Sparse-Scale Transformer with Bidirectional Awareness for Time Series Forecasting

Ying Liu[1], Bo Liu[1,*], Sheng Huang[2], Gang Luo[3], Wenbo Hu[1], Meng Wang[1], Richang Hong[1]

[1]School of Computer Science and Information Engineering, Hefei University of Technology
[2]School of Software Engineering and Big Data, Chongqing University
[3]Anhui Guoqi Technology Co., Ltd
liuying1@mail.hfut.edu.cn, {boliu, wenbohu, hongrc}@hfut.edu.cn, huangsheng@cqu.edu.cn, luogang@gqgraph.com, eric.mengwang@gmail.com

Abstract

Time series forecasting (TSF) plays a crucial role in many real-world applications, such as weather prediction and economic planning. While Transformer-based models have shown strong capabilities in modeling long-range dependencies, effectively capturing the multi-scale temporal dynamics inherent in time series remains a major challenge. Existing methods often adopt time-windows of varying sizes, which may introduce noisy or irrelevant representations when mismatched with the underlying temporal patterns, potentially leading to overfitting. In this paper, we propose Sparse-Scale Transformer (SSformer) with Bidirectional Awareness for Time Series Forecasting to enhance the multi-scale modeling for time series. Specifically, we propose a novel Sparse-Scale Convolution (SSC) block that imposes sparsity on scales to obtain the informative representations by evaluating the intra-scale segment similarity of time series, and utilizes scale-specific convolutions to extract local patterns. Furthermore, we design a Bidirectional-Scale Interaction (BSI) block to explicitly model scale correlations in both coarse-to-fine and fine-to-coarse directions. Finally, scale predictions are ensembled to fully exploit the complementary forecasting capabilities across scales. Extensive experiments on various real-world datasets demonstrate that SSformer achieves state-of-the-art performance with superior efficiency.

Code — https://github.com/yingliu-coder/SSformer

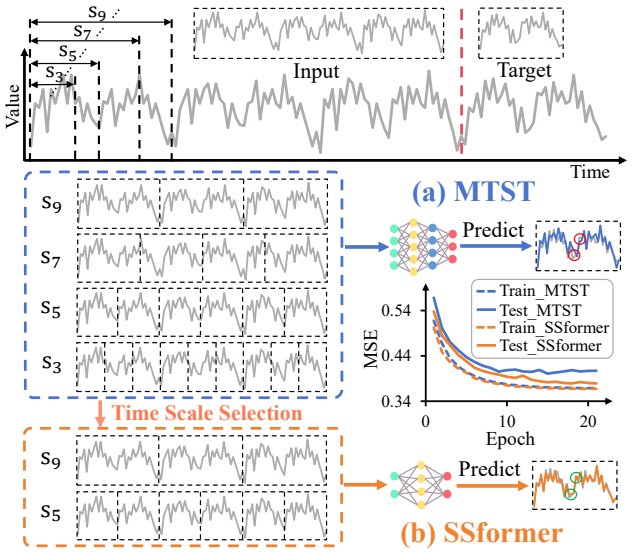

Figure 1: (a) MTST applies fixed time-windows regardless of temporal structures in time series, introducing noise and causing overfitting. (b) The proposed model, SSformer imposes sparsity on scales to obtain the more discriminative representations, alleviating overfitting and improving inference efficiency.

Introduction

Time Series Forecasting (TSF) is a fundamental task in machine learning, with broad applications in domains such as weather prediction (Wu et al. 2023b; Nguyen et al. 2024), economic planning (Lazcano, Herrera, and Monge 2023; Cheng et al. 2022), energy consumption (Gao et al. 2023; Stefenon et al. 2023), and traffic analysis (Liu et al. 2023; Kieu et al. 2024). The objective is to model temporal dependencies in historical observations to predict future values.

Classical methods such as autoregressive integrated moving average (ARIMA) (Box and Jenkins 1968) and seasonal-trend decomposition via Loess (STL) (Cleveland et al. 1990) can model low-order linear dynamics, trend, and seasonality effectively but are limited in capturing the complex, nonlinear, and often nonstationary behaviors prevalent in real-world time series. Recent advances in deep learning have significantly propelled the field of time series forecasting. In particular, Transformer-based models have demonstrated strong potential to capture long-range dependencies and complex temporal patterns, emerging as the dominant paradigm (Shao et al. 2025; Qiu et al. 2024; Liu et al. 2024).

Real-world time series often exhibit rich temporal dynamics on multiple scales (Wang et al. 2024; Cai et al. 2024). For example, hourly weather data capture short-term fluctuations, daily patterns reflect local trends, while yearly aggregates reveal long-term climatic behavior. These characteristics underscore the necessity of incorporating a multi-

*Bo Liu is the corresponding author

scale temporal modeling paradigm to effectively capture complex and scale-varying temporal dependencies. A common method is to downsample the raw series by multiple scale factors via an average pooling operation (Shabani et al. 2023; Wang et al. 2023a), but the nature of downsample makes them lose fine-grained local patterns. One popular approach adopted by existing Transformer based TSF models is using 1D convolution with varying-size time-windows. For example, MTST (Zhang et al. 2024a) divides the time series into different temporal scales with various time-windows; Ada-MSHyper (Shang et al. 2024) employs convolutional aggregation windows with different sizes to get the feature representations at different scales. Using the time-window whose size is different from time-series pattern tends to introduce noisy feature, leading to model overfitting problem. Fig. 1 illustrates this by comparing MTST and SSformer. As shown in Fig. 1(a), MTST uniformly applies a set of scales with fixed time-windows regardless of whether they match the underlying temporal patterns in the input series. This rigid strategy may cause noise, potentially leading to overfitting prediction. In contrast, SSformer (Fig. 1(b)) selectively retains the most informative scales, effectively filtering out noisy representations and improving both prediction accuracy and training stability. This scale-sparse strategy encourages more effective and efficient multi-scale modeling.

In this paper, we propose a Sparse-Scale Transformer (SSformer) with Bidirectional Awareness for Time Series Forecasting to improve the capability of multi-scale modeling. In concrete, we introduce a Sparse-Scale Convolution (SSC) block that adaptively selects the most effective temporal scales based on the temporal patterns of each time series. Then, scale-specific convolutional operations are designed to capture local temporal dependencies. Subsequently, with the aim of adequately utilizing the correlations across scales, we propose a Bidirectional-Scale Interaction (BSI) block to incorporate two complementary interaction pathways. One is coarse-to-fine interaction where coarse-grained representations provide trend-aware guidance to the feature learning of fine-scales. The other is fine-to-coarse interaction where fine-grained representations supplement the lost details of coarse-scales by injecting the detail-rich information. Finally, we ensembles multi-scale predictions to leverage complementary forecasting capabilities. Extensive experimental results demonstrate that SSformer significantly improves the ability of multi-scale modeling with Transformer architecture and achieves state-of-the-art performance on various real-world datasets. For example, compared with Times2D (Nematirad, Pahwa, and Natarajan 2025), SSformer achieves an impressive 9.2% reduction in MSE, 4.6% reduction in MAE, and 77.8% reduction in model parameters on Weather dataset.

