MCformer: Multivariate Time Series Forecasting with Mixed-Channels Transformer

Wenyong Han, Tao Zhu *Member, IEEE*, Liming Chen *Senior Member, IEEE*
Huansheng Ning *Senior Member, IEEE*, Yang Luo *Member, IEEE*, Yaping Wan *Member, IEEE*

***Abstract*—The massive generation of time-series data by large-scale Internet of Things (IoT) devices necessitates the exploration of more effective models for multivariate time-series forecasting. In previous models, there was a predominant use of the Channel Dependence (CD) strategy (where each channel represents a univariate sequence). Current state-of-the-art (SOTA) models primarily rely on the Channel Independence (CI) strategy. The CI strategy treats all channels as a single channel, expanding the dataset to improve generalization performance and avoiding inter-channel correlation that disrupts long-term features. However, the CI strategy faces the challenge of inter-channel correlation forgetting. To address this issue, we propose an innovative Mixed Channels strategy, combining the data expansion advantages of the CI strategy with the ability to counteract inter-channel correlation forgetting. Based on this strategy, we introduce MCformer, a multivariate time-series forecasting model with mixed channel features. The model blends a specific number of channels, leveraging an attention mechanism to effectively capture inter-channel correlation information when modeling long-term features. Experimental results demonstrate that the Mixed Channels strategy outperforms pure CI strategy in multivariate time-series forecasting tasks.**

***Index Terms*—Multivariate time series, time series forecasting, Long time series, self-attention**

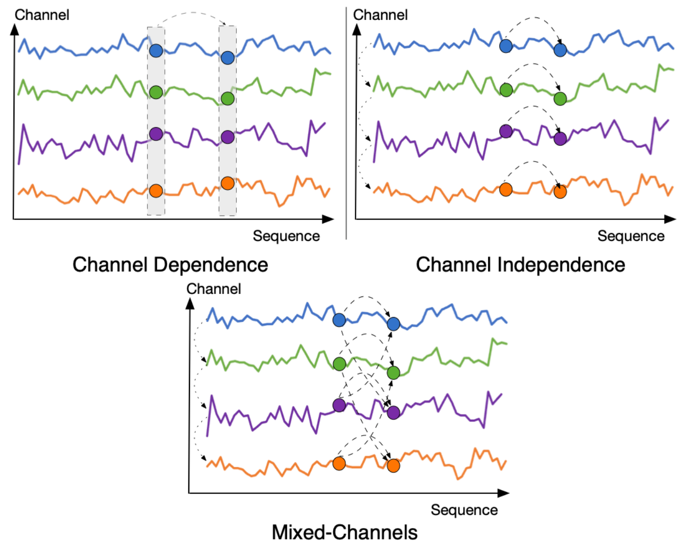


Fig. 1. The difference between the CI, CD and Mixed Channels strategies.

I. Introduction

WITH the widespread application of Internet of Things (IoT) devices in fields such as meteorology [1]–[3], traffic [4]–[7], and electricity [8], the increasing number of devices has resulted in the generation of a significant amount of time-series data. These data can be utilized for decision-making [9], resource allocation [10], and forecasting future trends [11], [12], thereby enhancing the efficiency and reliability of IoT systems. Time-series forecasting tasks arising from IoT devices aim to forecast future states based on historical data. Given that IoT data typically exhibits characteristics such as non-linearity, rapid sampling, and multi-channel aspects, this task poses certain challenges.

Due to the typically longer sampling intervals and numerous sampling channels in time-series data generated by IoT devices, when dealing with multivariate time series from such IoT devices, it is necessary to consider both long sequence modeling and the complex interrelationships between multiple channels. Given the outstanding capability of Transformers in modeling long sequences demonstrated in the field of natural language processing, this ability is also crucial in time-series forecasting tasks, leading to the emergence of models like Log-Trans [13], Informer [14], Reformer [15], Autoformer [16], FEDformer [17], ScaleFormer [18], Pyraformer [19], FPP-former [20], and others. These models have made significant progress in the realm of long time-series modeling.In recent years, some research has shifted focus to the challenges of multivariate time series, exemplified by models such as Cross-former [21] and SageFormer [22]. These models undertake learning across all channels, with a specific focus on capturing dependencies between multiple variables. All these models can be considered as Channel Dependence (CD) Strategy models (A univariate sequence is treated as a channel),This approach takes multivariate data as a whole input and allows the model to learn the correlation between channels, as shown in the Fig. 1.

