Multi-scale Period-dependent Transformer for Time Series Forecasting

Jiatian Pi [1, ⋆] , Chenyue Wang [1, ⋆], Kanlun Tan [2], Xin Wang [3], and Qiao Liu [1, ⋆⋆]

[1] National Center for Applied Mathematics, Chongqing Normal University, Chongqing, China
[2] State Key Laboratory of Intelligent Vehicle Safety Technology, Chongqing, China
[3] Chongqing Changan Automobile Company Limited, Chongqing, China
⋆ Equal Contribution
NCAMCwcy_edu@163.com

Abstract. The periodicity of the time series is useful in improving the performance of forecasting models by revealing long-term trends, seasonal variations and oscillatory phenomena. Existing methods usually use a single-scale periodicity assumption at a certain fixed stage, which is uncoupled from the inherent multi-scale and continuous nature of the periodicity. This leads to a bottleneck in the exploitation of the periodic property of the time series by these methods. It limits the ability of the models to explore the underlying periodic information for capturing reliable dependencies and constructing them efficiently in forecasts.
To this end, in this paper, we make full use of the multi-scale information of time-series periodicity and construct a continuous periodic relational interaction at multiple different stages. By modeling dependencies and feature aggregation at the sub-sequence level, we are able to break the bottleneck of underutilization of periodic information. Specifically, we first extract the inherent stationary periodic measurement of sequence data and embed the multi-layers period pattern to model seasonal regularity. Second, to capture the long-range periodicity correlation, we propose a novel attention mechanism that performs convergence of representation under the predictive paradigm with efficient sparse filtering based on periodic segments. Third, we implement the sequence decomposition with multi-period scale to separate precisely tendency and seasonality. Therefore, the intrinsic patterns of the time series can be reasonably deciphered and analyzed respectively. Extensive experimental results on five benchmarks show that our method achieves favorable results, especially on the significantly periodic data.

Keywords: Time-series forecasting, Periodicity, Transformer.

1 Introduction

Time series prediction receives much attention due to the wide range of applications, such as business cycles [1], energy consumption [2], weather forecasting [3], and power supply pressure [4]. It has a main feature that the forecast time span is very long, such as several months or years, and it contains significant periodicity characteristics [5, 6].

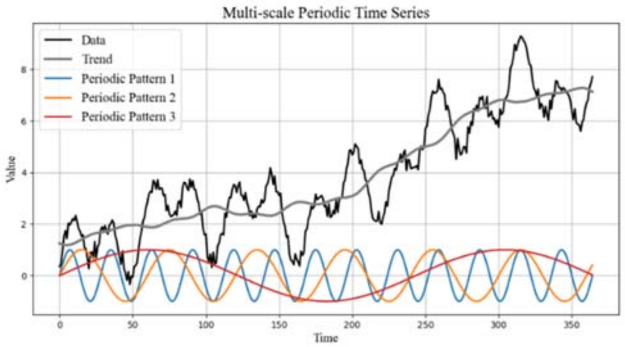


Fig. 1. Multi-scale periodic patterns of the time series data. Different periodic patterns denote different scale information.

As shown in Fig.1, time series information can be decomposed into multiple periodic patterns with different scales which is critical to the analysis and modeling of known time series. For example, by measuring multiple periodic patterns, long-term stable trends with spanning years or decades can be discovered on a large scale, and seasonal variation repeating annually or monthly becomes more evident. This allows us to interpret the recurring effects like holiday sales or weather changes. Similarly, fluctuation phenomena, e.g., rising and falling business cycles, can also be identified in time series that span long enough. In summary, the cyclical multi-scale nature of time series data provides key insight into data patterns that change over time.

In recent years, deep learning based time series prediction methods [7,8,9] have made remarkable progress. Mainly because they have powerful global information aggregation capabilities. Especially, the Transformer architecture [10,11] has been introduced into time series prediction recently, deep learning based methods have achieved further improvement. Because the Transformer has more powerful long-range dependencies modeling capabilities. Unfortunately, using the original Transformer directly for long-term series forecasting will bring extremely high time complexity and memory consumption. To improve inference speed, LogTrans [12] designs a sparse attention module and Informer[13] uses a generative style decoder to replace the step-by-step method. More recently works [14,15] began to explore how to use periodicity to further improve the performance of time series forecasting. For example, Fedformer introduces a sequence decomposition and frequency domain computation to capture potential periodic. Autoformer proposes a sub-sequence-level information aggregation for modeling periodicity. However, these methods exploit periodic information in a single-scale manner only at a certain stage of time series forecasting, which is uncoupled from the inherent multi-scale and continuous nature of the periodic. Consequently, these methods cannot capture the multi-scale periodic dependence and model the internal periodicity patterns of time series data effectively.