Our contributions are summarized as follows:

- Motivated by the overfitting problem arising from mismatches between time-window sizes and underlying temporal patterns, we propose SSformer to filter out noisy scales, which effectively improves the model efficiency and forecasting accuracy.
- Going beyond previous multi-scale methods, we propose a Sparse-Scale Convolution (SSC) block to select the informative scales and introduce scale-specific convolution operations to capture local dependencies.
- To effectively leverage the correlation across different scales, we design a Bidirectional-Scale Interaction (BSI) block that explicitly establishes scale interaction in both coarse-to-fine and fine-to-coarse directions.

Related Work

Time Series Forecasting

Time series forecasting has been extensively studied using deep learning methods based on various foundational architectures, including RNN, MLP, CNN, and Transformer. RNN-based methods represent sequential time steps under the Markov assumption (Lai et al. 2018; Shen, Li, and Kwok 2020) but fail in capturing long-term dependencies and are limited in efficiency due to their inherently sequential computation paradigm. MLP-based methods offer notably reduced computational complexity (Wang et al. 2024; Huang et al. 2024; Hu et al. 2025), but the simplicity of linear mappings suffers from bottlenecks while modeling sophisticated temporal dependencies. CNN-based models (Wang et al. 2023a; Wu et al. 2023a) are efficient in extracting short-term patterns but are constrained by fixed kernel sizes and limited receptive fields, which hamper their ability to capture long-term dependencies and multi-scale features.

Transformer-based models have recently gained increasing interest in TSF due to their powerful capacity to capture long-range dependencies. FEDformer (Zhou et al. 2022) applies Fourier transformation to the frequency domain to model the global dependencies. iTransformer (Liu et al. 2024) utilizes attention on the inverted dimension to capture multivariate correlations. Times2D (Nematirad, Pahwa, and Natarajan 2025) reshapes the 1D time series into a 2D tensor by using the Fast Fourier Transform to address the multi-periodicity and sharp fluctuations in time series data.

Multi-Scale Feature Extraction for Time Series

Multi-scale modeling is effective for disentangling patterns and extracting features in many domains such as natural language processing (Nawrot et al. 2022; Li et al. 2022a; Zhao et al. 2021) and computer vision (Li et al. 2022b; Zhang et al. 2021; Wang et al. 2023b), yet remains relatively underexplored in TSF. Pyraformer (Liu et al. 2022) uses a pyramidal attention module to summarize multi-scale features, but concatenating tokens of all scales into time dimension. This overlook the inter-scale interaction that is beneficial to improve forecasting performance through integrating with characteristics across different scales.

NHITS (Challu et al. 2023) and Scaleformer (Shabani et al. 2023) employ multi-scale downsampling tokenization to model dependencies of distinct scales. However, they typically utilize pooling operation or down-sampling to generate tokens, which contains weak semantic information and neglect inherent temporal correlations between the segments of time series. To extract comprehensive semantic information, MTST (Zhang et al. 2024a) and MTPNet (Zhang et al.

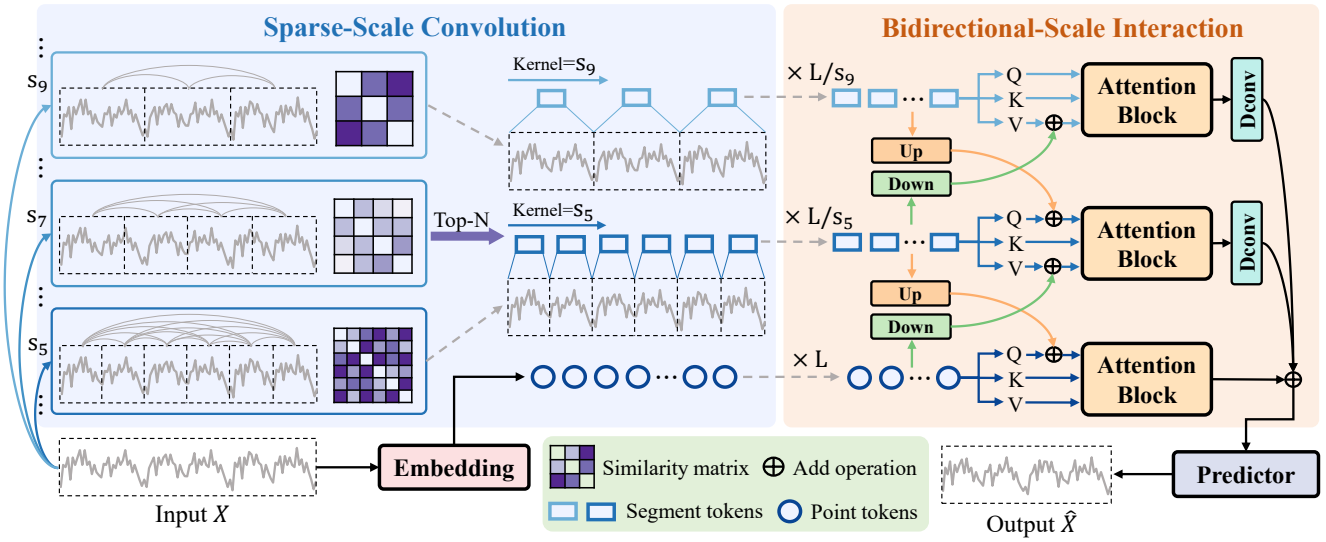

Figure 2: Overall architecture of SSformer. The Sparse-Scale Convolution (SSC) block obtains the sparse scales by evaluating the intra-scale segment similarity, then assessing and removing redundant scales. The scale-specific convolutions extract local temporal patterns. The Bidirectional-Scale Interaction (BSI) block effectively utilizes the scale correlations in both coarse-to-fine (interaction of queries) and fine-to-coarse (interaction of values) directions. The keys remains unchanged to preserve the semantic consistency. Finally, we aggregate the representations from all scales to enhance the forecasting ability.

2024b) divide time series into patches with multiple lengths to obtain segment-level input tokens and model segment-wise interaction in the attention mechanism. Nevertheless, these patch-based methods struggle to capture local patterns, which is essential for modeling complex temporal variations in time series. Moreover, their fixed patch sizes is unsuitable for all time series samples, which often distort temporal structures. Pathformer (Chen et al. 2024) incorporates multiple patch sizes corresponding to different scales into the model and leverages backpropagation to adaptively select the specific ones. But this method still introduces noise into the model from redundant scales and incurs high computational overhead.