However, these CD strategy models also have drawbacks. DLinear [23] has surpassed existing models with a simple architecture. PatchTST [24] introduces a Channel Independent (CI) strategy model, further improving the state-of-the-art (SOTA). The CI strategy treats all channels as a single channel,

This work was supported in part by the National Natural Science Foundation of China(62006110), the Natural Science Foundation of Hunan Province (2021JJ30574). (Corresponding author: Tao Zhu.)

Wenyong Han, Yang Luo, Tao Zhu and Yaping Wan are with the School of Computer Science, University of South China, 421001 China (e-mail:wyhan@stu.usc.edu.com, luoyang@usc.edu.cn, tzhu@usc.edu.cn, ypwan@aliyun.com).

Liming Chen was with School of Computer Science and Technology, Dalian University of Technology, China. Huansheng Ning was Department of Computer & Communication Engineering, University of Science and Technology Beijing, 100083 China (email: l.chen@ulster.ac.uk, ninghuansheng@ustb.edu.cn).

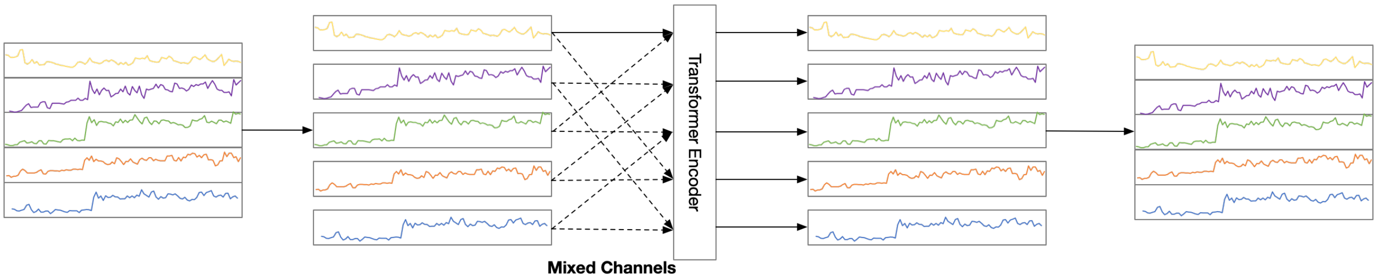


Fig. 2. Overview of the Mixed Channels method: Multivariate time series data is initially decomposed by channel, resulting in individual channel data. Subsequently, based on the channel interval size, data from different channels is mixed. The mixed data will share parameters in the Transformer Encoder.

thereby expanding the dataset and enhancing the model's generalization capability, as shown in the Fig. 1. The success of the CI strategy has drawn attention to the impact of both CD and CI strategies on models, as seen in models like PRReg [25], PETformer [26], CSformer [27], itransformer [28], and others. Subsequently, TiDE [29] (a CI model based on MLP) not only performs similarly to PatchTST but also excels in spatiotemporal efficiency. Research on PETformer found that channel independence is superior to channel dependence, possibly because multivariate features can interfere with the extraction of long sequence features. This result goes against intuition, as in deep learning, more information typically improves model generalization.

In summary, there are two main reasons why existing SOTA models are mostly based on the CI strategy: firstly, the CI strategy can expand the dataset to improve the generalization performance of the model, as seen in PatchTST; secondly, the CI strategy can avoid the destruction of long-term feature information by channel-wise correlation information, as demonstrated by PETformer. However, the CI strategy also has drawbacks, as it tends to overlook inter-channel feature information. In cases with a large number of channels, there may be an issue of inter-channel correlation forgetting, akin to the forgetting of long-term information in RNNs [30]. In this context, we propose a Mixed Channels strategy. This strategy retains the advantages of the CI strategy in expanding the dataset while effectively avoiding the disruption of long-term feature information by channels. It also addresses the issue of inter-channel correlation forgetting. Fig. 1 illustrates the differences between the Mixed Channels strategy and CI and CD. Based on the aforementioned strategy, we propose a multi-channel time series forecasting model with mixed channel features. Fig. 2 provides an overview of the model. Specifically, our model first expands the data using the CI strategy, then mixes a specific number of channels, and allows the attention mechanism to effectively capture the correlation information between channels when modeling long-term feature information. Finally, the encoder result is unflattened to obtain the predicted values of all channels. The contributions of our proposed model can be summarized as follows:

- We propose a Mixed Channels strategy that seeks to minimize the drawbacks of channel features disrupting long-term information under the CD strategy, while retaining the advantages of expanding the dataset under the CI strategy. This enables the model to more effectively learn inter-channel dependency information.
- Based on the Mixed Channels strategy, we present a Multivariate time-series forecasting model with mixed channel features. By employing the Mixed-Channels Block, the model expands the dataset and integrates inter-channel dependency information through a blended approach.
- Furthermore, MCformer has been experimentally evaluated on five real-world Multivariate datasets, achieving outstanding results compared to the current SOTA. We conduct two ablation experiments, investigating the impact of the Mixed-channels approach on datasets with different channel quantities and the influence of varying channel mixing quantities on model performance. This further substantiates the effectiveness of our approach. Additionally, we explore the rich feature relationships among multiple channels. Using correlation analysis, we present changing curves in the inter-channel correlations across multiple datasets, illustrating the dynamic evolution of feature relationships among channels over time.