To solve this problem, we propose a multi-scale period-dependent Transformer framework for long-term time series forecasting. The proposed method fully exploits periodic information by performing feature aggregation and dependency modeling at the sub-sequence level. Moreover, our method can capture continuous periodic relational interactions by exploiting periodicity information at multiple different stages. Specifically, our method consists of three kinds of periodic-dependent modules: the Multi-Layers Periodic Stacked Embedding, the Multi-Head Foresight Attention, and the Multi-Scale Polishing Sequence Decomposition. To model the period-based multiple seasonal patterns of the time series, we first propose a Multi-Layers Periodic Stacked Embedding that can make the initial inputs obtain the significance projection under each seasonal regularity. Then, to capture the long-range periodic dependence, we design a novel information convergence, named Multi-Head Foresight Attention. It not only uses periodic measurement to choose the perception field but also differentiates the query and key in the attention mechanism. Meanwhile, we provide a corresponding sparse screening mechanism, which emphasizes the difference between stable trends and seasonal fluctuations. It also conforms to the characteristics of time series and reduces the quadratic complexity effectively. Finally, to separate the stable trend term and the periodic seasonal term inherent in the time series, we employ a deep submodule Multi-Scale Polishing Sequence Decomposition inside the architecture. This enables the model to obtain progressive decomposition capabilities based on multi-period planning. Our method achieves favorable accuracy on the five time series forecasting datasets, especially on significantly periodic data.

The contributions are summarized as follows:

– We propose a multi-scale periodic-dependent long-term time series forecasting method that makes full use of periodic information in multiple different stages to improve feature representation capabilities.

– We design a Multi-Layers Periodic Stacked Embedding model, a Multi-Head Foresight Attention, and a Multi-Scale Polishing Sequence Decomposition. These three modules utilize periodic information from different perspectives and form a continuous framework of periodic relationship interaction.

– We conduct extensive experiments on five benchmarks and the results demonstrate that our method achieves favorable performance.

2 Methodology

2.1 Overview of algorithm

The encoder-decoder architecture is designed as the basic architecture of our model, and it has been used in various areas of deep learning with effective performance. It extracts the information of the input representation X^t and turns it into a hidden representation H^t, then the decoder is responsible for decoding Y^t from the hidden representation. In this process, each sub-module works closely together to continuously disentangle the trend and season of the series and extract the complex correlation information of the variables within the time series to model period-dependent representation.

In the input phase of encoder and decoder, we embed the temporal feature data, the relevant covariance, and the location information required by the attention mechanism, respectively. This is based on extensive transformer-based related work that has been proven effective. Inside the Multi-Layers Periodic Stacked Embedding, we perform periodic calculation by frequency domain analysis to capture the periodicity dependence of the sequence and transmit it to each sub-module, then we embed the multi-layers period encoded in the model input. Next, we construct a Multi-Head Foresight Attention mechanism designed to capture long-range temporal aggregation of time series local representation. To enable the model to progressively decompose seasonality and tendency, we add the Multi-Scale Polishing Sequence Decomposition module and string it across multiple sub-modules of the overall architecture. Two identical feed-forward network structures composed of one-dimensional convolution, linear layer, and norm are used to model the trend term and season term, respectively, then the output is combined. Finally, we use a generative decoder to obtain extensive sequence outputs, thus avoiding the spread of cumulative errors in the inference phase.

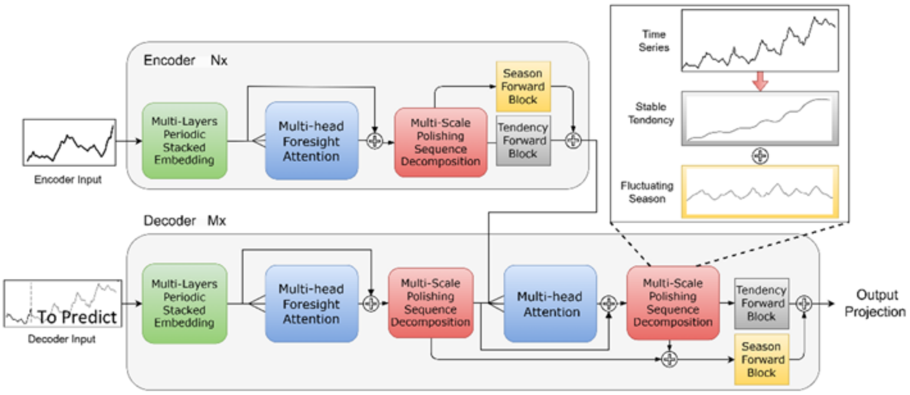

Encoder Nx
Encoder Input
Multi-Layers Periodic Stacked Embedding
Multi-head Foresight Attention
Multi-Scale Polishing Sequence Decomposition
Season Forward Block
Tendency Forward Block
Time Series
Stable Tendency
Fluctuating Season
Decoder Mx
To Predict
Decoder Input
Multi-Layers Periodic Stacked Embedding
Multi-head Foresight Attention
Multi-Scale Polishing Sequence Decomposition
Multi-head Attention
Multi-Scale Polishing Sequence Decomposition
Tendency Forward Block
Season Forward Block
Output Projection

Fig. 2. Framework of the proposed method.