Methodology

The main objective of multivariate time series forecasting is to predict the most probable future values by utilizing historical observations. We denote the input time series $X \in \mathbb{R}^{L \times D}$, where L is the length of the lookback sequence, D is the number of features, and x_l^d represents the value of the d-th feature at historical time step l. Our target is to fit a mapping function F: $X \rightarrow \hat{X}$, where $\hat{X} \in \mathbb{R}^{T \times D}$ and T is the length of the future time steps to be predicted.

As previously highlighted, the fundamental challenges in TSF is to tackle intricate temporal patterns. In this paper, we propose Sparse-Scale Transformer (SSformer) with Bidirectional Awareness for Time Series Forecasting, which is effective for capturing complex temporal patterns across different scales. The Sparse-Scale Convolution (SSC) block is designed to adaptively disentangle the multi-scale patterns based on the characteristics of time series and employs scale-specific convolutions to extract local dependencies. The Bidirectional-Scale Interaction (BSI) block explicitly establishes interactions across temporal scales in both coarse-to-fine and fine-to-coarse directions to enhance the forecasting capability of Transformer architecture. The overall architecture is illustrated in Fig. 2.

Sparse-Scale Convolution

Time series exhibit distinct characteristics across different scales: fine-grained scales capture intricate local patterns, whereas coarse-grained scales reveal overarching trends. The temporal variability inherent in different time series calls for scale-specific modeling tailored to their individual characteristics. Rather than using fixed-size patches or down-sampling for multi-scale time series feature extraction, we propose a Sparse-Scale Convolution (SSC) which leverages 1D convolution as a more adaptive way to capture local dependencies.

Consider the conventional multi-scale tokenization with fixed patches. We define a set of M patch size values as $P = \{p_1, p_2, ..., p_M\}$, with each patch size corresponding to a patch tokenization operation. Given the input time series $X \in \mathbb{R}^{L \times D}$, each patch tokenization operation with the patch size p_m embeds X into patches $\{X_1, X_2, ..., X_{L/p_m}\}$, where each patch $X_i \in \mathbb{R}^{p_m \times D}$. Different patch sizes produce tokens of diverse scales, where small p_m presents fine-grained scale, and large p_m presents coarse-grained scale. However, the fixed patch size set P is indiscriminately applied to all time series samples, which does not consider the inherent periodicity or temporal structure of the individuals. This leads to the mismatched scale representations which obscure meaningful patterns, introduce noise, and ultimately degrade the forecasting performance.

To motivate our approach, we begin with an important intuition in time series. Given a time series X with an inherent periodicity of T_x, segments of time series with length T_x tend to exhibit the highest degree of similarity. Conversely, if dividing X into segments of length T_x yields maximal similarity, it indicates that T_x is the pattern that most closely approximates the underlying periodicity of the time series.

For each input time series, we evaluate a set of candidate segment lengths $\mathcal{S} = \{s_1, s_2, ..., s_M\}$, and assess the intra-scale segment similarity to determine which segment lengths best align with the sequence's periodic structures. For each segment length s_m, we divide X into S (with $S = L/s_m$) segments as $\{X_1, X_2, ..., X_S\}$, where each segment $X_i \in \mathbb{R}^{s_m \times D}$ contains s_m time steps. Notably, to make segment comparisons meaningful and invariant to amplitude, each segment X_i is individually normalized at the time dimension. We then compute two pair-wise distance matrice: a cosine similarity matrix $C \in \mathbb{R}^{S \times S}$ and and an Euclidean distance matrix $E \in \mathbb{R}^{S \times S}$, defined as:

$$C_{i,j} = \frac{X_i \cdot X_j}{\|X_i\|_2 \|X_j\|_2}, \tag{1}$$

$$E_{i,j} = \sqrt{2 - 2C_{i,j}}. \tag{2}$$

To aggregate the overall dissimilarity among segments, we calculate the mean of the non-diagonal elements in the Euclidean matrix:

$$\bar{e} = \frac{1}{S(S-1)} \sum_{i \neq j} E_{i,j}. \tag{3}$$

where $S(S-1)$ is the number of non-diagonal elements. Finally, we define the similarity score of segment length s_m as the inverse of average pair-wise distance:

$$\sigma(s_m) = \frac{1}{1 + \bar{e}}. \tag{4}$$

This score reflects the similarity level among segments with length s_m, a higher $\sigma(s_m)$ indicates that segments are more shape-consistent, suggesting that s_m aligns well with a latent periodicity in the time series. We repeat this process over a candidate length set $\mathcal{S} = \{s_1, s_2, ..., s_M\}$ and select the top-N using $\text{Max}(\cdot)$ function with highest similarity scores:

$$\mathcal{S}^* = \text{Max}(\{\sigma(s_1), \sigma(s_2), ..., \sigma(s_M)\}, N). \tag{5}$$

where $\mathcal{S}^* = \{s_1^*, s_2^*, ..., s_N^*\}(s_i^* < s_{i+1}^*)$ is the optimal set of segment length, which is used to generate the tokens of corresponding scales. This process allows us to adaptively determine the most representative scales from fine to coarse granularity, tailored to the temporal structure of each input time series.

Once the optimal set $\mathcal{S}^* = \{s_1^*, s_2^*, ..., s_N^*\}$ is identified, we treat them as kernel sizes to construct a set of scale-specific 1D convolutional filters. Each convolutional filter is adaptively designed to capture local temporal dependencies within its corresponding receptive field. Formally, for each specific scale $s_i^* \in \mathcal{S}^*$, we perform a 1D convolution operation $\text{Conv}_{s_i^*}(\cdot)$ with the kernel size s_i^* and the stride s_i^* to extract local patterns:

$$H_i = \text{ReLU}(\text{Conv}_{s_i^*}(X)) \in \mathbb{R}^{u_i \times d_N}, \tag{6}$$

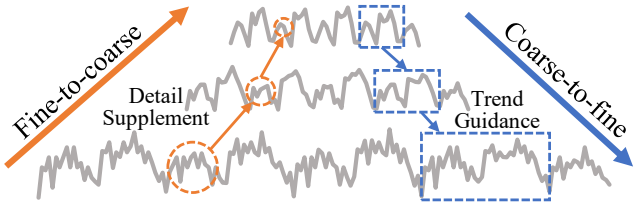

Figure 3: The description of Bidirectional-Scale Interaction. It establishes scale correlations in both coarse-to-fine and fine-to-coarse directions.

where $\text{ReLU}(\cdot)$ is the activation function, $\{H_i\}_{i=1}^N$ is the final multi-scale segment tokens, $u_i = \lceil L/s_i^* \rceil$ is the length of H_i, and d_N is the dimension of hidden state.

