-frequency variations.

The corresponding reconstruction process is defined as

$$c_n^{j-1} = \sum_k h(n-2k)c_k^j + \sum_k g(n-2k)d_k^j \quad (4)$$

Through recursive decomposition, the original parking availability sequence is represented as one approximation component and multiple detail components across scales. The approximation component describes the slowly varying occupancy pattern, while the detail components reflect short-term perturbations induced by stochastic parking behavior and measurement noise. This decomposition enables the forecasting model to capture heterogeneous temporal dependencies more effectively and improves its ability to model complex parking availability dynamics. The detailed decomposition and reconstruction procedure is illustrated in Figure 1. The arrows indicate the direction of signal flow in the recursive wavelet decomposition. At each decomposition level, the input is passed through the high-pass filter g and the low-pass filter h , followed by downsampling by a factor of 2, to obtain the detail component D_i and the approximation component A_i , respectively. The approximation component A_i is then recursively decomposed at the next level.

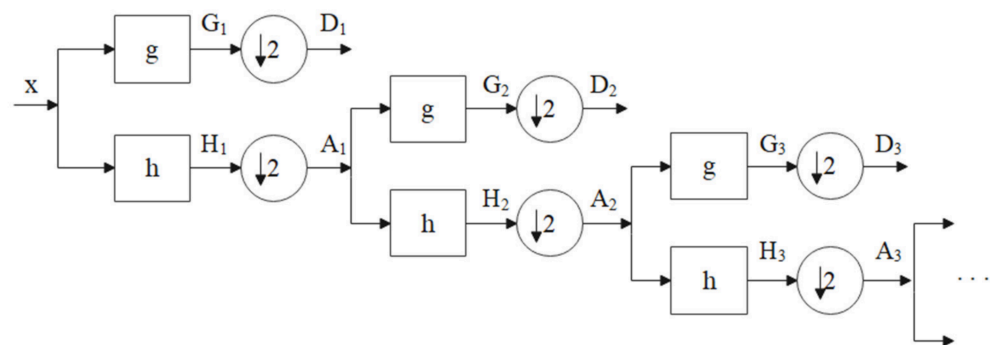


Figure 1. Mallat algorithm framework.

2.3. Autoformer with Dynamic Multi-Kernel Decomposition

2.3.1. Autoformer Backbone

Autoformer is an encoder–decoder forecasting architecture designed for long-sequence time-series modeling. Its effectiveness mainly stems from two key mechanisms: sequence decomposition and autocorrelation-based dependency learning. Instead of directly modeling the raw sequence as a whole, Autoformer first separates the input into trend and seasonal components, and then captures long-range temporal dependencies through the autocorrelation mechanism [36,37].

For an input sequence x , the original Autoformer uses a moving-average operator to extract the trend component:

$$x_t = AvgPool(Padding(x)) \quad (5)$$

where x_t denotes the trend component. $Padding(\cdot)$ denotes the boundary padding operation used to preserve the sequence length, $AvgPool(\cdot)$ denotes the moving-average operator.

The corresponding seasonal component is defined as

$$x_s = x - x_t \quad (6)$$

where x_s denotes the seasonal component. Based on this decomposition, the encoder mainly learns temporal and long-range dependencies from the decomposed sequence, while the decoder progressively refines the trend and seasonal information to generate future predictions.

Although this decomposition mechanism is simple and effective, it relies on a single fixed-size moving-average kernel. Such a setting assumes that the dominant temporal structure of the input sequence can be adequately captured at one predefined scale. However, parking availability series often exhibit heterogeneous dynamics across multiple temporal scales, including slowly varying occupancy trends, periodic fluctuations, and abrupt short-term disturbances. Under such conditions, a fixed-kernel decomposition may not fully separate trend and seasonal information, thereby limiting the modeling capability of Autoformer.

2.3.2. Dynamic Multi-Kernel Decomposition Module

To improve the adaptability of sequence decomposition, this study introduces a dynamic multi-kernel decomposition module into Autoformer. The key idea is to replace the original single-kernel moving-average operator with a set of moving-average kernels of different sizes, and then dynamically aggregate the resulting trend estimates according to the characteristics of the input sequence.

Given an input sequence $x \in R^{L \times d}$, multiple trend candidates are first generated using moving-average operators with different kernel sizes. For the m -th kernel with size k_m , the corresponding trend candidate is defined as

$$\tilde{x}_t^{(m)} = AvgPool_{k_m}(Padding(x)), m = 1, 2, \dots, M \quad (7)$$

where M is the number of kernels, and $\tilde{x}_t^{(m)}$ denotes the trend estimate at the m -th temporal scale. A smaller kernel emphasizes short-term variations, whereas a larger kernel captures smoother and longer-term evolution patterns.

The associated seasonal candidate is correspondingly given by

$$\tilde{x}_s^{(m)} = x - \tilde{x}_t^{(m)} \quad (8)$$

where $\tilde{x}_s^{(m)}$ denotes the seasonal candidate associated with the m -th kernel, obtained by subtracting the corresponding trend candidate from the input sequence. Thus, multiple trend and seasonal candidates are generated in parallel at different temporal scales.

To enable input-adaptive scale selection, a dynamic weighting mechanism is further introduced. Specifically, the hidden representation is first summarized along the temporal dimension by mean pooling:

$$\tilde{x} = \frac{1}{L} \sum_{l=1}^L x_l \quad (9)$$

where L denotes the sequence length, x_l denotes the hidden representation at the l -th time step, and $\bar{x} \in R^d$ is the pooled representation of the input sample. This pooled vector is then passed through a two-layer multilayer perceptron with GELU activation to produce scale-specific logits:

$$z = f(\bar{x}) = W_2 GELU(W_1 \bar{x} + b_1) + b_2 \quad (10)$$

where W_1 and W_2 are learnable weight matrices, b_1 and b_2 are learnable bias vectors, $f(\cdot)$ denotes the two-layer multilayer perceptron, and $z = [z_1, z_2, \dots, z_M]$ denotes the unnormalized importance scores of the kernels. The corresponding normalized weights are computed by

$$\alpha_m = \frac{\exp(z_m)}{\sum_{r=1}^M \exp(z_r)}, m = 1, 2, \dots, M \quad (11)$$

where α_m is the adaptive weight assigned to the m -th kernel and satisfies $\sum_{m=1}^M \alpha_m = 1$.

Using these weights, the final dynamic trend component is obtained by weighted fusion of the multi-scale trend candidates:

$$x_t^{dyn} = \sum_{m=1}^M \alpha_m \tilde{x}_t^{(m)} \quad (12)$$

where x_t^{dyn} denotes the final dynamic trend component obtained by weighted aggregation of the multi-scale trend candidates.

Accordingly, the dynamic seasonal component is defined as

$$x_s^{dyn} = x - x_t^{dyn} \quad (13)$$

where x_s^{dyn} denotes the final dynamic seasonal component corresponding to the dynamically aggregated decomposition result. Dynamic means that the weights assigned to different kernels are adaptively generated from the current input representation, rather than being predefined and fixed. Compared with the original single-kernel decomposition, the proposed module enables the model to adaptively integrate temporal information across different receptive fields. This is particularly important for parking availability prediction, where the dominant temporal pattern may vary with parking facility type, time period, demand intensity, and prediction horizon. By introducing dynamic multi-scale decomposition, the model becomes more capable of distinguishing long-term structural evolution from local volatility.

2.3.3. Integration into the Autoformer Framework

The proposed dynamic multi-kernel decomposition module is introduced only into the encoder of Autoformer, while the decoder retains the original single-kernel decomposition strategy. The encoder is mainly responsible for extracting hierarchical representations from the observed input sequence, where multi-scale temporal structures and latent trend-seasonal interactions have not yet been explicitly separated. At this stage, using only a fixed decomposition kernel may be insufficient, because a single kernel corresponds to only one smoothing scale and may not adequately capture the heterogeneous temporal dynamics of parking availability series. A relatively small kernel is more sensitive to short-term fluctuations, whereas a larger kernel is more suitable for long-term smoothing. Therefore, fixed-kernel decomposition may lead to scale mismatch when different temporal patterns coexist in the same sequence [38,39].