II. Related Work

Long Time Series Forecasting. In the past, long time series forecasting has been a focal point in the field of time series analysis. The approaches to address this problem can be broadly categorized into several models: statistical methods, MLP-based methods, CNN-based methods, RNN-based methods, and Attention-based methods. Statistical models like ARIMA [31] assume time series stationarity, but many real-world time series are non-stationary. RNN-based models (such as LSTNet [32] and DeepAR [33]) excel in capturing sequence features, but their reliance on hidden states for feature propagation makes them notably deficient in modeling long sequences. In recent years, CNN-based models (like MICN [34]and TimesNet [35]) have emerged, leveraging the outstanding performance of CNNs in the image domain to model multivariate and long sequence features. With the advent of Transformers, which have demonstrated excellent performance in natural language processing, they have gained attention in the time series domain as well. Models like Informer [14], Autoformer [16], and FEDformer [17] have achieved SOTA by building upon and improving the Transformer. Informer effectively extracts critical information by employing an improved ProbSparse self-attention mechanism.

Autoformer captures both local and global features of time series by leveraging the decomposition and autocorrelation concepts from traditional time series analysis. FEDformer, on the other hand, transforms attention from the time domain to the frequency domain, reducing complexity through Fourier transformation. MLP-based models, gaining more attention in long sequence forecasting after the publication of Dlinear [23], have spurred questioning of the modeling capability of Transformers in time series. Subsequently, PatchTST [24], utilizing the native ViT [36] with a single-channel strategy, surpassed Dlinear once again.

Multivariate Time series Forecasting. In the realm of multivariate time series forecasting, the challenge of long time series forecasting has persistently garnered attention. However, in recent years, especially with the powerful complex feature extraction capabilities demonstrated by Transformers across various modalities of data, more research in multivariate time series forecasting has started focusing on a model's ability to model interactions among multiple variables. We briefly summarize some research on Multivariate feature modeling in previous models based on the CD strategy. For instance, models like TimesNet [35] and LSTnet [32] leverage CNNs to capture dependencies across dimensions. In Former-based models, such as Crossformer [21], the input multivariate time series is embedded into a 2D vector array to retain both temporal and dimensional information. A Two-Stage Attention (TSA) layer is then introduced to efficiently capture dependencies across time and dimensions.The Client [37] and iTransformer [28] models differ from traditional enhanced Transformer models in that they transform the function of Attention from learning long-term features to learning inter-variable features. They utilize linear modules to learn trend information and attention modules to capture cross-variable dependency relationships.SageFormer [22]is a Series-aware Graph-enhanced Transformer model designed to effectively capture and model dependencies between series using a graph structure. Although PatchTST [24] employs a CI strategy for training, it still learns cross-variable features through model parameterization. The recent TiDE [29](MLP-Based Model) follows a similar strategy. Due to the impact of PatchTST, analyses in PRReg [25] and PETformer [26] have explored the reasons for the lower effectiveness of CD strategies compared to CI strategies, proposing methods to enhance the effectiveness of CD strategies.

III. METHOD

A. Problem Definition

In the task of multivariate time series forecasting, historical observations are represented as $X = \{\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_t\} \in \mathbb{R}^{t\times M}$, where t is the time step and M is the number of variables. The observations at each time step t are represented by a M-dimensional vector: $\mathbf{x}_t = [x_t^1, x_t^2, \dots, x_t^M]^\top$. Our objective is to predict the multivariate observations at future time steps based on past observations. The problem can be formalized as follows: given the observed sequence $\{\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_t\}$ up to time step t, predict the multivariate observation sequence $\{\mathbf{x}_{t+1}, ..., \mathbf{x}_{t+h}\}$ for time steps $t+1$ to $t+h$, where h is the number of time steps to be predicted. We incorporate a Mixed-Channels Block into the vanilla Transformer Encoder to expand the dataset and blend inter-channel dependency information. The architecture of our MCformer model is shown in Fig. 3.