Bidirectional-Scale Interaction

Temporal patterns in real-world time series often span multiple scales. Utilizing the correlations across these scales is crucial for modeling complex patterns. However, many Transformer models fail to effectively establish the correlations of different scales. As shown in Fig. 3, we propose the Bidirectional-Scale Interaction (BSI) block to enable information interaction between temporal scales in both coarse-to-fine and fine-to-coarse directions, facilitating the integration of global trends and local details.

Instead of adopting a heavy encoder-decoder transformer, we introduce an encoder-only Transformer to enhance the modeling of temporal dependencies in TSF, offering improved efficiency by avoiding the computation of sequential decoding inherent in traditional Transformer architectures (Zhou et al. 2021; Feng, Huang, and Krompass 2024).

Concretely, in BSI block, we design N parallel attention blocks which aligns with the number of sparse scales. Then, we compute the self-attention of the multi-scale segment tokens $\{H_i\}_{i=1}^N$ to model the segment-wise global dependencies. To utilize complementary forecasting capabilities, we retain the tokenization scheme employed in previous Transformer-based models that embed the raw time series X into point tokens $H_0 \in \mathbb{R}^{u_0 \times d_N} (u_0 = L)$ and add a new attention block to model the point-wise global dependencies. Therefore, the incorporated multi-scale tokens are $\{H_i\}_{i=0}^N$. We adopt corresponding linear layers to get queries $\{Q_i\}_{i=0}^N$, keys $\{K_i\}_{i=0}^N$, and values $\{V_i\}_{i=0}^N$, where $Q_i, K_i, V_i \in \mathbb{R}^{u_i \times d_N}$.

Coarse-to-Fine Interaction. In coarse-to-fine interaction, we allow a coarse-grained representation H_i to guide a finer-grained representation $H_{i-1} (i > 0)$ by injecting trend-aware guidance into the fine-scale queries. To align the sequence lengths, we first upsample the coarse queries Q_i to match the finer scale u_{i-1} by using a linear layer $\text{UpLayer}_{i \to i-1}(\cdot)$, denoted as:

$$\tilde{Q}_{i \to i-1} = \text{UpLayer}_{i \to i-1}(Q_j) \in \mathbb{R}^{u_i \times d_N}. \tag{7}$$

We then incorporate the $\tilde{Q}_{i \to i-1}$ into the finer-scale queries:

$$Q'_{i-1} = Q_{i-1} + \tilde{Q}_{i \to i-1} \tag{8}$$

where Q'_{i-1} is the enriched queries.

Models		SSformer Ours		Times2D (2025)		iTransformer (2024)		MTST (2024)		TimeMixer (2024)		PatchTST (2023)		TimesNet (2023)		Scaleformer (2023)		MICN (2023)		FEDformer (2022)	
Metric		MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE
Weather	96	**0.156**	**0.204**	0.181	0.221	0.174	0.214	0.175	0.216	0.163	0.209	0.177	0.218	0.172	0.220	0.220	0.289	0.198	0.261	0.217	0.296
	192	**0.203**	**0.245**	0.228	0.260	0.221	0.254	0.219	0.255	0.208	0.250	0.225	0.259	0.219	0.261	0.341	0.385	0.239	0.299	0.276	0.336
	336	**0.250**	**0.284**	0.280	0.297	0.278	0.296	0.276	0.296	0.251	0.287	0.278	0.297	0.280	0.306	0.463	0.455	0.285	0.336	0.339	0.380
	720	**0.338**	**0.337**	0.355	0.345	0.358	0.347	0.351	0.346	0.339	0.341	0.354	0.348	0.365	0.359	0.682	0.565	0.351	0.388	0.403	0.428
Electricity	96	**0.141**	**0.238**	0.175	0.269	0.148	0.240	0.160	0.248	0.153	0.247	0.181	0.270	0.168	0.272	0.182	0.297	0.180	0.293	0.193	0.308
	192	**0.157**	**0.251**	0.184	0.273	0.162	0.253	0.171	0.263	0.166	0.256	0.188	0.274	0.184	0.289	0.188	0.300	0.189	0.302	0.201	0.315
	336	**0.174**	**0.268**	0.199	0.288	0.178	0.269	0.188	0.281	0.185	0.277	0.204	0.293	0.198	0.300	0.210	0.324	0.198	0.312	0.214	0.329
	720	**0.209**	**0.303**	0.235	0.319	0.225	0.317	0.230	0.315	0.225	0.310	0.246	0.324	0.220	0.320	0.232	0.339	0.217	0.330	0.246	0.355
ETTm1	96	**0.318**	**0.355**	0.330	0.364	0.334	0.368	0.323	0.360	0.320	0.357	0.329	0.367	0.338	0.375	0.355	0.398	0.365	0.387	0.379	0.419
	192	**0.360**	**0.379**	0.361	0.382	0.377	0.391	0.363	0.386	0.361	0.381	0.367	0.385	0.374	0.387	0.428	0.455	0.403	0.408	0.426	0.441
	336	0.393	**0.402**	0.407	0.417	0.426	0.420	0.393	0.406	**0.390**	0.404	0.399	0.410	0.410	0.411	0.524	0.487	0.436	0.431	0.445	0.459
	720	**0.449**	0.440	0.472	0.450	0.491	0.459	0.453	0.441	0.454	0.441	0.454	**0.439**	0.478	0.450	0.558	0.517	0.489	0.462	0.543	0.490
ETTm2	96	**0.171**	**0.254**	0.174	0.262	0.180	0.264	0.174	0.256	0.175	0.258	0.175	0.259	0.187	0.267	0.182	0.275	0.197	0.296	0.203	0.287
	192	**0.235**	0.302	0.241	0.303	0.250	0.309	0.243	0.302	0.237	**0.299**	0.241	0.302	0.249	0.309	0.251	0.318	0.284	0.361	0.269	0.328
	336	0.307	**0.338**	0.301	0.342	0.311	0.348	0.301	0.340	**0.298**	0.340	0.305	0.343	0.321	0.351	0.340	0.375	0.381	0.429	0.325	0.366
	720	0.403	**0.395**	0.397	0.396	0.412	0.407	0.397	**0.395**	**0.391**	0.396	0.402	0.400	0.408	0.403	0.435	0.433	0.549	0.522	0.421	0.415
ETTh1	96	**0.373**	**0.393**	0.376	0.395	0.386	0.405	0.376	**0.393**	0.375	0.400	0.414	0.419	0.384	0.402	0.396	0.440	0.426	0.446	0.395	0.424
	192	0.432	0.431	0.435	0.423	0.441	0.436	**0.429**	0.422	**0.429**	**0.421**	0.460	0.445	0.436	0.429	0.434	0.460	0.454	0.464	0.469	0.470
	336	0.483	0.454	0.457	0.439	0.487	0.458	**0.444**	**0.436**	0.484	0.458	0.501	0.466	0.491	0.469	0.462	0.476	0.493	0.487	0.530	0.499
	720	0.519	0.495	**0.465**	**0.464**	0.503	0.491	0.469	0.466	0.498	0.482	0.500	0.488	0.521	0.500	0.494	0.500	0.526	0.526	0.598	0.544
ETTh2	96	**0.281**	**0.332**	0.290	0.338	0.297	0.349	0.293	0.342	0.289	0.341	0.302	0.348	0.340	0.374	0.364	0.407	0.372	0.424	0.358	0.397
	192	**0.367**	**0.385**	0.379	0.393	0.380	0.400	0.371	0.389	0.372	0.392	0.388	0.400	0.402	0.414	0.466	0.458	0.492	0.492	0.429	0.439
	336	0.394	0.418	**0.377**	**0.407**	0.428	0.432	0.382	0.409	0.386	0.414	0.426	0.433	0.452	0.452	0.479	0.476	0.607	0.555	0.496	0.487
	720	**0.395**	**0.420**	0.412	0.433	0.427	0.445	0.415	0.434	0.412	0.434	0.431	0.446	0.462	0.468	0.487	0.492	0.824	0.655	0.463	0.474
Solar	96	0.197	**0.236**	0.236	0.283	0.203	0.237	0.236	0.253	**0.189**	0.259	0.234	0.286	0.250	0.292	0.271	0.331	0.257	0.325	0.242	0.342
	192	**0.222**	0.268	0.278	0.310	0.233	**0.261**	0.257	0.284	**0.222**	0.283	0.267	0.310	0.296	0.318	0.288	0.332	0.278	0.354	0.285	0.380
	336	0.245	**0.271**	0.291	0.321	0.248	0.273	0.276	0.309	**0.231**	0.292	0.290	0.315	0.319	0.330	0.358	0.412	0.298	0.375	0.282	0.376
	720	0.247	0.287	0.290	0.317	0.249	**0.275**	0.289	0.321	**0.223**	0.285	0.289	0.317	0.338	0.337	0.377	0.437	0.299	0.379	0.357	0.427
Exchange	96	0.085	**0.200**	**0.080**	0.207	0.086	0.206	**0.080**	0.205	0.086	0.204	0.088	0.205	0.107	0.234	0.109	0.240	0.102	0.235	0.148	0.278
	192	**0.171**	**0.296**	0.179	0.299	0.177	0.299	0.178	0.299	0.176	0.298	0.176	0.299	0.226	0.344	0.241	0.353	0.172	0.316	0.271	0.315
	336	0.348	0.429	0.325	0.410	0.331	0.417	0.318	0.405	0.345	0.426	0.301	**0.397**	0.367	0.448	0.471	0.508	**0.272**	0.407	0.460	0.427
	720	0.843	0.715	0.848	0.688	0.847	0.691	0.845	0.683	0.848	0.692	0.901	0.714	0.964	0.746	1.259	0.865	**0.714**	**0.658**	1.195	0.695
Traffic	96	0.420	0.288	0.521	0.356	**0.395**	**0.268**	0.422	0.271	0.462	0.285	0.462	0.295	0.593	0.321	0.564	0.351	0.577	0.350	0.587	0.366
	192	0.434	0.298	0.505	0.341	**0.417**	**0.276**	0.437	0.281	0.473	0.296	0.466	0.296	0.617	0.336	0.570	0.349	0.589	0.356	0.604	0.373
	336	0.468	0.301	0.528	0.355	**0.433**	**0.283**	0.451	0.285	0.498	0.296	0.482	0.304	0.629	0.336	0.576	0.349	0.594	0.358	0.621	0.383
	720	0.487	0.309	0.566	0.373	**0.467**	**0.302**	0.490	0.309	0.506	0.313	0.514	0.322	0.640	0.350	0.602	0.360	0.613	0.361	0.626	0.382