To address this issue, adaptive multi-kernel decomposition is introduced at the encoder stage, enabling the model to dynamically adjust the contribution of different temporal scales and to learn more flexible representations of non-stationary patterns. The key idea is to use multiple kernels to provide complementary smoothing effects at different temporal scales, so that both local fluctuations and broader temporal evolution can be represented more adequately. In this way, the decomposition process is no longer restricted to a single predefined scale.

In each encoder layer, the input representation is first processed by the autocorrelation mechanism to capture long-range dependencies. The resulting hidden states are then

decomposed by a dynamic multi-kernel module, which adaptively extracts trend-related information while preserving seasonal representations across multiple temporal scales. The dynamic weighting mechanism generates sample-dependent normalized weights from the current input representation, so that the contribution of each kernel is determined adaptively rather than fixed in advance. After the feed-forward transformation, a second dynamic decomposition step is applied to further refine the representation of multi-scale temporal dynamics throughout the encoding process.

Let x^{l-1} denote the input of the l -th encoder layer. The encoder update process can be summarized as

$$\tilde{x}_1^l = x^{l-1} + AC(x^{l-1}) \quad (14)$$

$$x_{s,1}^l, x_{t,1}^l = DD_1(\tilde{x}_1^l) \quad (15)$$

$$\tilde{x}_2^l = x_{s,1}^l + FFN(x_{s,1}^l) \quad (16)$$

$$x_{s,2}^l, x_{t,2}^l = DD_2(\tilde{x}_2^l) \quad (17)$$

where l denotes the encoder-layer index; $AC(\cdot)$ and $FFN(\cdot)$ denote autocorrelation and feed-forward transformation, respectively; $DD_1(\cdot)$ and $DD_2(\cdot)$ denote the first and second dynamic decomposition operations in the encoder layer; $x_{s,1}^l$ and $x_{t,1}^l$ denote the seasonal and trend components after the first decomposition; and $x_{s,2}^l$ and $x_{t,2}^l$ denote the corresponding components after the second decomposition.

By contrast, the decoder mainly performs iterative sequence generation based on the representations learned by the encoder. At this stage, retaining the original single-kernel decomposition is expected to help maintain a stable trend-seasonal refinement process while avoiding additional structural complexity in the generative stage. Therefore, the proposed framework applies dynamic multi-kernel decomposition only to the encoder, so as to enhance representation learning while preserving the original simplicity of the decoder design.

2.3.4. Input Representation and Feature Embedding

To preserve the numerical characteristics and temporal semantics of parking availability data, the model adopts a combined embedding strategy. Specifically, the raw parking availability sequence is first mapped through a value embedding layer to represent the magnitude variation in the signal. Meanwhile, temporal information associated with the sequence, such as hour-of-day, day-of-week, and other calendar-related attributes, is encoded through a time embedding layer. The final input representation is formulated as

$$X_i = E_i(X) + T_i(X) \quad (18)$$

where X denotes the original input sequence, i denotes the temporal index, $E_i(X)$ denotes the value embedding at the i -th time step, $T_i(X)$ denotes the corresponding time embedding, and X_i denotes the final input representation.

This embedding strategy allows the model to jointly exploit numerical evolution patterns and temporal context information. When combined with the proposed dynamic multi-kernel decomposition module, the embedded sequence can provide richer and more discriminative features for downstream forecasting.

3. Prediction Model Framework

This study develops a parking availability prediction model that combines DWT-based denoising and reconstruction with an encoder-enhanced Autoformer equipped with a dynamic multi-kernel decomposition mechanism. The overall framework is illustrated in Figure 2.

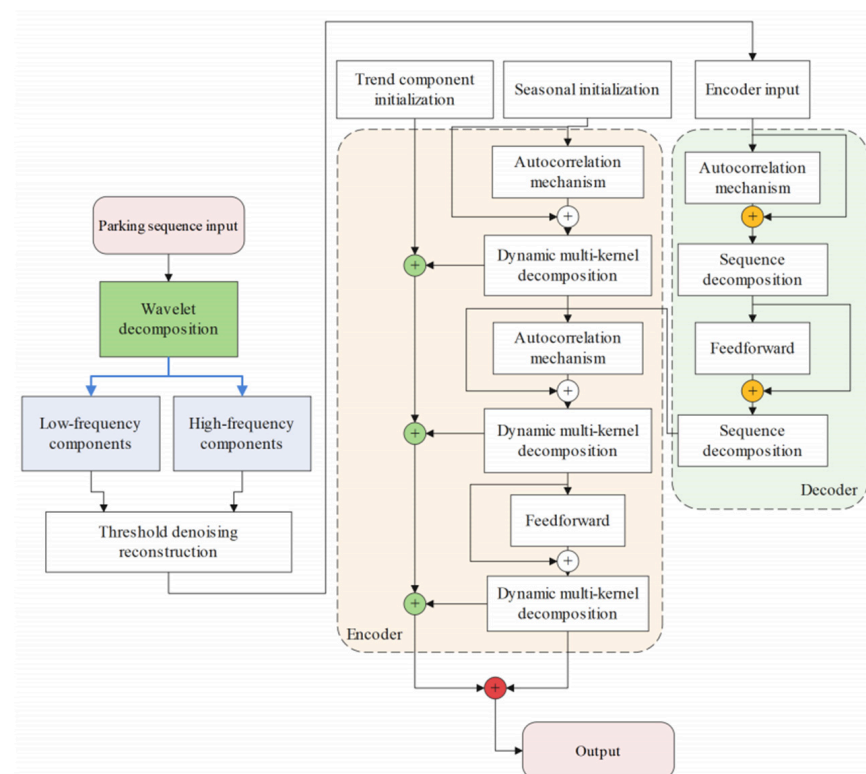


Figure 2. The framework of the prediction model.

The colored addition nodes denote element-wise summation. The white and yellow nodes represent residual additions in the encoder and decoder, respectively; the green nodes accumulate the extracted trend components; and the red node combines the seasonal and trend outputs.

The prediction procedure can be summarized as follows.

Step 1: The original parking availability sequence is collected and preprocessed to handle missing values and abnormal observations.

Step 2: Before the subsequent forecasting process, the preprocessed sequence is first decomposed by DWT into low-frequency and high-frequency components. The preprocessed sequence is decomposed by DWT into low-frequency and high-frequency components. Soft-threshold denoising is then applied to the high-frequency detail coefficients, while the low-frequency approximation component is retained. After signal reconstruction, a cleaner sequence is obtained as the input for subsequent forecasting, which helps suppress noise-like disturbances while preserving the dominant temporal structure.

Step 3: The reconstructed sequence is mapped into the hidden space through value embedding and temporal embedding. The embedded representation is then fed into the encoder, where the autocorrelation mechanism captures long-range periodic dependencies. Meanwhile, the original single-kernel decomposition is replaced by dynamic multi-kernel decomposition modules, which adaptively aggregate multi-scale trend estimates and preserve the corresponding seasonal representations for subsequent encoding.

Step 4: The decoder retains the original Autoformer structure. Based on the initialized seasonal and trend components, it progressively refines the hidden representations and generates the final prediction of future parking availability.

Step 5: The predicted values are compared with the observed values, and the model parameters are updated through backpropagation using the predefined loss function.

4. Case Study

4.1. Data Source

The dataset used in this study was collected from the Liangshan Jingyuan parking lot in Chongqing, China, and represents real parking operation data. It covers the period from 3 January 2023 to 29 December 2023, with a sampling interval of 2 h, yielding 1248 observations in total. The raw records include vehicle entries, vehicle exits, total vehicle counts, and parking utilization information. During data quality inspection, no abnormal values were identified, and a small number of missing observations were filled by linear interpolation to preserve the temporal continuity of the series. Based on these records, the parking availability series was constructed as the prediction target of this study.

A sample of the raw data is presented in Table 2, and the temporal variation in the original parking availability series is illustrated in Figure 3. As shown in the figure, the sequence exhibits evident periodicity at the macro level, while also containing substantial local fluctuations and irregular disturbances. Such characteristics indicate that the parking availability series is not only structured but also noisy.