B. Reversible instance Normalization

The introduction of Reversible Instance Normalization (RevIN) [38] aims to address the issue of non-uniform temporal distribution between training and testing data, commonly referred to as distribution shift. Before the Mixed Channels module, we apply instance normalization to normalize each channel's data. A single channel is represented as $\mathbf{x}^i = [x_1^i, x_2^i, \dots, x_t^i]$, where for each instance x_t^i, we calculate the mean and standard deviation. After obtaining the forecasting results, these non-stationary information components are added back to the predicted values.

$$RevIN(\mathbf{x}^i) = \left\{ \gamma_i \frac{\mathbf{x}^i - Mean(\mathbf{x}^i)}{\sqrt{Var(\mathbf{x}^i) + \varepsilon}} \right\}, i = 1, 2, \cdots, M \quad (1)$$

C. Mixed-Channels Block

We introduce a method called the Mixed Channels Module to enhance the representation of multivariate time series datasets.

Flatten We employ a Channel Independent (CI) strategy to flatten the data from M channels. For a given sample X, after flattening, we obtain $X_F = \text{Flatten}(X) \in \mathbb{R}^{tM\times 1}$. The flattened X_F is then treated as if it were M individual samples.

Mixed Channels Mixed Channels involves combining data from different channels after Flatten. We perform the mixed channels operation through the following steps:

1) Compute Interval Size: We calculate the interval size $\left\lfloor \frac{M}{m} \right\rfloor$, where m is the number of channels to be mixed.
2) Mixed Channels Operation: For a given time step t, starting from the target channel, we stack every other channel at an interval stride to form $U^i \in \mathbb{R}^{t\times m}$. Specifically, the output of the Mixed Channels Module is defined as:

$$\begin{aligned} U^i &= MixedChannels(\mathbf{x}^i, m) \\ &= [stack(\mathbf{x}^i, C^1, C^2, \dots, C^m)] \end{aligned} \quad (2)$$

where C^i represents the i-th channel taken at the i-th interval, and $1 \leq i \leq m$. By introducing the Mixed Channels Module, our aim is to enhance the expressive power of the input data, introducing multi-channel information to better capture the features of the time series data.

Patch and Projection In the current research [24], [26], [29], it has been observed that, compared to using time-point data as input, employing patches can better capture local information and also encompass richer dependencies between variables. Therefore, we utilize Patch to aggregate the sequence after mixing channels, and employ a single-layer MLP to project channel dependencies as well as adjacent temporal dependencies. This is expressed as:

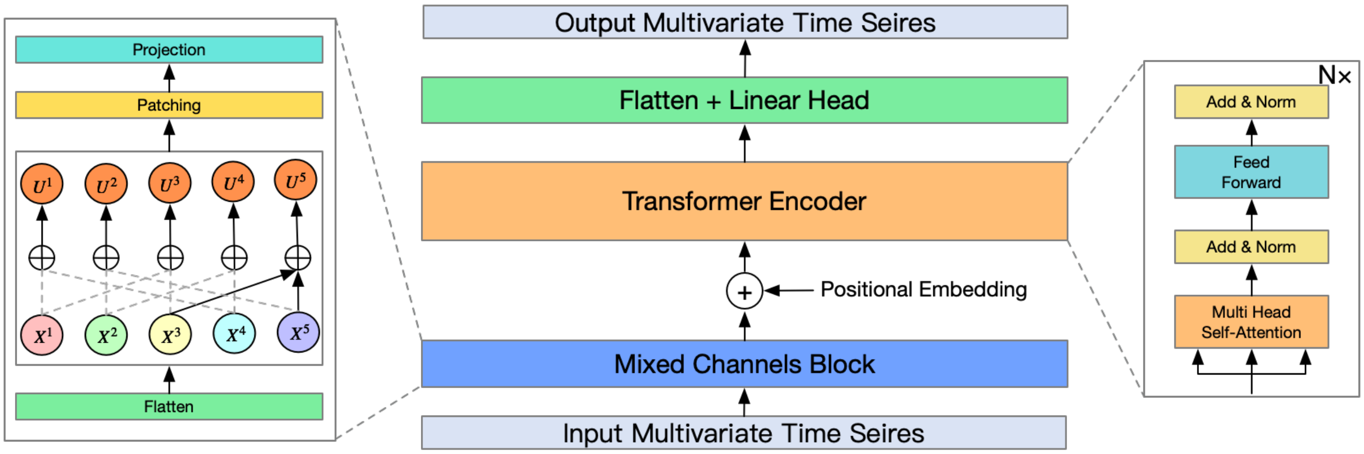


Fig. 3. Mixed Channels architecture: In the Mixed-Channels Block, we decompose multivariate time series data into single channels and then blend the data from different channels. The blended data is then segmented into multiple patches, with each patch composed of adjacent samples. These patches are transformed into input tokens through a projection process.