Table 1: Multivariate time series forecasting results. A lower MSE or MAE indicates a better prediction. We fix the lookback length $L = 96$ and set the prediction length $T \in \{96, 192, 336, 720\}$ for all experiments. The best results are highlighted in **bold** and the second-best are underlined.

Fine-to-Coarse Interaction. In the fine-to-coarse direction, we integrate detail-rich information from a fine-grained representation H_{i-1} into a coarser-grained representation H_i to supplement lost details. Specifically, we downsample the fine-grained values V_{i-1} to match the coarser scale u_i by using a linear layer $\text{DownLayer}_{i-1\to i}(\cdot)$, denoted as:

$$\tilde{V}_{i-1\to i} = \text{DownLayer}_{i-1\to i}(V_{i-1}) \in \mathbb{R}^{u_i \times d_N}. \quad (9)$$

We then enhance the coarser values V_j as:

$$V_i' = V_i + \tilde{V}_{i-1\to i} \quad (10)$$

where V_i' is the enriched values.

Subsequently, the enriched queries and values is used to compute attention with its corresponding keys:

$$\text{Attn}_i = \text{Softmax}(\frac{Q_i' K^T}{\sqrt{d_N}})V_i' \quad (11)$$

Following the general formulation, the encoder also include layer normalization and feed forward networks with residual connections. Afterwards it generates the final multi-scale representations denoted as $\{Z_i\}_{i=0}^N$.

Finally, we aggregate the refined representations from all scales and use it to predict the future time series:

$$\hat{X} = \text{Predictor}(\sum_{i=0}^{N} Z_i) \quad (12)$$

where $\text{Predict}(\cdot)$ is a linear layer to obtain the predictions.

The Bidirectional-Scale Interaction block explicitly models the interaction of different scales, which enhances the extraction of multi-scale temporal patterns, leading to more accurate forecasting performance.

Experiment

Experimental Setup

Datasets. We conduct experiments on six real-world benchmark datasets, including Weather, Electricity, ETT (ETTm1, ETTm2, ETTh1, ETTh2), Solar, Exchange, and Traffic.