Table 2. Sample data from Liangshan Jingyuan parking lot.

Time	2:00	4:00	6:00	8:00	10:00	12:00	14:00	16:00	18:00	20:00	22:00	0:00
Vehicle Entries	10	0	1	7	52	44	28	27	89	61	72	14
Vehicle Exits	2	1	0	81	94	37	40	27	47	51	24	5
Total Vehicles	248	247	248	174	132	139	127	127	169	179	227	236
Parking Utilization Rate (%)	88.76	89.49	89.86	63.04	47.83	50.36	46.01	46.01	61.23	64.86	82.25	85.51

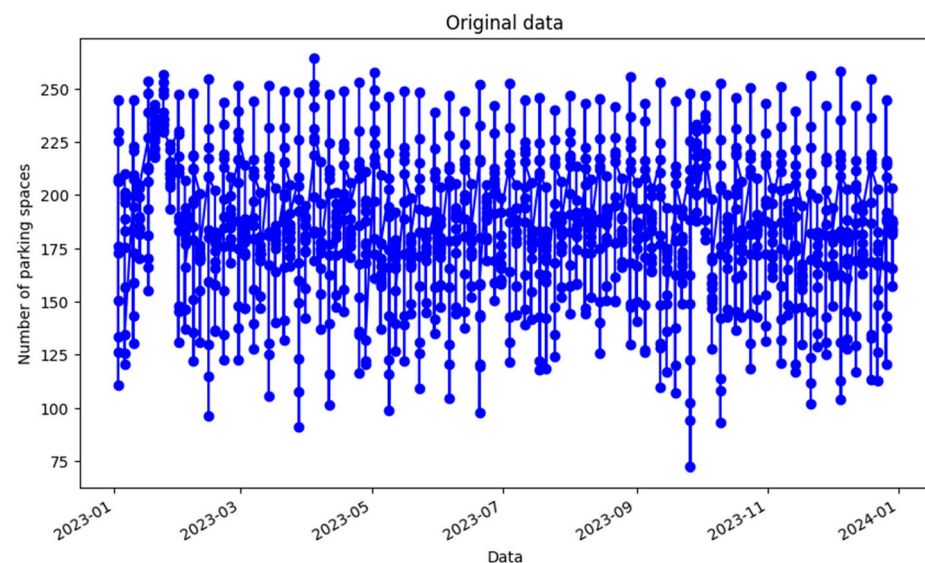


Figure 3. Original parking availability series before denoising.

4.2. Data Preprocessing

To improve the input quality for subsequent forecasting, wavelet-based denoising was performed during data preprocessing. Since the raw parking availability series contains short-term fluctuations and noise-like disturbances, directly feeding the original sequence into the forecasting model may obscure its intrinsic temporal structure and degrade prediction accuracy.

Following the wavelet configuration introduced in Section 2.1, the original parking availability series was decomposed using the db4 wavelet with a four-level decomposition,

as shown in Figure 4. Based on the decomposed coefficients, soft-threshold denoising was applied to the high-frequency detail components, while the low-frequency approximation component was retained. Specifically, for each processed detail-coefficient level, the threshold was determined as

$$T_r = \sqrt{2 \log_2(N)} \quad (19)$$

where N denotes the length of the corresponding detail-coefficient sequence. Let d_j denote a wavelet detail coefficient. The soft-thresholding rule is given by

$$\tilde{d}_j = \begin{cases} \text{sgn}(d_j)(|d_j| - T_r), & |d_j| \geq T_r \\ 0, & |d_j| < T_r \end{cases} \quad (20)$$

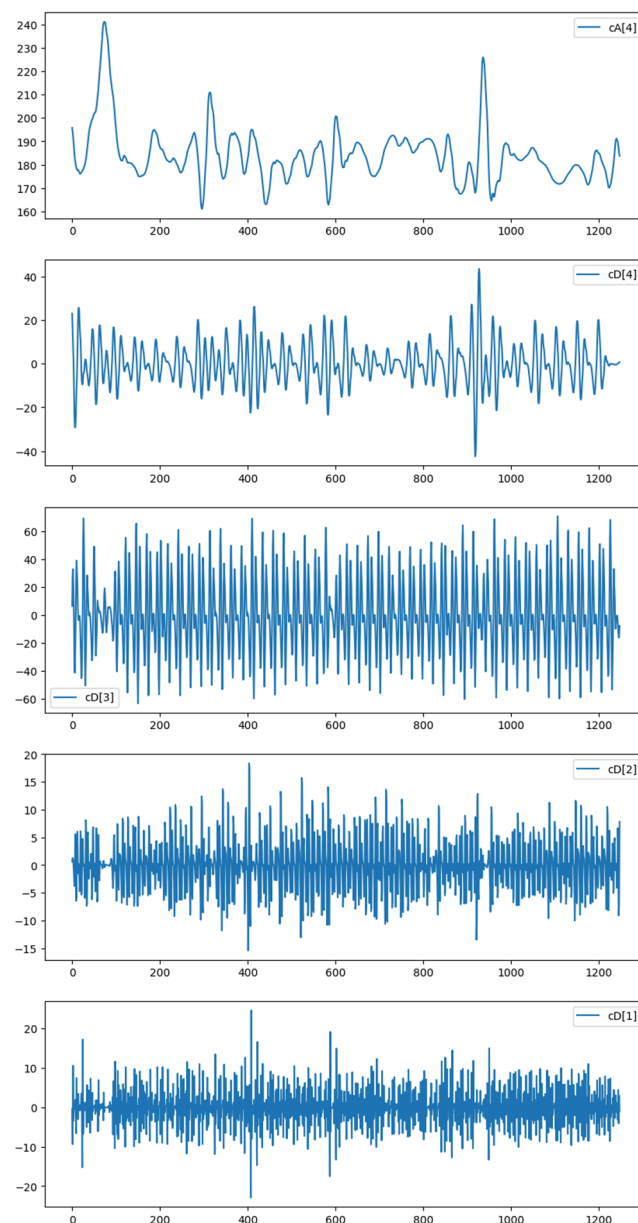


Figure 4. Wavelet decomposition of the original parking availability series.

Coefficients with small magnitudes were set to zero, whereas larger coefficients were shrunk toward zero. After denoising, the signal was reconstructed by inverse wavelet transform and used as the input to the subsequent forecasting model.

The denoising process preserves the dominant temporal structure of the original sequence while suppressing irregular high-frequency disturbances. The comparison between the original and denoised parking availability series is shown in Figure 5. It can be observed that the reconstructed sequence remains consistent with the overall temporal pattern of the raw data, while exhibiting reduced local noise and smoother variation, which provides a more robust input for downstream forecasting.

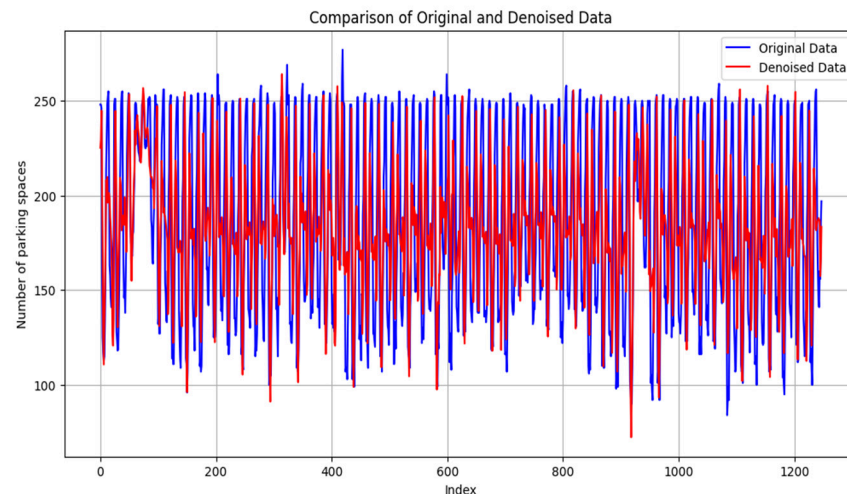


Figure 5. Comparison between the original and denoised parking availability series.