2.2 Multi-Layers Periodic Stacked Embedding

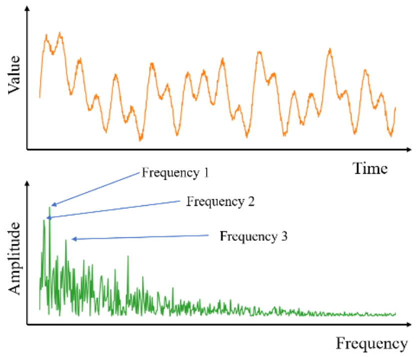


Fig. 3. Multi-scale periodic patterns of the time series data. Different periodic patterns denote different scale information.

As shown in Fig.3, periodic patterns are temporal variations that occur in series and they reveal recurring regularity and underlying fluctuation of seasonality and provide a basis for building predictive models. By identifying cyclicality, we can use historical data to predict future cyclical changes. In order to obtain and embed the various periodic patterns in the time series, we need to first find the specific period length and significance. Technically, we switch the sequence $\mathcal{X}$ that is embedded with original values, covariables, and location information to the frequency domain for specific analysis as follows:

$$\begin{aligned} &\mathcal{A} = |FFT\ (\mathcal{X})|\ ,\ \{f_1, \cdots, f_k\} = \underset{f \in \left\{1, \cdots, \left[\frac{L}{2}\right]\right\}}{\arg \mathrm{Topk}} (\mathcal{A}), \\ &l_i = \left\lceil \frac{L}{f_i} \right\rceil, i \in \{1, \cdots, k\}, \\ &\hat{S}(l_1), \cdots, \hat{S}(l_k) = Softmax\left(\mathcal{A}_{f_1}, \cdots, \mathcal{A}_{f_k}\right). \end{aligned} \tag{1}$$

We use the fast fourier transform(FFT) for calculation of amplitude values $\mathcal{A}$ of the time series data. Specifically, $\mathcal{A}$ is the calculated amplitude of each frequency, and the i-th value $\mathcal{A}_i$ represents the intensity of the i-th frequency periodic basis function. In order to reduce meaningless high-frequency noise and atypical period factors in the sequence, and to strengthen the important period model, we select the top-k salient amplitudes with $\mathcal{A}$ to find the significant sequence frequencies $\{f_1, \cdots, f_k\}$, where $k = \lfloor c \times lo\ g\ L \rfloor$, c is the hyper-parameter. Based on the selected sequence frequencies, we can obtain top-k significant period lengths $\{l_1,\ \cdots, l_k\}$. Furthermore, the degree of period significance $\{\hat{S}(l_1), \cdots, \hat{S}(l_k)\}$ under the corresponding period length can be calculated by $Softmax$ with their corresponding amplitudes. Then the module performs two tasks. One is to pass these period-based dependencies information to each submodule of the encoder-decoder architecture. The other task is to summarize and embed each significant period into the input of the model as the initial encoding by Multi-Layers Periodic Stacked Embedding (MPS-Embedding).

$$MPS - Embeddin\, g(\mathcal{X}) = Conv1d\left(\mathcal{X} + \textstyle\sum_{i=1}^{k} \hat{S}(l_i) Linea\, r(\mathrm{Roll}(\mathcal{X}, l_i))\right) \tag{2}$$

$Conv1d$ and $Linear$ stand for convolution and fully-connected network architecture of the embedded stage standard, respectively. $Roll$ represents the delay l_i operation over $\mathcal{X}$, in which elements removed from the first position will be re-introduced into the last position. The $Roll$ operation provides a reconstructed sequence with a specified period length l_i, where the first l_i time points of this reconstructed time series are the last l_i time points of the original sequence. The reconstructed sequence expresses the repeatable relationship of the sequence every period length l_i apart through the mismatch operation as a way of verifying the reliability and confidence of this periodical pattern. And, we use linear projection to progressively approximate the potential role of the period pattern in the prediction and limit its influence by its significance $\hat{S}(l_i)$. By stacking independently modeled multi-layers period patterns, we encode the period-dependent information into the initial input of the model, so that the data periodic regularity can be fully considered and completely disassembled throughout the model.