Baselines. We select nine representative time series forecasting methods served as our baselines, including (1) Transformer-based methods: Times2D (Nematirad, Pahwa, and Natarajan 2025), iTransformer (Liu et al. 2024),

Model	Metric	Weather				Electricity				ETTm2				ETTh2			
		96	192	336	720	96	192	336	720	96	192	336	720	96	192	336	720
SSformer	MSE	**0.156**	**0.203**	**0.250**	**0.338**	**0.141**	**0.157**	**0.174**	**0.209**	**0.171**	**0.235**	**0.307**	**0.403**	**0.281**	**0.367**	**0.384**	**0.395**
	MAE	**0.204**	**0.245**	**0.284**	**0.337**	**0.238**	**0.251**	**0.268**	**0.303**	**0.254**	**0.302**	**0.338**	**0.395**	**0.332**	**0.385**	**0.411**	**0.420**
W/O-SS	MSE	0.183	0.229	0.279	0.356	0.166	0.184	0.189	0.234	0.185	0.253	0.319	0.415	0.298	0.383	0.396	0.401
	MAE	0.218	0.266	0.302	0.360	0.254	0.278	0.289	0.319	0.268	0.323	0.354	0.416	0.354	0.402	0.428	0.443
W/O-Conv	MSE	0.175	0.216	0.255	0.357	0.152	0.174	0.187	0.220	0.179	0.242	0.315	0.412	0.287	0.378	0.386	0.397
	MAE	0.211	0.255	0.302	0.352	0.245	0.258	0.283	0.322	0.261	0.312	0.347	0.404	0.342	0.397	0.428	0.435
W/O-CTF	MSE	0.162	0.209	0.264	0.350	0.149	0.168	0.179	0.217	0.175	0.238	0.312	0.408	0.283	0.374	0.389	0.399
	MAE	0.209	0.249	0.295	0.345	0.248	0.263	0.273	0.312	0.259	0.305	0.343	0.401	0.338	0.393	0.421	0.425
W/O-FTC	MSE	0.158	0.213	0.261	0.339	0.145	0.164	0.182	0.210	0.174	0.238	0.311	0.409	0.283	0.376	0.385	0.398
	MAE	0.210	0.252	0.286	0.342	0.246	0.266	0.274	0.318	0.259	0.306	0.401	0.399	0.333	0.387	0.415	0.423

Table 2: Ablation study of designed components. W/O-SS, W/O-Conv, W/O-CTF, and W/O-FTC represent removing the sparse scale, convolution operation, coarse-to-fine interaction, and fine-to-coarse interaction.

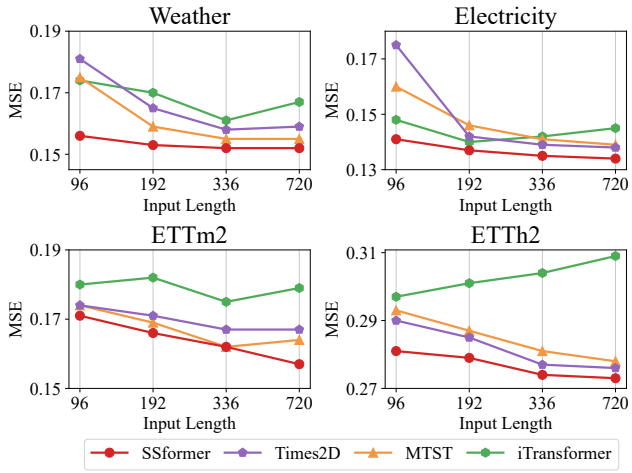

Figure 4: The MSE performance on four datasets with the lookback length $L \in \{96, 192, 336, 720\}$ and fixed prediction length $T = 96$.

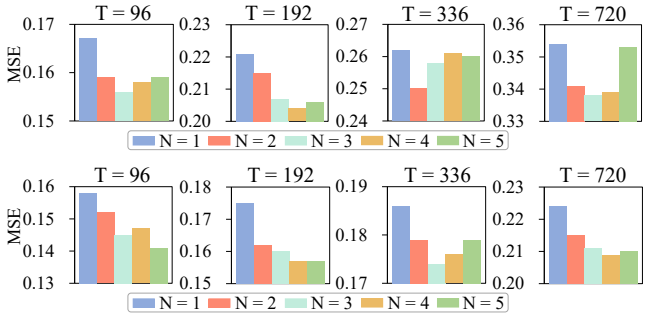

Figure 5: The MSE performance with lookback length $L = 96$ and prediction length $T \in \{96, 192, 336, 720\}$ varies with N on Weather (upper) and Electricity (lower) dataset.

MTST (Zhang et al. 2024a), PatchTST (Nie et al. 2023), Scaleformer (Shabani et al. 2023), and FEDformer (Zhou et al. 2022). (2) CNN-based methods: TimesNet (Wu et al. 2023a) and MICN (Wang et al. 2023a). (3) MLP-based method: TimeMixer (Wang et al. 2024).

Implementation Details. Notably, experimental results reported by the above mentioned baselines cannot be compared directly due to different input and prediction lengths. To ensure fair comparisons, we align the input length of all baselines that are set to $L = 96$ and the prediction lengths that are set to $T \in \{96, 192, 336, 720\}$. For training, we utilize the L2 loss and the Adam optimizer with an initial learning rate of 1×10^{-4}. Following the baselines, we also use Mean Squared Error (MSE) and Mean Absolute Error (MAE) as the evaluation matrics. All experiments are conducted using PyTorch on a NVIDIA A40 48GB GPU and are repeated five times for consistency.

Main Results

The forecasting results are presented in Tab. 1. SSformer stands out with the best performance in 41 cases and the second-best in 15 cases out of the overall 72 cases. Notably, SSformer outperforms strong linear model TimeMixer and CNN-based models TimesNet with 6.6% and 11.5% reduction respectively in MSE on Electricity dataset. This reveals the promising capability of Transformer architecture in advancing time series forecasting. Compared with the general Transformer models Times2D and iTransformer, SSformer demonstrates a significant improvement on four ETT datasets where multitudinous variations frequently overlap and interact. Moreover, compared with the multi-scale Transformer model MTST, SSformer exhibits better performances, with 7.1% and 14.0% reduction in MSE on Weather and Solar datasets. This demonstrates that the scale sparsity and bidirectional interaction modeling is beneficial for enhancing the forecasting performance.

Model Analysis

Ablation study on designed components. We perform ablation studies on four datasets to assess the impact of each SSformer component. Tab. 2 presents the results.