4.3. Experimental Results

All experiments were implemented in Python on a Windows 11 64-bit platform. The hardware environment consisted of a 12th Gen Intel(R) Core(TM) i5-12600K CPU and an NVIDIA GeForce RTX 3070 Ti GPU. The software environment included Python 3.12.7, PyTorch 2.4.1 with CUDA 12.1, NumPy 1.26.4, Pandas 2.2.2, and scikit-learn 1.5.1.

Before model training, forecasting samples were first generated from the reconstructed parking availability series using a sliding-window scheme. Under the current setting, the resulting sample set was then divided into training and test subsets with a ratio of 80% and 20%, respectively. A random seed was set for the sample partition procedure. StandardScaler was adopted for normalization. All reported evaluation metrics were computed after inverse transformation to the original scale. To verify the applicability and stability of the proposed DWT-Autoformer model under different forecasting horizons, three prediction scenarios were designed in this study, corresponding to short-, medium-, and long-term forecasting tasks, respectively. All three experiments used the denoised and reconstructed parking availability series as input. Except for the input length, label length, and prediction length, the remaining model structure and training parameters were kept unchanged.

Specifically, the hidden dimension of the model was set to 256, the feed-forward dimension was set to 1024, and the number of attention heads was set to 8. The number of encoder and decoder layers was set to 2 and 1, respectively. The moving-average kernel sizes of the dynamic multi-kernel decomposition module on the encoder side were set to 7, 25, and 49, and the hidden dimension of the dynamic weighting network was set to 32. The single-kernel moving-average size in the decoder was set to 25. During training, the Adam optimizer was adopted, the learning rate was set to 1×10^{-4} , the loss function was MSE, the batch size was set to 32, and the maximum number of training epochs was set to 10.

The settings of the three forecasting scenarios are shown in Table 3. To ensure comparability across different forecasting tasks, the input length was fixed at 96, while only the combinations of label length and prediction length were adjusted. Among them, the 24-step

forecasting scenario adopted a relatively long label length to enhance trend initialization in short-term forecasting; the 48-step forecasting scenario used a shorter label length to examine the model performance under a weaker decoder prior; and the 96-step forecasting scenario adopted a longer label length together with a longer forecasting horizon to evaluate the model's multi-scale modeling capability in long-term prediction.

Table 3. Forecasting horizon settings.

Forecasting Scenario	Input Length	Label Length	Prediction Length
Scenario 1 (short-term forecasting)	96	48	24
Scenario 2 (medium-term forecasting)	96	12	48
Scenario 3 (long-term forecasting)	96	48	96

Table 4 reports the test results of the three forecasting scenarios in the form of mean $\pm$ standard deviation over 10 independent runs. Overall, the proposed model maintains high predictive accuracy and good stability under different forecasting horizons, with MAE values all below 6, WMAPE values all below 3.2%, and R^2 values all above 0.94. These results indicate that the combination of DWT-based denoising and encoder-side dynamic multi-kernel decomposition can effectively adapt to the multi-scale temporal characteristics of parking availability series.

Table 4. Prediction results under different forecasting horizons.

Forecasting Scenario	MAE	RMSE	WMAPE (%)	R^2
24-step forecasting	5.5914 ± 0.3496	7.4130 ± 0.5842	3.0644 ± 0.1915	0.9414 ± 0.0095
48-step forecasting	5.6885 ± 0.5118	7.5204 ± 0.4370	3.1171 ± 0.2803	0.9430 ± 0.0063
96-step forecasting	5.2726 ± 0.2611	6.9271 ± 0.1628	2.8901 ± 0.1431	0.9515 ± 0.0023

As can be seen from Table 4, the 96-step forecasting scenario achieved the best overall performance, with its MAE, RMSE, and WMAPE all being the lowest among the three experiments. This indicates that, although the forecasting horizon increases, the proposed model is still able to maintain strong long-term forecasting capability. By contrast, the results of the 24-step and 48-step forecasting scenarios are relatively close, with the 48-step forecasting scenario showing slightly higher errors than the 24-step forecasting scenario. This result suggests that forecasting performance does not simply decrease monotonically as the forecasting horizon increases, but is also closely related to the periodic structural characteristics of the series itself and the label length setting on the decoder side. For the parking availability series considered in this study, periodic patterns at longer temporal scales are relatively pronounced, which provides favorable conditions for the model to maintain high accuracy in long-horizon forecasting.

Taking the 24-step forecasting setting as an example, the proposed framework also shows good practical applicability in terms of computational efficiency, real-time inference, and scalability. On the CUDA 12.1-based experimental platform with an NVIDIA GeForce RTX 3070 Ti GPU, the total training time was 9.0783 s, with an average of 0.9078 s per epoch. The total inference time on the test set containing 226 samples was 0.0645 s, corresponding to an average inference time of 0.0081 s per batch and 0.000286 s per sample. Since the parking availability series was sampled at 2 h intervals, such inference latency is sufficiently low for real-time single-site forecasting. The dynamic multi-kernel module is introduced only into the encoder, which helps control the additional online computational cost. From

the perspective of scalability, the proposed framework is well suited to the current single-site forecasting task, while its extension to larger parking networks or multi-source input settings may require further optimization.

Figures 6–8 present the local prediction results of the three forecasting scenarios, respectively. It can be observed from the figures that the proposed model can effectively track the overall variation trend of the original series, and the predicted values remain highly consistent with the true values over most time intervals. For periodic peaks, troughs, and their adjacent fluctuations, the model also demonstrates good fitting performance, indicating that the DWT-denoised input sequence is able to provide more stable structural information for subsequent temporal modeling. Meanwhile, the encoder-side dynamic multi-kernel decomposition mechanism helps separate long-term trends from local fluctuations, enabling the model to more accurately recover the periodic variation patterns of the parking availability series under different forecasting horizons.

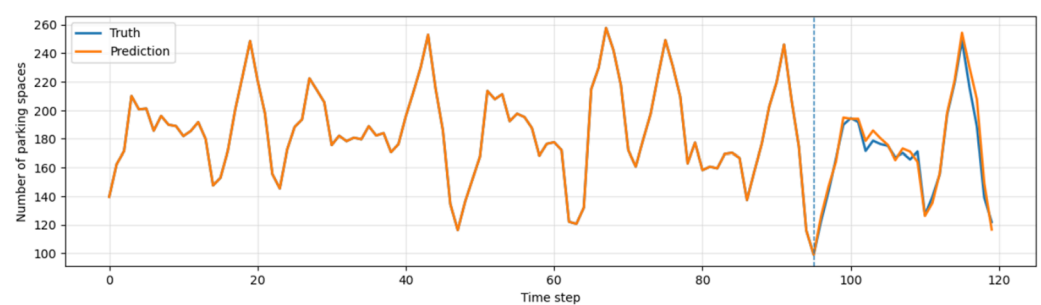


Figure 6. Prediction results for a step length of 24.

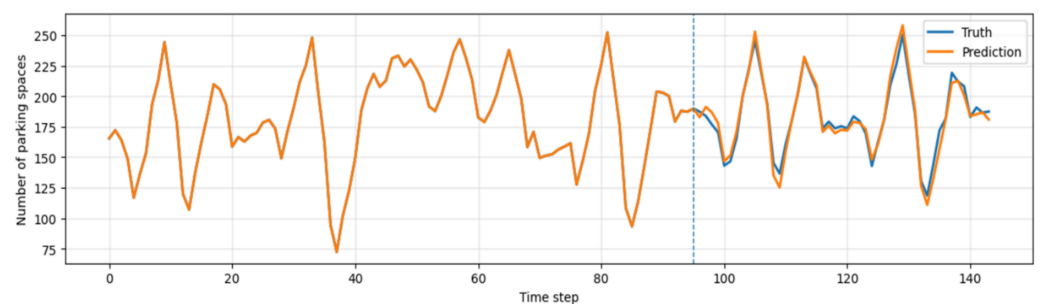


Figure 7. Prediction results for a step length of 48.