2.3 Multi-Head Foresight Attention Mechanism

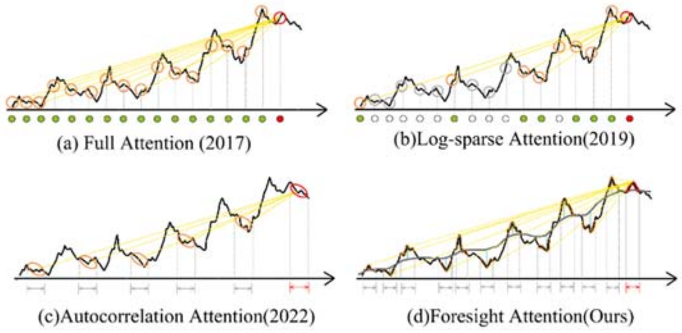


Fig. 4. Comparison of different attention mechanisms.

As shown in Fig.4, different from traditional self-attention family and autocorrelation mechanism, we propose the Foresight Attention mechanism. This mechanism adopts period information convergence and selects certain important sub-sequences for the interaction of stable trends and seasonal fluctuations to expand the information utilization. We then describe in detail how Foresight Attention performs period-dependent information aggregation and importance filtering. While inheriting the basic multi-head structure of transformer that respectively linearly project input $\boldsymbol{X}$ into $\boldsymbol{h}$ distinct query matrices $\boldsymbol{Q_i} = \boldsymbol{XW_i^Q}$, key matrices $\boldsymbol{K_i} = \boldsymbol{XW_i^K}$, and value matrices $\boldsymbol{V_i} = \boldsymbol{XW_i^V}$, with $\boldsymbol{i = 1, \cdots, h}$. After these linear projections with learnable parameters, the canonical self-attention is defined based on tuple inputs and performs the scaled dot-product. However, in the forecast task requirements, the output that needs to be modeled is a subsequent paragraph of the current information rather than itself. Therefore, the Query of foresight contributes more to their predictive effectiveness if some segments with respect to the aggregation of Query in the future period are highly relevant to the Key's information in the past period. Inspired by this intuition, the Foresight Attention mechanism carries out a time-serial-oriented design on the classical Query, Key, and Value.

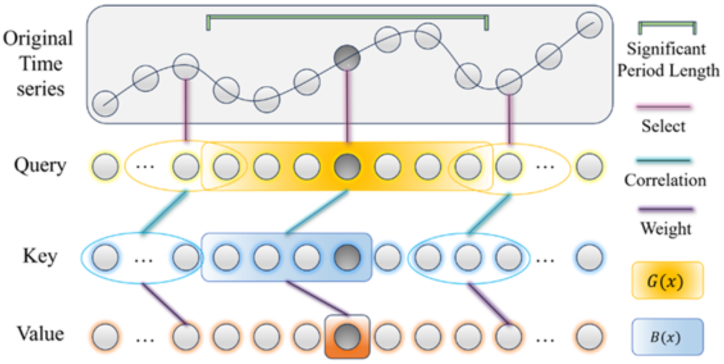


Fig. 5. Visualisation of the core concepts of the Foresight Attention mechanism..

As shown in Fig.5, the Query and Key are no longer single sampled points on the time series. Instead, the Query is designed as a local sub-sequence aggregation with foresight that knows the inside information of the future clearly. Meanwhile, the Key is regarded as a summary of past experience that converges the past local known content in the temporal sequence without the message of the not occurred events. The Value keeps precise and specific time points to ensure that the model prediction granularity and accuracy are not affected by the minor loss of information or ambiguous data during the aggregation of adjacent messages. The Foresight Attention mechanism we designed computes a sequence of vector outputs, as shown in the following:

$$\text{ForesightAttention}\ (\boldsymbol{Q}_i, \boldsymbol{K}_i, \boldsymbol{V}_i) = Softma\,x\left(\frac{\tilde{\boldsymbol{q}}_i \boldsymbol{k}_i^{\top}}{\sqrt{d_v}}\right)\boldsymbol{V}_i \tag{3}$$

Similar but different from the scaled dot-product attention. The $\tilde{\boldsymbol{q}}$ represents the $\boldsymbol{Q}_i$ that has been processed by $\boldsymbol{Select}$ which to obtain the local sub-sequence convergence and then to screen by importance sparse filtering. Meanwhile, $\boldsymbol{k}_i$ represents the empirical summary of past locally known information that has been processed by $\mathcal{B}$. Technically, where $\mathcal{G}$ denotes the filter of the discretized sliding window convolution, the size of the receptive scope is chosen as the most significant period length l, and using a one-dimensional weighted average convolution kernel with a Gaussian-like distribution to collect past experience and future development. Meanwhile, to find the difference in sequence trend around the current point in time, the filter $\mathcal{B}$ is oriented to collect context about past experiences on the sequence rather than the local fullness of information convergence generated by $\mathcal{G}$, thus $\mathcal{B}$ changing the sliding window size by halving the center of the convolution kernel. In general, through the differential of the sense of convolutional filter to achieve the similarity sub-sequence calculation and convergence, then capture the intrinsic dependence and development trend of the time series. Among them, the period length $\boldsymbol{l}$ and the corresponding significance degree $\hat{S}(l)$ are obtained by frequency domain analysis of the Multi-Layers Periodic Stacked Embedding module. The sparse screening $\boldsymbol{Select}$ mechanism is responsible for finding salient discrepancy points in the difference before and after the Gaussian filter $\mathcal{G}$ processing, which are generally manifested in turnaround or fluctuant points on the temporal sequence and sharp-shaped points that zigzag and meander on the graph. We believe that the attention mechanism should rightly pay more attention to and put more effort into such sequence changes and time fragments to grasp the deep reason for the sequence characteristics of the undulation. Data points that are sparsely filtered out are replaced with a mean value to satisfy the matrix calculation, which is similar to the strategy adopted by Informer.