We conduct experiments by removing the sparse scale operation (denoted as W/O-SS). As shown in the results, it leads to an significant increase in both MSE and MAE, with particularly rises of 13.5% and 7.5% on the Electricity dataset. This result further highlights the importance of scale

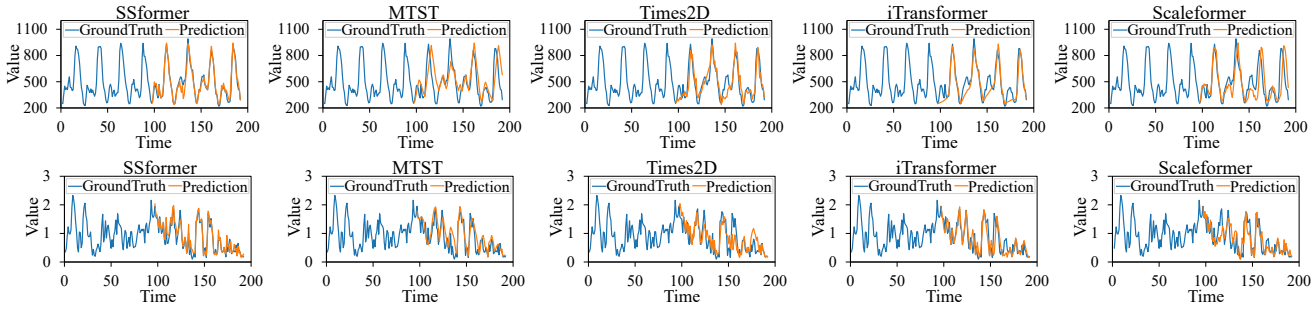

Figure 6: Visualization of input-96-predict-96 results on the Electricity (upper) and Weather (lower) datasets.

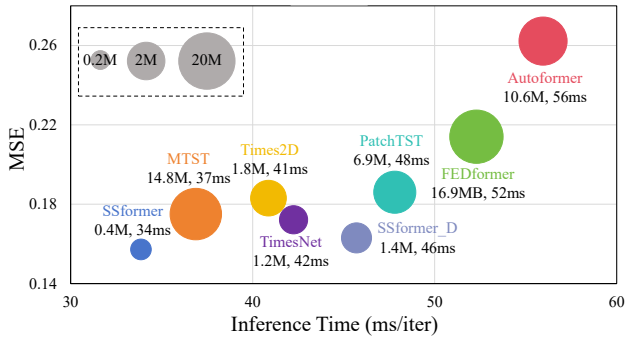

Figure 7: Inference time (ms/iter), model parameters (M), and MSE comparisons on Weather dataset. The lookback length and prediction length are all set to 96. SSformer_D is the model incorporating a decoder into SSformer.

sparsity in time series forecasting.

To exhibit the utilization of convolution operations, we replace the convolutional tokenization method in SSC with patching (W/O-Conv). As a result, this leads to a noticeable performance degradation, with an increase of 5.9% and 4.5% in MSE and MAE respectively on the Weather dataset, which confirm that convolutional operations play a vital role in learning temporal dependencies.

To show the effectiveness of BSI, we remove the coarse-to-fine interaction (W/O-CTF) and fine-to-coarse interaction (W/O-FTC) respectively. The results show that removing these two interactions decreases performance. Therefore, taking full advantage of scale correlations is conducive to boost the forecasting capability.

Performance with different lookback lengths. We set lookback lengths $L = \{96, 192, 336, 720\}$ while fixing the prediction length $T = 96$. As shown in Fig. 4, SSformer consistently achieves the lowest MSE across all lookback lengths, demonstrating its superior capability in capturing long-term dependencies and local patterns. Notably, as the lookback length increases, SSformer steadily improves its forecasting accuracy, whereas other models (e.g., iTransformer and Times2D) tend to suffer from performance degradation or saturation. This improvement arises because the Sparse-Scale Convolution of SSformer is effective to filter noise by focusing on the most representative temporal patterns and discarding uninformative noise. Moreover, SSformer achieves superior performance even when the lookback length is as short as 96, while other models typically require 192 or 336. This highlights SSformer's strong ability to extract informative patterns from limited lookback length.

Sensitivity of hyperparameters. To investigate the sensitivity of the hyperparameter N, we conduct experiments on the Weather and Electricity datasets, with results showed in Fig. 5. For the Weather dataset, performance improves as N increases from 1 to 3, with the best results at $N = 3$. Further increasing N to 4 or 5 brings no additional gains. For the Electricity dataset, $N = 4$ and $N = 5$ achieve the best overall performance. These results demonstrate that imposing sparsity on scales is beneficial, while overly large N may dilute important patterns.

Visualization of prediction results. To provide a intuitional comparison, we give the prediction showcases of SSformer and four competitive baselines on Weather and Electricity datasets in Fig. 6. Our observations reveal that SSformer produces predictions that more closely align with the ground truth, particularly in terms of periodic variations and local fluctuations. Moreover, compared with the multi-scale method MTST and Scaleformer that tend to overfit sharp or noisy fluctuations, SSformer yields more robust predictions that better reflect the underlying temporal patterns.

Efficiency Analysis. We conduct a comparison of inference time and model parameters on the Weather dataset, using official model settings and a consistent batch size. The results shown in Fig. 7 demonstrate that SSformer outperforms other models in inference speed and model parameters, which can be attributed to its encoder-only architecture and the inherent parallelism of multi-scale processing.

Conclusion

In this paper, we propose the Sparse-Scale Transformer (SSformer) with Bidirectional Awareness for Time Series Forecasting to address the overfitting problem in multi-scale modeling by remove the noisy scale representations. We further establish the bidirectional scale interaction to enhance the pattern extraction. Experimental results on various datasets demonstrate that SSformer achieves state-of-the-art forecasting performance and exhibits superior efficiency.

Acknowledgments

Ying Liu and Richang Hong are partially supported by The National Science Foundation of China Project (No. 92467302). Wenbo Hu is partially supported by The National Science Foundation of China Project (No. 62306098).

References

Box, G. E.; and Jenkins, G. M. 1968. Some recent advances in forecasting and control. *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, 17: 91–109.

Cai, W.; Liang, Y.; Liu, X.; Feng, J.; and Wu, Y. 2024. Msgnet: Learning multi-scale inter-series correlations for multivariate time series forecasting. In *Proceedings of the AAAI Conference on Artificial Intelligence*.

Challu, C.; Olivares, K. G.; Oreshkin, B. N.; Ramirez, F. G.; Canseco, M. M.; and Dubrawski, A. 2023. Nhits: Neural hierarchical interpolation for time series forecasting. In *Proceedings of the AAAI Conference on Artificial Intelligence*.

Chen, P.; Zhang, Y.; Cheng, Y.; Shu, Y.; Wang, Y.; Wen, Q.; Yang, B.; and Guo, C. 2024. Pathformer: Multi-scale Transformers with Adaptive Pathways for Time Series Forecasting. In *International Conference on Learning Representations*.

Cheng, D.; Yang, F.; Xiang, S.; and Liu, J. 2022. Financial time series forecasting with multi-modality graph neural network. *Pattern Recognition*, 121: 108218.

Cleveland, R. B.; Cleveland, W. S.; McRae, J. E.; Terpenning, I.; et al. 1990. STL: A seasonal-trend decomposition. *J. off. Stat*, 6: 3–73.

Feng, C.; Huang, L.; and Krompass, D. 2024. General time transformer: an encoder-only foundation model for zero-shot multivariate time series forecasting. In *Proceedings of the ACM International Conference on Information and Knowledge Management*.