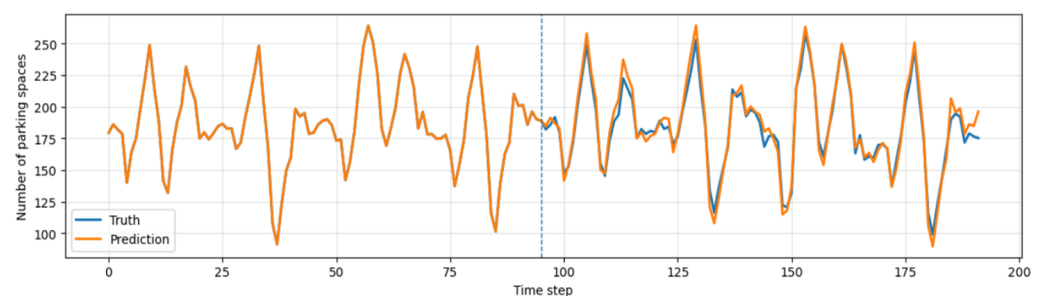


Figure 8. Prediction results for a step length of 96.

Furthermore, Figures 6–8 also show that some deviations still exist between the predicted and true curves around local mutation points or intervals with rapid peak–trough switching. This phenomenon is more evident in the 48-step and 96-step forecasting scenarios, indicating that when the forecasting interval becomes longer, local high-frequency disturbances may still affect the model to a certain extent. Nevertheless, overall, the predicted curves remain highly consistent with the true curves in both phase and amplitude,

suggesting that the proposed model does not exhibit obvious cumulative error propagation as the forecasting horizon increases.

In Figures 6–8, the blue dashed line represents the forecasting origin; the left side shows the historical observation sequence used by the model, while the right side shows the prediction horizons of 24, 48, and 96 time steps, respectively.

4.4. Analysis of the Dynamic Multi-Kernel Decomposition Module

To further examine the role of the proposed dynamic multi-kernel decomposition mechanism under different forecasting tasks, the average kernel weights of the encoder-side dynamic decomposition blocks were calculated on one test batch, as shown in Figure 9. Since the encoder contains two layers and each layer includes two dynamic decomposition blocks, four groups of kernel weights are obtained under each forecasting scenario.

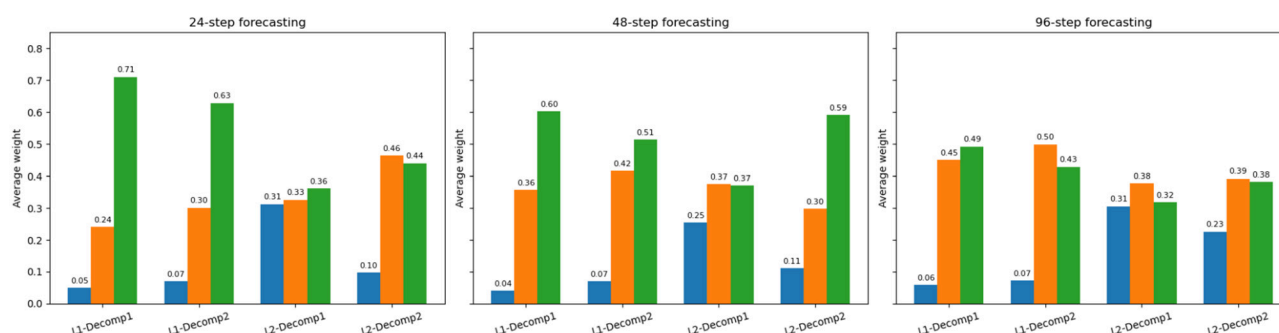


Figure 9. Average kernel-weight distributions of the encoder-side dynamic decomposition blocks under different forecasting horizons.

As shown in Figure 9, the dynamic multi-kernel decomposition module exhibits distinct scale preferences under different forecasting horizons. In the 24-step forecasting scenario, the first-layer decomposition blocks assign the highest weights to the large-scale kernel (49), indicating that short-term prediction relies more on smooth trend information. In the 48-step forecasting scenario, the contribution of the medium-scale kernel (25) becomes more pronounced, suggesting that medium-term prediction requires a better balance between trend information and intermediate-scale variations. In the 96-step forecasting scenario, the weight allocation becomes more balanced across different kernel scales, implying that long-term forecasting depends more on collaborative multi-scale modeling rather than a single dominant temporal scale.

These findings indicate that the proposed dynamic multi-kernel decomposition mechanism does not rely on a fixed kernel preference, but instead adaptively adjusts the importance assigned to different temporal scales according to the forecasting task and hierarchical feature representation. Its main contribution is to enhance the model's ability to represent heterogeneous multi-scale temporal patterns.

4.5. Comparison with Benchmark Models

To further validate the effectiveness of the proposed model, several representative forecasting models were selected as baselines, including LSTM, TCN, Transformer, TimesNet, and the original Autoformer. These baselines represent different sequence modeling paradigms, namely recurrent modeling, convolutional temporal modeling, standard attention-based modeling, periodic-pattern modeling, and decomposition-based long-sequence forecasting. In addition, three recent and competitive forecasting models, namely PatchTST, iTransformer, and FEDformer, as well as the decomposition-based linear model DLinear and the deep residual forecasting model N-BEATS, were further included to strengthen the comparison. All models were evaluated on the same dataset under the

same train-test split and 24-step forecasting setting. Each experiment was independently repeated 10 times, and the results in Table 5 are reported as mean $\pm$ standard deviation. To further reflect the computational cost of different methods, Table 5 also reports the overall runtime of each model under the current experimental setting.

Table 5. Model error comparison.

Model	MAE	RMSE	WMAPE (%)	Overall Runtime (s)
The proposed model	5.5914 $\pm$ 0.3496	7.4130 $\pm$ 0.5842	3.0644 $\pm$ 0.1915	9.14
Autoformer	6.2095 $\pm$ 0.2302	8.3732 $\pm$ 0.5209	3.4031 $\pm$ 0.1261	6.94
Transformer	10.8499 $\pm$ 0.7365	14.1363 $\pm$ 1.3565	5.9463 $\pm$ 0.4019	3.45
Timesnet	11.5796 $\pm$ 0.6623	14.9225 $\pm$ 1.1634	6.3462 $\pm$ 0.3769	10.31
LSTM	19.9742 $\pm$ 1.1435	24.3592 $\pm$ 0.7909	10.9469 $\pm$ 0.6226	1.97
TCN	11.0246 $\pm$ 0.1961	14.6618 $\pm$ 0.0592	6.0421 $\pm$ 0.1073	2.75
PatchTST	8.4849 $\pm$ 1.0644	11.4362 $\pm$ 1.7609	4.6502 $\pm$ 0.5844	21.44
iTransformer	8.2561 $\pm$ 1.0711	11.1348 $\pm$ 1.8847	4.5248 $\pm$ 0.5882	2.05
FEDformer	6.9348 $\pm$ 1.6363	8.9764 $\pm$ 1.7748	3.8006 $\pm$ 0.8968	7.62
DLinear	15.6545 $\pm$ 0.4005	20.3688 $\pm$ 0.5826	8.5807 $\pm$ 0.2157	1.34
N-BEATS	8.7183 $\pm$ 0.3071	11.8412 $\pm$ 0.5826	4.7780 $\pm$ 0.1651	2.64

From Table 5, the following conclusions can be drawn:

- (1) The proposed model achieves the lowest mean MAE, RMSE, and WMAPE among all compared methods, indicating the best overall forecasting accuracy. Compared with the original Autoformer, the proposed model further improves all three error metrics, confirming the effectiveness of combining DWT-based denoising with encoder-side dynamic multi-kernel decomposition. This advantage is further supported by the repeated-experiment interval estimates. For instance, under the 24-step forecasting setting, the 95% confidence interval of the proposed model for MAE is [5.3413, 5.8415], compared with [6.0448, 6.3742] for Autoformer, and the same pattern is observed for RMSE and WMAPE.
- (2) Autoformer and FEDformer show the strongest benchmark performance after the proposed model, while PatchTST, iTransformer, and N-BEATS also achieve relatively competitive results. Autoformer and FEDformer explicitly incorporate decomposition ideas and are therefore better suited to long-sequence forecasting with trend-fluctuation interactions. PatchTST improves long-sequence representation through patch-based tokenization, and N-BEATS benefits from stacked residual backcast-forecast blocks with strong nonlinear approximation ability. Although iTransformer also performs competitively, its structural advantage is more pronounced in multi-variate forecasting.
- (3) Transformer, TimesNet, and TCN yield noticeably larger errors. Although these models can capture temporal dependence through global attention, periodic-pattern modeling, or dilated convolution, respectively, they do not explicitly perform adaptive decomposition under mixed temporal scales, which limits their performance in the present task.
- (4) LSTM is less effective in long-horizon multi-step forecasting because recurrent modeling is more prone to error accumulation and weaker long-range dependency capture. DLinear benefits from a simple decomposition-based linear structure, but its fixed decomposition and linear extrapolation capacity are limited for highly nonlinear and non-stationary parking availability series.
- (5) In terms of computational cost, the proposed model introduces a certain amount of additional runtime due to the DWT-based denoising step and the encoder-side dynamic multi-kernel decomposition mechanism. Nevertheless, its overall runtime

remains acceptable under the current experimental setting, while the gain in forecasting accuracy is evident. Therefore, the proposed framework provides a reasonable trade-off between computational efficiency and predictive performance, and remains applicable in practical parking availability forecasting scenarios.