2.4 Multi-Scale Polishing Sequence Decomposition

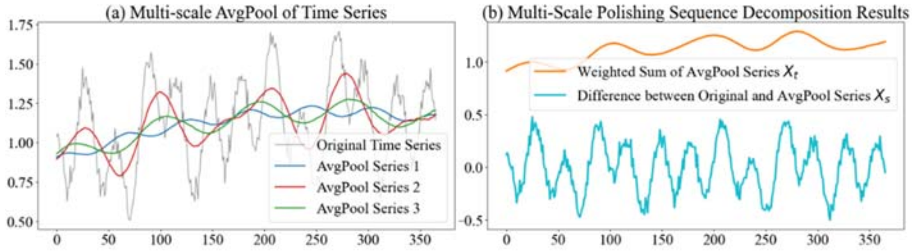


Fig. 6. (a) Multi-Scale Polishing Sequence Decomposition computes multiple moving averages of the original sequence based on multiple significant period lengths. (b) The period significance degree is then used as a weight in the summation of multiple AvgPool Series, to obtain the trend term.

Instead of using a sequence smoothing operation in the pre-processing stage as in the traditional approach, we place the smoothing block as an internal sub-module in the model. By running alternately with other sub-modules, the model obtained the inherent ability to decompose complex time patterns step by step. In addition, we improve the manually specified single-scale and ordinary multi-scale strategies. We use cyclicality to guide the multi-scale progressive smooth decomposition. Compared with the sliding window size specified by the hyper-parameter, our method can achieve sequence adaptation by relying on reliable period-dependent calculation. The multi-scale design allows each period length to be considered, and the corresponding significance ensures the rationality of the decomposition degree. In general, the Multi-Scale Polishing Sequence Decomposition block can strip the seasonal fluctuation from the long-term stable tendency of the series intermediate hidden variable, and deliver the regular and unlearned content to the subsequent module for further analysis. It enables the model to progressively capture the overall profile of the time series and grasp the direction of the trend. As shown in Fig.6, the $\text{Multi-Scale Polishing Sequence Decomposition}(\mathcal{X})$ process is:

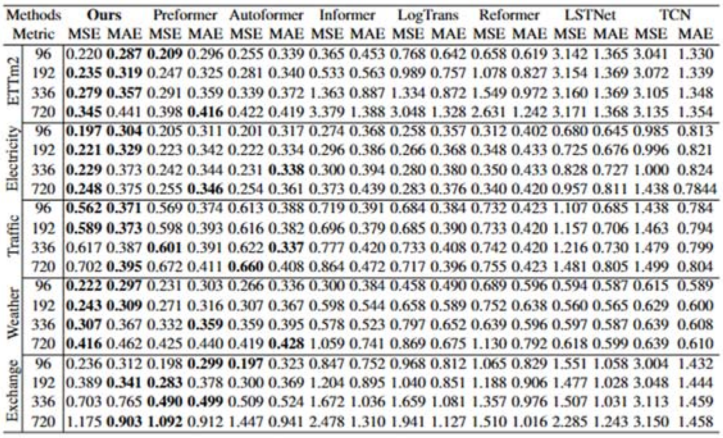
$$\begin{gathered}\mathcal{X}_t = \sum_{i=1}^{k} AvgPoo\,l(\text{ Padding }(\mathcal{X}), l_i)\hat{\mathcal{S}}(l_i) \\ \mathcal{X}_s = \mathcal{X} - \mathcal{X}_l\end{gathered} \tag{3}$$

Where $AvgPool$ represents the average pooling operation, where the second parameter points to the sliding window size. Then l_i and $\hat{\mathcal{S}}(l_i)$ are the information of the periodicity pattern that is the period length and significance degree. The single average pooling operation will bring about a fixed decomposition mode, while the multi-scale design makes decomposition more reliable and reasonable. The smooth kernel size specified by the period length has a natural advantage in the decomposition of trend and seasonal terms, and the stable fluctuations of the seasonal terms can be perfectly captured by the same length of the period. The corresponding degree of significance guarantees the range of sequence decomposition of multiple periodic scales.