Gao, J.; Chen, Y.; Hu, W.; and Zhang, D. 2023. An adaptive deep-learning load forecasting framework by integrating transformer and domain knowledge. *Advances in Applied Energy*, 10: 100142.

Hu, Y.; Liu, P.; Zhu, P.; Cheng, D.; and Dai, T. 2025. Adaptive multi-scale decomposition framework for time series forecasting. In *Proceedings of the AAAI Conference on Artificial Intelligence*.

Huang, Q.; Shen, L.; Zhang, R.; Cheng, J.; Ding, S.; Zhou, Z.; and Wang, Y. 2024. Hdmixer: Hierarchical dependency with extendable patch for multivariate time series forecasting. In *Proceedings of the AAAI Conference on Artificial Intelligence*.

Kieu, D.; Kieu, T.; Han, P.; Yang, B.; Jensen, C. S.; and Le, B. 2024. TEAM: Topological evolution-aware framework for traffic forecasting. *Proceedings of the VLDB Endowment*, 18: 265–278.

Lai, G.; Chang, W.-C.; Yang, Y.; and Liu, H. 2018. Modeling Long- and Short-Term Temporal Patterns with Deep Neural Networks. In *The International ACM SIGIR Conference on Research & Development in Information Retrieval*.

Lazcano, A.; Herrera, P. J.; and Monge, M. 2023. A combined model based on recurrent neural networks and graph convolutional networks for financial time series forecasting. *Mathematics*, 11: 224.

Li, B.; Zheng, T.; Jing, Y.; Jiao, C.; Xiao, T.; and Zhu, J. 2022a. Learning multiscale transformer models for sequence generation. In *International Conference on Machine Learning*.

Li, Y.; Wu, C.-Y.; Fan, H.; Mangalam, K.; Xiong, B.; Malik, J.; and Feichtenhofer, C. 2022b. Mvitv2: Improved multiscale vision transformers for classification and detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*.

Liu, H.; Dong, Z.; Jiang, R.; Deng, J.; Deng, J.; Chen, Q.; and Song, X. 2023. Spatio-temporal adaptive embedding makes vanilla transformer sota for traffic forecasting. In *Proceedings of the ACM International Conference on Information and Knowledge Management*.

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*.

Nawrot, P.; Tworkowski, S.; Tyrolski, M.; Kaiser, L.; Wu, Y.; Szegedy, C.; and Michalewski, H. 2022. Hierarchical Transformers Are More Efficient Language Models. In *Findings of the Association for Computational Linguistics*.

Nematirad, R.; Pahwa, A.; and Natarajan, B. 2025. Times2d: Multi-period decomposition and derivative mapping for general time series forecasting. In *Proceedings of the AAAI Conference on Artificial Intelligence*.

Nguyen, T.; Shah, R.; Bansal, H.; Arcomano, T.; Maulik, R.; Kotamarthi, R.; Foster, I. T.; Madireddy, S.; and Grover, A. 2024. Scaling transformer neural networks for skillful and reliable medium-range weather forecasting. In *Advances in Neural Information Processing Systems*.

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*.

Qiu, X.; Hu, J.; Zhou, L.; Wu, X.; Du, J.; Zhang, B.; Guo, C.; Zhou, A.; Jensen, C. S.; Sheng, Z.; and Yang, B. 2024. TFB: Towards Comprehensive and Fair Benchmarking of Time Series Forecasting Methods. *Proceedings of the VLDB Endowment*, 17: 2363–2377.

Shabani, A.; Abdi, A.; Meng, L.; and Sylvain, T. 2023. Scaleformer: Iterative Multi-scale Refining Transformers for Time Series Forecasting. In *International Conference on Learning Representations*.

Shang, Z.; Chen, L.; Wu, B.; and Cui, D. 2024. Ada-MSHyper: Adaptive Multi-Scale Hypergraph Transformer for Time Series Forecasting. In *Advances in Neural Information Processing Systems*.

Shao, Z.; Wang, F.; Xu, Y.; Wei, W.; Yu, C.; Zhang, Z.; Yao, D.; Sun, T.; Jin, G.; Cao, X.; Cong, G.; Jensen, C. S.; and Cheng, X. 2025. Exploring Progress in Multivariate Time Series Forecasting: Comprehensive Benchmarking and Heterogeneity Analysis. *IEEE Transactions on Knowledge and Data Engineering*, 37: 291–305.

Shen, L.; Li, Z.; and Kwok, J. 2020. Timeseries Anomaly Detection using Temporal Hierarchical One-Class Network. In *Advances in Neural Information Processing Systems*.

Stefenon, S. F.; Seman, L. O.; Mariani, V. C.; and Coelho, L. d. S. 2023. Aggregating prophet and seasonal trend decomposition for time series forecasting of Italian electricity spot prices. *Energies*, 16: 1371.

Wang, H.; Peng, J.; Huang, F.; Wang, J.; Chen, J.; and Xiao, Y. 2023a. MICN: Multi-scale Local and Global Context Modeling for Long-term Series Forecasting. In *International Conference on Learning Representations*.

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, W.; Chen, W.; Qiu, Q.; Chen, L.; Wu, B.; Lin, B.; He, X.; and Liu, W. 2023b. Crossformer++: A versatile vision transformer hinging on cross-scale attention. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 46: 3123–3136.

Wu, H.; Hu, T.; Liu, Y.; Zhou, H.; Wang, J.; and Long, M. 2023a. TimesNet: Temporal 2D-Variation Modeling for General Time Series Analysis. In *International Conference on Learning Representations*.

Wu, H.; Zhou, H.; Long, M.; and Wang, J. 2023b. Interpretable weather forecasting for worldwide stations with a unified deep model. *Nature Machine Intelligence*, 5: 602–611.

Zhang, P.; Dai, X.; Yang, J.; Xiao, B.; Yuan, L.; Zhang, L.; and Gao, J. 2021. Multi-scale vision longformer: A new vision transformer for high-resolution image encoding. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*.

Zhang, Y.; Ma, L.; Pal, S.; Zhang, Y.; and Coates, M. 2024a. Multi-resolution time-series transformer for long-term forecasting. In *International Conference on Artificial Intelligence and Statistics*.

Zhang, Y.; Wu, R.; Dascalu, S. M.; and Jr., F. C. H. 2024b. Multi-Scale Transformer Pyramid Networks for Multivariate Time Series Forecasting. *IEEE Access*, 12: 14731–14741.

Zhao, Y.; Luo, C.; Zha, Z.-J.; and Zeng, W. 2021. Multi-scale group transformer for long sequence modeling in speech separation. In *Proceedings of the International Conference on International Joint Conferences on Artificial Intelligence*.

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 *Proceedings of the 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*.