To further illustrate the forecasting behavior of different models at the sample level, Figure 10 presents the 24-step forecasting results for a test sample.

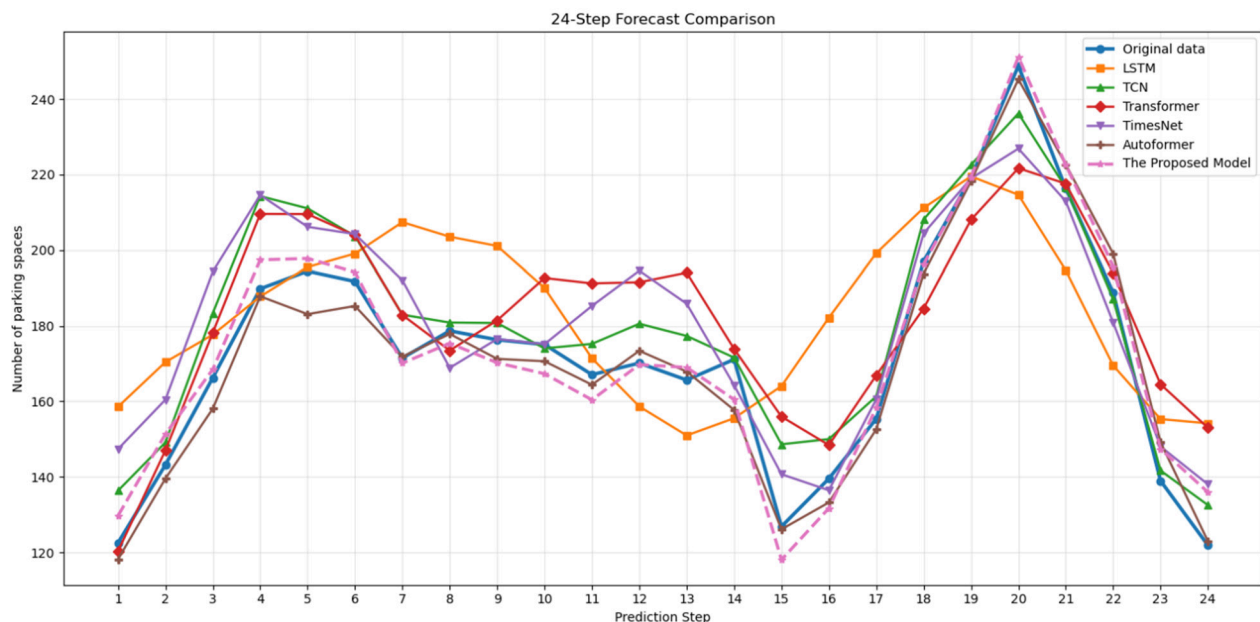


Figure 10. Comparison of 24-step forecasting results for Sample 4.

As shown in the figure, although all benchmark models are able to capture the general fluctuation pattern of the traffic flow series, noticeable differences remain in their abilities to track local variations and turning points. Compared with the baseline methods, the proposed model exhibits the closest agreement with the ground truth over most prediction horizons. In particular, it achieves more accurate fitting in peak, trough, and transition regions, indicating a stronger capability to characterize complex temporal dynamics and multi-scale variations. By contrast, the original Autoformer can capture the overall trend reasonably well, but still presents observable deviations in several local intervals, especially when the series changes rapidly. These observations are consistent with the quantitative results reported in Table 5, further confirming the effectiveness of the proposed model for multi-step parking occupancy forecasting.

To further examine the prediction behavior of the proposed framework, the residual distribution on the test set was analyzed, where the residual is defined as the difference between the predicted value and the true value. Figure 11 presents the residual histogram of the proposed model. The blue dashed line denotes zero residual. This analysis helps reveal whether the model exhibits systematic prediction bias and how the prediction errors are distributed overall.

As shown in Figure 11, the residuals of the proposed model are mainly concentrated around zero, indicating that the model does not exhibit obvious systematic bias on the test set. The residual mean is close to zero, which further suggests that the overall prediction is approximately unbiased. In addition, most residuals fall within a relatively compact range, showing that the majority of prediction errors remain moderate. However, a small number of tail residuals can still be observed, especially on the negative side, implying that larger

underestimation errors may occur in some periods with abrupt local fluctuations or rapid changes in parking availability.

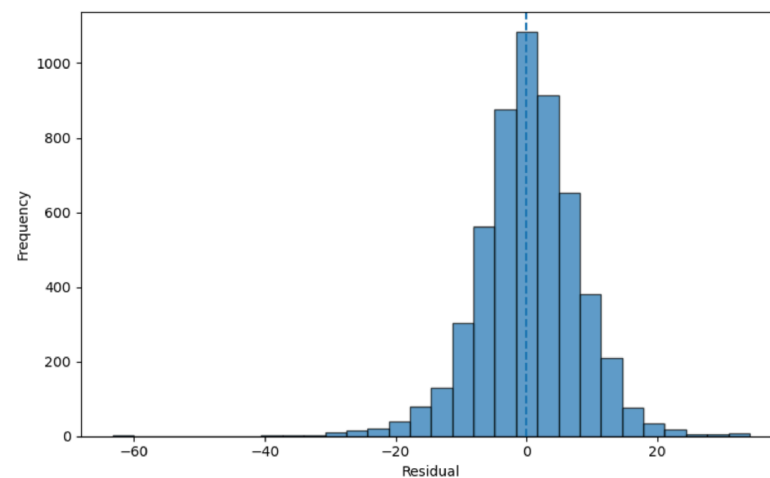


Figure 11. Residual distribution of the proposed model on the test set.

4.6. Ablation Study

To further clarify the contribution of each component in the proposed framework, an ablation study was conducted. Since the proposed model consists of two main components, namely DWT-based denoising and the encoder-side dynamic multi-kernel decomposition module, four model variants were compared: (1) the original Autoformer, (2) DWT-Autoformer, which uses only DWT denoising, (3) DMKD-Autoformer, which introduces only the dynamic multi-kernel decomposition module, and (4) the full proposed model, which combines both DWT denoising and dynamic multi-kernel decomposition. Table 6 reports the ablation results as mean $\pm$ standard deviation over 10 independent runs.

Table 6. Ablation results of different model components.

Model	MAE	RMSE	WMAPE (%)
Autoformer	6.2095 $\pm$ 0.2302	8.3732 $\pm$ 0.5209	3.4031 $\pm$ 0.1261
DWT-Autoformer	6.0893 $\pm$ 0.1663	8.1514 $\pm$ 0.2344	3.3373 $\pm$ 0.0911
DMKD-Autoformer	9.7248 $\pm$ 0.2768	12.4957 $\pm$ 0.3549	5.3266 $\pm$ 0.1516
The proposed model	5.5914 $\pm$ 0.3496	7.4130 $\pm$ 0.5842	3.0644 $\pm$ 0.1915

Compared with the original Autoformer, DWT-Autoformer improves forecasting performance, indicating that wavelet-based denoising helps suppress noise-like disturbances and provides cleaner input signals for subsequent prediction. By contrast, DMKD-Autoformer performs worse than the original Autoformer when the dynamic multi-kernel decomposition module is introduced without DWT denoising. This result suggests that adaptive multi-scale decomposition alone is insufficient to offset the influence of noisy and non-stationary raw inputs, and that the effectiveness of the dynamic decomposition mechanism depends on an appropriately denoised input representation. The full proposed model achieves the best overall performance, demonstrating that DWT-based denoising and encoder-side dynamic multi-kernel decomposition are complementary when jointly applied.