3 Experiments

3.1 Comparison with state-of-the-art methods

Table 1. Multivariate long sequence time-series forecasting results on five datasets. A lower MSE or MAE indicates a better prediction, and the best results are highlighted in bold.



Methods		Ours		Preformer		Autoformer		Informer		LogTrans		Reformer		LSTNet		TCN	
Metric		MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE
ETTm2	96	0.220	**0.287**	**0.209**	0.296	0.255	0.339	0.365	0.453	0.768	0.642	0.658	0.619	3.142	1.365	3.041	1.330
	192	**0.235**	**0.319**	0.247	0.325	0.281	0.340	0.533	0.563	0.989	0.757	1.078	0.827	3.154	1.369	3.072	1.339
	336	**0.279**	**0.357**	0.291	0.359	0.339	0.372	1.363	0.887	1.334	0.872	1.549	0.972	3.160	1.369	3.105	1.348
	720	**0.345**	0.441	0.398	**0.416**	0.422	0.419	3.379	1.388	3.048	1.328	2.631	1.242	3.171	1.368	3.135	1.354
Electricity	96	**0.197**	**0.304**	0.205	0.311	0.201	0.317	0.274	0.368	0.258	0.357	0.312	0.402	0.680	0.645	0.985	0.813
	192	**0.221**	**0.329**	0.223	0.342	0.222	0.334	0.296	0.386	0.266	0.368	0.348	0.433	0.725	0.676	0.996	0.821
	336	**0.229**	0.373	0.242	0.344	0.231	**0.338**	0.300	0.394	0.280	0.380	0.350	0.433	0.828	0.727	1.000	0.824
	720	**0.248**	0.375	0.255	**0.346**	0.254	0.361	0.373	0.439	0.283	0.376	0.340	0.420	0.957	0.811	1.438	0.7844
Traffic	96	**0.562**	**0.371**	0.569	0.374	0.613	0.388	0.719	0.391	0.684	0.384	0.732	0.423	1.107	0.685	1.438	0.784
	192	**0.589**	**0.373**	0.598	0.393	0.616	0.382	0.696	0.379	0.685	0.390	0.733	0.420	1.157	0.706	1.463	0.794
	336	0.617	0.387	**0.601**	0.391	0.622	**0.337**	0.777	0.420	0.733	0.408	0.742	0.420	1.216	0.730	1.479	0.799
	720	0.702	**0.395**	0.672	0.411	**0.660**	0.408	0.864	0.472	0.717	0.396	0.755	0.423	1.481	0.805	1.499	0.804
Weather	96	**0.222**	**0.297**	0.231	0.303	0.266	0.336	0.300	0.384	0.458	0.490	0.689	0.596	0.594	0.587	0.615	0.589
	192	**0.243**	**0.309**	0.271	0.316	0.307	0.367	0.598	0.544	0.658	0.589	0.752	0.638	0.560	0.565	0.629	0.600
	336	**0.307**	0.367	0.332	**0.359**	0.359	0.395	0.578	0.523	0.797	0.652	0.639	0.596	0.597	0.587	0.639	0.608
	720	**0.416**	0.462	0.425	0.440	0.419	**0.428**	1.059	0.741	0.869	0.675	1.130	0.792	0.618	0.599	0.639	0.610
Exchange	96	0.236	0.312	0.198	**0.299**	**0.197**	0.323	0.847	0.752	0.968	0.812	1.065	0.829	1.551	1.058	3.004	1.432
	192	0.389	**0.341**	**0.283**	0.378	0.300	0.369	1.204	0.895	1.040	0.851	1.188	0.906	1.477	1.028	3.048	1.444
	336	0.703	0.765	**0.490**	**0.499**	0.509	0.524	1.672	1.036	1.659	1.081	1.357	0.976	1.507	1.031	3.113	1.459
	720	1.175	**0.903**	**1.092**	0.912	1.447	0.941	2.478	1.310	1.941	1.127	1.510	1.016	2.285	1.243	3.150	1.458

The proposed model demonstrates satisfactory performance in the majority of cases and prediction length settings for multivariate long-term series forecasting, as indicated in Table.1. Our model exhibits significantly superior results compared to the classical baseline, showcasing an average improvement of 69.6\% (from 1.34 to 0.407) over RNN-based LSTNet and 76.2\% (from 1.712 to 0.407) over CNN-based TCN models. In comparison to the recently popular and advanced sparse attention-based transformer family, including Informer, LogTrans, and Reformer, our method achieves an average mean squared error (MSE) reduction of 49.1\% (from 0.801 to 0.407). Particularly noteworthy is the performance of our method on the ETTm2 dataset. When compared to the previous Autoformer model, which performed well in terms of period utilization, our method exhibits a relative MSE reduction of 13.7\% (from 0.255 to 0.220) under the predict-96 setting, 16.3\% (from 0.281 to 0.235) under the predict-192 setting, 17.6\% (from 0.339 to 0.289) under the predict-336 setting, and 22.2\% (from 0.422 to 0.350) under the predict-720 setting. Furthermore, we observe a steady increase in the performance of prediction errors in our method as the prediction length O grows. This finding indicates that our model maintains better long-term robustness and showcases its competitiveness in terms of long-term time-series forecasting. It exhibits a win-or-loss scenario compared to the Preformer model only when applied to the Exchange dataset. This phenomenon can be attributed to the anisotropic nature of specific fluctuations observed in the Exchange dataset, which lack evident periodic characteristics. Collectively, these results highlight the ability of our method to effectively address multivariate time-series forecasting tasks in real-world applications, such as weather early warning systems and long-term energy consumption planning.