4.7. Additional Evaluation Under a Temporally Consistent Preprocessing Protocol

To further enhance the rigor of the experimental evaluation, an additional assessment was conducted under a temporally consistent preprocessing protocol. The purpose of this

supplementary experiment was to verify the predictive performance of the proposed framework under a stricter chronological setting, in which preprocessing, sample construction, model training, and testing were all carried out in a fully time-consistent manner.

The revised procedure consisted of four steps. First, a chronological cutoff point was defined on the original parking availability series, and the training and test subsets were separated in time so that all training observations preceded the test observations. Second, after the chronological split was determined, forecasting samples were constructed by a sliding-window strategy. For each sample, only the historical observations available up to the corresponding forecasting origin were used to form the model input, while the subsequent observations were used as prediction targets. Samples spanning the training-test boundary were excluded to maintain a strict temporal separation between the two subsets. Third, for the proposed DWT-based framework, wavelet denoising was not performed on the entire sequence in advance. Instead, it was applied separately within each historical input window, and the denoised sequence obtained from that window was then fed into the forecasting model. In this way, the preprocessing stage remained fully consistent with the temporal order of the prediction task.

In the revised protocol, each historical input window was independently processed by discrete wavelet transform using the Daubechies 4 (db4) wavelet with a four-level decomposition. Symmetric boundary extension was adopted in the implementation. The approximation coefficient was retained, while soft-threshold denoising was applied to the detail coefficients. The threshold calculation rule and the corresponding soft-thresholding operation followed the formulation introduced in Section 4.2. After thresholding, the denoised signal was reconstructed by inverse wavelet transform and used as the model input for that sample. The implementation was based on the PyWavelets library, and the reconstructed sequence was aligned with the original input-window length before being fed into the forecasting model.

Fourth, all benchmark models were re-evaluated under the same temporally consistent protocol to ensure a fair comparison. In addition, each model was repeatedly run 10 times to further examine result stability, and the final results are reported as mean $\pm$ standard deviation.

The comparative results are summarized in Table 7. Under this stricter evaluation setting, the proposed DWT-based multi-kernel Autoformer still achieved the best overall performance among all compared models. Specifically, the proposed method obtained the lowest MAE, RMSE, and WMAPE values across repeated runs. Compared with the original Autoformer, PatchTST, and FEDformer, the proposed framework consistently maintained superior predictive accuracy under the temporally consistent preprocessing setting. These results indicate that the effectiveness of the proposed method remains stable after adopting the revised evaluation procedure.

Table 7. Comparative results under a temporally consistent preprocessing protocol.

Model	MAE	RMSE	WMAPE (%)
The proposed model	24.7333 $\pm$ 0.8340	36.7983 $\pm$ 0.8795	13.8164 $\pm$ 0.4659
Autoformer	25.7773 $\pm$ 1.2870	38.1513 $\pm$ 1.3441	14.3996 $\pm$ 0.7190
PatchTST	29.2916 $\pm$ 0.3966	39.4419 $\pm$ 0.4224	16.3628 $\pm$ 0.2216
FEDformer	28.1981 $\pm$ 0.2890	37.3577 $\pm$ 0.3449	15.7520 $\pm$ 0.1615

5. Conclusions

- (1) This study developed a DWT-Autoformer framework for parking availability forecasting by combining wavelet-based denoising with encoder-side dynamic multi-kernel decomposition. DWT suppresses high-frequency noise while preserving the dom-

inant trend and periodic structure, whereas dynamic multi-kernel decomposition strengthens the representation of heterogeneous temporal patterns across different scales. The proposed model achieved MAE values below 6, WMAPE values below 3.2%, and R2 values above 0.94 under different forecasting horizons, and reduced MAE, RMSE, and WMAPE by about 10% relative to the original Autoformer under the 24-step forecasting setting.

- (2) This study is based on real-world single-site parking availability data. Such a single-sequence setting is less affected by external disturbances, making the data more direct to interpret. Meanwhile, the current data scope is still limited, and further validation on larger and more diverse datasets is needed to better assess the robustness, generalization, and scalability of the proposed framework.
- (3) Future research may further expand the dataset scale, incorporate multi-source information such as traffic flow, weather, and event data, and examine the practical applicability of the proposed framework under more diverse parking and urban traffic scenarios.

Author Contributions: Conceptualization, C.Z. (Chao Zeng), C.Z. (Changhao Zhao), X.H. and C.M.; methodology, C.Z. (Chao Zeng), X.H. and C.M.; software, C.Z. (Changhao Zhao) and X.H.; validation, C.Z. (Changhao Zhao) and X.H.; formal analysis, C.Z. (Chao Zeng), C.Z. (Changhao Zhao) and X.H.; investigation, C.Z. (Chao Zeng) and C.Z. (Changhao Zhao); resources, C.Z. (Chao Zeng) and C.Z. (Changhao Zhao); data curation, C.Z. (Chao Zeng) and C.Z. (Changhao Zhao); writing—original draft preparation, C.Z. (Changhao Zhao) and X.H.; writing—review and editing, C.Z. (Chao Zeng), X.H. and C.M.; visualization, C.Z. (Changhao Zhao) and X.H.; supervision, C.Z. (Chao Zeng) and C.M.; project administration, C.Z. (Chao Zeng); funding acquisition, C.Z. (Chao Zeng). All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported by the Chongqing Social Science Planning Project (2025NDYB058)—Research on the Air-Ground Traffic Coupling Mechanism and Traffic Facility Renewal Strategies from the Perspective of High-Quality Development of Chongqing’s Low-Altitude Economy.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The original contributions presented in the study are included in the article; further inquiries can be directed to the corresponding author.

Conflicts of Interest: The authors declare no conflicts of interest.

References

- Ma, C.; Huang, X.; Li, J. A review of research on urban parking prediction. *J. Traffic Transp. Eng. Engl. Ed.* **2024**, *11*, 700–720. [\[CrossRef\]](#)
- Ozkaya, M.; Turunc, A. A reference architecture for smart car parking management systems. *Systems* **2025**, *13*, 70. [\[CrossRef\]](#)
- Tekouabou, S.C.K.; Cherif, W.; Silkan, H. Improving parking availability prediction in smart cities with IoT and ensemble-based model. *J. King Saud Univ. Comput. Inf. Sci.* **2022**, *34*, 687–697. [\[CrossRef\]](#)
- Xiao, X.; Peng, Z.; Lin, Y.; Jin, Z.; Shao, W.; Chen, R.; Mao, G. Parking prediction in smart cities: A survey. *IEEE Trans. Intell. Transp. Syst.* **2023**, *24*, 10302–10326. [\[CrossRef\]](#)
- Awan, F.M.; Saleem, Y.; Minerva, R.; Crespi, N. A comparative analysis of machine/deep learning models for parking space availability prediction. *Sensors* **2020**, *20*, 322. [\[CrossRef\]](#) [\[PubMed\]](#)
- Dong, C.; Chang, N. Overview of the identification of traffic accident-prone locations driven by big data. *Digit. Transp. Saf.* **2023**, *2*, 67–76. [\[CrossRef\]](#)
- Zeng, C.; Ma, C.; Wang, K.; Cui, Z. Predicting vacant parking space availability: A DWT-Bi-LSTM model. *Phys. A* **2022**, *599*, 127498. [\[CrossRef\]](#)
- Yu, F.; Guo, J.; Zhu, X.; Shi, G. Real time prediction of unoccupied parking space using time series model. In *Proceedings of the 2015 International Conference on Transportation Information and Safety (ICTIS)*; IEEE: Piscataway, NJ, USA, 2015; pp. 370–374.