3.2 Model Analysis

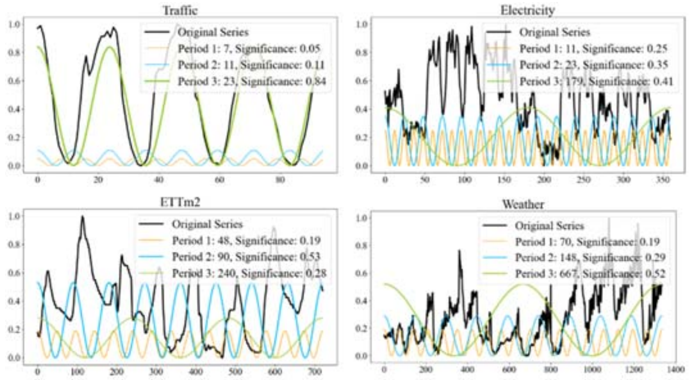


Fig. 7. Visualize our proposed multi-scale periodicity metric on real-world datasets.

The multi-scale period idea of the present model is demonstrated by visualization on the dataset used for the large-scale experiments, as shown in Fig.7. In all four datasets with different input lengths, our method demonstrates excellent multiscale period analysis, and the three scales of period stacking can almost reconstruct the original sequence. The periodic dependence capture used here relies heavily on Equation.1, which calculates the k significant periods of each original time series. This includes the property of period length and significance degree. The period length is mostly employed in the model to specify a reasonable size of the receptive domain, while the significance degree is mostly applied to score the content at the corresponding scale. The multi-scale period provides an inherent perspective for analyzing the time series inputs, provides a reasonable point of view for capturing the periodic dependence, and also allows the model to restore and reconstruct the sequence periodicity to the maximum extent possible.

3.3 Visualization

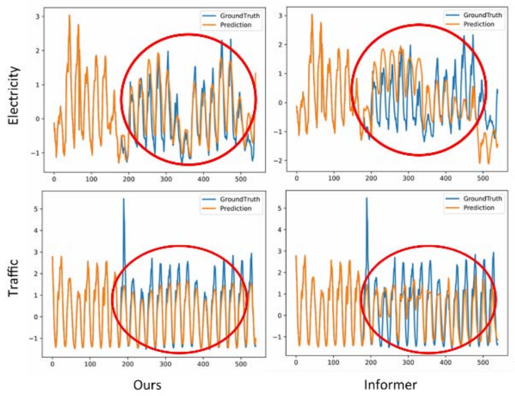


Fig. 8. Visualization of predictive effects on Electricity and Traffic datasets. Compare our approach with the Informer model.

The visualization of predicted effects on the two data sets, as depicted in Fig.8, serves as a faithful representation of our design philosophy and substantiates the validity of our model. Our model adeptly captures and reproduces the periodicity inherent in the time series data. Furthermore, it accurately preserves the range and duration of each cycle, even in the presence of trend shifts and continuous fluctuations caused by the superimposition of multiple cycles. In contrast, the Informer model, lacking the incorporation of cycle dependence in its design, exhibits imperfections in terms of capturing periodicity. However, our Foresight Attention mechanism effectively predicts the peaks and troughs of time series transitions and variations, demonstrating remarkable periodicity. Moreover, our model closely aligns with the trend of the ground truth without experiencing premature or delayed predictions. It faithfully adheres to the direction of the trend and avoids unnecessary fluctuations, crucial for ensuring realistic forecasting outcomes. Conversely, the Informer model's sparse probability screening, while achieving low complexity, results in the loss of significant sequence transition information. This limitation may lead to inadequate learning of seasonality fluctuations and abrupt changes in forecast trends. In summary, the comparison between our model and the Informer model highlights the superiority of our approach in accurately capturing and reproducing the periodicity of time series data. Our Foresight Attention mechanism plays a pivotal role in predicting transitions and variations with remarkable periodicity while faithfully following the trend direction. The limitations of the Informer model, including the loss of sequence transition information and potential fluctuations in seasonality and forecast trends, underscore the advantages of our design.

4 Conclusions

This paper proposes a long-term series prediction model based on multi-scale period-dependent modeling which fully making uses of the periodic characteristics of time series. We construct three modules that use period information in different stages and these three modules constitute a continuous period relationship interaction. It breaks the bottleneck of period information utilization and improves forecasting performance by emphasizing the discovery of periodic dependencies and the aggregation of fluctuation sub-sequences information. The proposed method achieves $O(LlogL)$ of time complexity, and extensive experiments on real-world datasets demonstrate that it achieves the better prediction results than existing comparable Transformer-base methods, especially on datasets with significant periodicity. In the future, we will continue to explore other possible ways of capturing dependencies, and to study the role and extend the applicability of periodicity in the field of time series forecasting.

Acknowledgments. This work is supported by the Natural Science Foundation of Chongqing (Grant Nos. CSTB2023NSCQ-LZX0160, CSTB2023NSCQ-LZX0012, CSTB2022NSCQ-LZX0040), the Open Project of State Key Laboratory of Intelligent Vehicle Safety Technology (Grant No. IVSTSKL-202302).

References

1. Nasiri, H., Ebadzadeh, M.M.: Multi-step-ahead stock price prediction using recurrent fuzzy neural network and variational mode decomposition. Applied Soft Computing 148, 110867 (2023)
2. Du, B., Huang, S., Guo, J., Tang, H., Wang, L., Zhou, S.: Interval forecasting for urban water demand using pso optimized kde distribution and lstm neural networks. Applied Soft-Computing 122, 108875 (2022)
3. Dikshit, A., Pradhan, B., Santosh, M.: Artificial neural networks in drought prediction in the 21st century–a scientometric analysis. Applied Soft Computing 114, 108080 (2022)
4. Papadimitriou, S., Yu, P.: Optimal multi-scale patterns in time series streams pp. 647–658 (2006)
5. Lim, B., Zohren, S.: Time-series forecasting with deep learning: a survey. Philosophical-Transactions of the Royal Society A 379(2194), 20200209 (2021)
6. Maddix, D.C., Wang, Y., Smola, A.: Deep factors with gaussian processes for forecasting. arXiv preprint arXiv:1812.00098 (2018)
7. Graves, A., Graves, A.: Long short-term memory. Supervised sequence labelling with recurrent neural networks pp. 37–45 (2012)
8. Bai, S., Kolter, J.Z., Koltun, V.: An empirical evaluation of generic convolutional and recurrent networks for sequence modeling. arXiv preprint arXiv:1803.01271 (2018)
9. Oreshkin, B.N., Carpov, D., Chapados, N., Bengio, Y.: N-beats: Neural basis expansion analysis for interpretable time series forecasting. arXiv preprint arXiv:1905.10437 (2019)
10. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł.Polosukhin, I.: Attention is all you need. Advances in neural information processing systems 30 (2017)
11. Kitaev, N., Kaiser, Ł., Levskaya, A.: Reformer: The efficient transformer. arXiv preprint arXiv:2001.04451 (2020)
12. Li, S., Jin, X., Xuan, Y., Zhou, X., Chen, W., Wang, Y.X., Yan, X.: Enhancing the locality and breaking the memory bottleneck of transformer on time series forecasting. Advances in neural information processing systems 32 (2019)
13. Zhou, H., Zhang, S., Peng, J., Zhang, S., Li, J., Xiong, H., Zhang, W.: Informer: Beyond efficient transformer for long sequence time-series forecasting 35(12), 11106–11115 (2021)
14. Zhou, T., Ma, Z., Wen, Q., Wang, X., Sun, L., Jin, R.: Fedformer: Frequency enhanced decomposed transformer for long-term series forecasting pp. 27268–27286 (2022)
15. Wu, H., Xu, J., Wang, J., Long, M.: Autoformer: Decomposition transformers with auto-correlation for long-term series forecasting. Advances in Neural Information Processing Systems 34, 22419–22430 (2021)
16. Cleveland, R.B., Cleveland, W.S., McRae, J.E., Terpenning, I.: Stl: A seasonal-trend decomposition. J. Off. Stat 6(1), 3–73 (1990)
17. Wang, S., Li, B.Z., Khabsa, M., Fang, H., Ma, H.: Linformer: Self-attention with linear complexity. arXiv preprint arXiv:2006.04768 (2020)
18. Wu H, Hu T, Liu Y, et al. Timesnet: Temporal 2d-variation modeling for general time series analysis[C]//The eleventh international conference on learning representations. 2022.
19. Chen P, Zhang Y, Cheng Y, et al. Pathformer: Multi-scale transformers with Adaptive Pathways for Time Series Forecasting[J]. arXiv preprint arXiv:2402.05956, 2024.
20. Torres, J.F., Hadjout, D., Sebaa, A., Mart´ınez- ´Alvarez, F., Troncoso, A.: Deep learning for time series forecasting: a survey. Big Data 9(1), 3–21 (2021)