9. Channamallu, S.S.; Kermanshachi, S.; Rosenberger, J.M.; Pamidimukkala, A. Parking occupancy prediction and analysis—A comprehensive study. *Transp. Res. Procedia* **2023**, *73*, 297–304. [\[CrossRef\]](#)
10. Fokker, E.S.; Koch, T.; van Leeuwen, M.; Dugundji, E.R. Short-term forecasting of off-street parking occupancy. *Transp. Res. Rec.* **2022**, *2676*, 637–654.
11. Fan, J.; Hu, Q.; Tang, Z. Predicting vacant parking space availability: An SVR method with fruit fly optimization. *IET Intell. Transp. Syst.* **2018**, *12*, 1414–1420. [\[CrossRef\]](#)
12. Zhao, Z.; Zhang, Y.; Zhang, Y. A comparative study of parking occupancy prediction methods considering parking type and parking scale. *J. Adv. Transp.* **2020**, *2020*, 5624586. [\[CrossRef\]](#)
13. Sadia, K.; Reza, R.; Alam, A.; Rahman, M.A. Car parking availability prediction: A comparative study of LSTM and random forest regression approaches. *Int. J. Autom. Artif. Intell. Mach. Learn.* **2021**, *2*, 16–29. [\[CrossRef\]](#)
14. Wang, T.; Hu, S.; Jiang, Y. Predicting shared-car use and examining nonlinear effects using gradient boosting regression trees. *Int. J. Sustain. Transp.* **2021**, *15*, 893–907.
15. Zhou, Y.; Ye, X.; Yan, X.; Wang, T.; Chen, J. Multi-step parking demand prediction model based on multi-graph convolutional transformer. *Systems* **2024**, *12*, 487. [\[CrossRef\]](#)
16. Arjona, J.; Linares, M.; Casanovas-Garcia, J.; Vázquez, J.J. Improving parking availability information using deep learning techniques. *Transp. Res. Procedia* **2020**, *47*, 385–392. [\[CrossRef\]](#)
17. Caicedo, F.; Blazquez, C.; Miranda, P. Prediction of parking space availability in real time. *Expert Syst. Appl.* **2012**, *39*, 7281–7290. [\[CrossRef\]](#)
18. Liu, Y.; Liu, C.; Luo, X. Spatiotemporal deep-learning networks for shared-parking demand prediction. *J. Transp. Eng. Part A Syst.* **2021**, *147*, 04021026. [\[CrossRef\]](#)
19. Qi, Q.; Cheng, R.; Ge, H. Short-term inbound rail transit passenger flow prediction based on BILSTM model and influence factor analysis. *Digit. Transp. Saf.* **2023**, *2*, 12–22. [\[CrossRef\]](#)
20. Zeng, C.; Ma, C.; Wang, K.; Cui, Z. Parking occupancy prediction method based on multi factors and stacked GRU-LSTM. *IEEE Access* **2022**, *10*, 47361–47370. [\[CrossRef\]](#)
21. Xu, Z.; Tang, X.; Ma, C.; Zhang, R. Research on parking space detection and prediction model based on CNN-LSTM. *IEEE Access* **2024**, *12*, 30085–30100. [\[CrossRef\]](#)
22. Yang, S.; Ma, W.; Pi, X.; Qian, S. A deep learning approach to real-time parking occupancy prediction in transportation networks incorporating multiple spatio-temporal data sources. *Transp. Res. Part C Emerg. Technol.* **2019**, *107*, 248–265. [\[CrossRef\]](#)
23. Li, G.; Xiang, Z. Parking demand forecasting based on improved complete ensemble empirical mode decomposition and GRU model. *Eng. Appl. Artif. Intell.* **2023**, *119*, 105717. [\[CrossRef\]](#)
24. Wang, C.; Chen, Y.; Zhang, S.; Zhang, Q. Stock market index prediction using deep Transformer model. *Expert Syst. Appl.* **2022**, *208*, 118128. [\[CrossRef\]](#)
25. Ji, T.; Zhao, C.; Ji, Y.; Du, Y. Mobile network traffic prediction using temporal fusion transformer. *IEEE Trans. Artif. Intell.* **2025**, *6*, 2685–2699. [\[CrossRef\]](#)
26. Moon, H.J.; Cho, S.B. Traffic prediction by graph transformer embedded with subgraphs. *Expert Syst. Appl.* **2025**, *272*, 126799. [\[CrossRef\]](#)
27. Ji, T.; Zhao, C.; Ji, Y.; Du, Y. A two-stage framework for parking search behavior prediction through adversarial inverse reinforcement learning and transformer. *Expert Syst. Appl.* **2024**, *255*, 124548. [\[CrossRef\]](#)
28. Huang, Y.; Dong, Y.; Tang, Y.; Li, L. Leverage multi-source traffic demand data fusion with transformer model for urban parking prediction. *arXiv* **2024**, arXiv:2405.01055.
29. Li, J.; Ma, H.; Li, X.; Ma, R. Semantics-aware spatial-temporal dynamic graph Transformer network for on-street parking occupancy prediction. *IEEE Trans. Intell. Transp. Syst.* **2026**, *27*, 842–856. [\[CrossRef\]](#)
30. Chen, H.; Zhang, S.; Xie, J.; Liu, H.; Chen, G. MV-STGCN: Multi-view spatial-temporal graph convolutional networks for on-street parking availability prediction. *ACM Trans. Spat. Algorithms Syst.* **2025**, *11*, 13. [\[CrossRef\]](#)
31. Zhou, T.; Ma, Z.; Wen, Q.; Wang, X.; Sun, L.; Jin, R. Fedformer: Frequency enhanced decomposed Transformer for long-term series forecasting. In *Proceedings of the 39th International Conference on Machine Learning*; PMLR: Baltimore, MD, USA, 2022; Volume 162, pp. 27268–27286.
32. Zeng, A.; Chen, M.; Zhang, L.; Xu, Q. Are Transformers effective for time series forecasting? In *Proceedings of the AAAI Conference on Artificial Intelligence*; AAAI Press: Washington, DC, USA, 2023; Volume 37, pp. 11121–11128.
33. Nie, Y.; Nguyen, N.H.; Sinthong, P.; Kalagnanam, J. A time series is worth 64 words: Long-term forecasting with Transformers. In *Proceedings of the Eleventh International Conference on Learning Representations (ICLR)*, Kigali, Rwanda, 1–5 May 2023.
34. Liu, Y.; Hu, T.; Zhang, H.; Wu, H.; Wang, S.; Ma, L.; Long, M. iTransformer: Inverted Transformers are effective for time series forecasting. In *Proceedings of the Twelfth International Conference on Learning Representations (ICLR)*, Vienna, Austria, 7–11 May 2024.

35. Zhu, C.; An, C.; He, R.; Zhang, C.; Cheng, L. Prediction of the vehicle lane-changing distance in an urban inter-tunnel weaving section based on wavelet transform and dual-channel neural network. *IET Intell. Transp. Syst.* **2024**, *18*, 2078–2096. [[CrossRef](#)]
36. Zhang, T. Network level spatial temporal traffic forecasting with hierarchical-attention-LSTM. *Digit. Transp. Saf.* **2024**, *3*, 233–245. [[CrossRef](#)]
37. Tao, H.; Lu, X. Automatic smoky vehicle detection from traffic surveillance video based on vehicle rear detection and multi-feature fusion. *IET Intell. Transp. Syst.* **2019**, *13*, 252–259. [[CrossRef](#)]
38. Yu, G.; Zou, J.; Hu, X.; Aviles-Rivero, A.I.; Qin, J.; Wang, S. Revitalizing multivariate time series forecasting: Learnable decomposition with inter-series dependencies and intra-series variations modeling. In Proceedings of the Forty-First International Conference on Machine Learning (ICML), Vienna, Austria, 21–27 July 2024.
39. Hu, Y.; Liu, P.; Zhu, P.; Cheng, D.; Dai, T. Adaptive multi-scale decomposition framework for time series forecasting. In Proceedings of the AAAI Conference on Artificial Intelligence, Philadelphia, PA, USA, 25 February–4 March 2025; Volume 39.

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.