$$\mathcal{P}^i = Projection(Patch(U^i)) \tag{3}$$

Here $\mathcal{P}^i \in \mathbb{R}^{P \times N}$, where P is length after Projection, N is the number of patches, $N = \left\lfloor \frac{(L-p)}{S} \right\rfloor + 2$, and p is the length of Patch and S is the stride length. This patch approach allows for the retention of both time dependencies and dependencies between multiple channels. It not only preserves the Transformer input's tokens but also further increases the size of the forecasting window while maintaining time dependencies. You can see in Fig. 4 how patches are utilized in time series tasks.

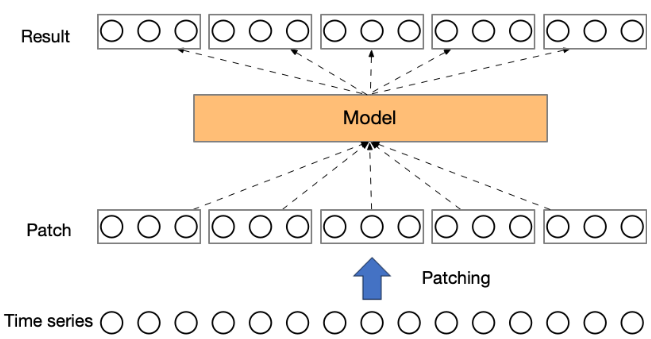


Fig. 4. Application of Patch to temporal models. Using patches significantly extends the historical time range of the input while maintaining the same token length.

Algorithm 1 Pseudocode of MCformer Training

Input Historical traffic observations X and future ground truth Y_h.
Output Predicted future observations $\hat{Y}_h$.
Shuffle data;
set $m \leftarrow stack\_len, p \leftarrow patch\_size$;
for each batch of in data **do**
  for $i = 1 \rightarrow M$ **do**
    The mean and variance is calculated by formula(1);
    $\mathbf{x}^i_{norm} \leftarrow RevIN(\mathbf{x}^i)$, normalize each instance;
  end
  $X_F \in \mathbb{R}^{tM \times 1} \leftarrow$ flattening all the $\mathbf{x}^i_{norm}$;
  $U \in \mathbb{R}^{tM \times m} \leftarrow$ calculate by formula (2);
  $\mathcal{P} \in \mathbb{R}^{M \times P \times N} \leftarrow$ the patches are divided and Projection by formula(3);
  $\hat{Y}_h \in \mathbb{R}^{hM \times 1} \leftarrow$ modeling with encoder;
  $\hat{Y}_h \in \mathbb{R}^{h \times M} \leftarrow$ Unflatten by forecasting window length h;
  for $t = 1 \rightarrow h$ **do**
    $\hat{Y}_{denorm} \leftarrow$denormalize $\hat{Y}_h$ each instance;
  end
  $loss \leftarrow$ calculate by formula(5);
  $Optimize(loss)$;
end

D. Encoder

We employ the native Transformer encoder [36], [39] to model the long-term and cross-variable features of the sequence, akin to the approach taken in PatchTST [24]. As the Transformer does not explicitly model the sequence's order, to provide positional information, we utilize learnable additive positional encoding $\mathcal{W}_{pos} \in \mathbb{R}^{P \times N}$. The positional encoding is added to the embedded representation of the input sequence $\mathcal{X}^i_{in} = \mathcal{P}^i + \mathcal{W}_{pos}$. This way, the model can differentiate elements at different positions.

The encoder consists of multiple layers with the same structure, and each layer comprises two sub-layers: the Multi-Head Self-Attention Layer and the Feedforward Neural Network Layer. The Multi-Head Self-Attention Layer is the first sub-layer of the encoder. In this layer, each element in the input sequence can interact with every other element, not just its neighboring elements. Three learnable matrices $\mathcal{W}^Q$, $\mathcal{W}^K$, and $\mathcal{W}^V$ are used to compute $Q^i = (\mathcal{X}^i_{in})^T\mathcal{W}^Q$,$K^i = (\mathcal{X}^i_{in})^T\mathcal{W}^K$, and $V^i = (\mathcal{X}^i_{in})^T\mathcal{W}^V$. This is achieved by calculating attention weights, where each element receives a set of weights indicating its importance to other elements.

With the use of a multi-head mechanism, the model can learn different aspects of attention, enabling it to better capture information within the sequence:

$$Attention(Q^i, K^i, V^i) = Softmax(\frac{Q^i(K^i)^T}{\sqrt{d_k}})V^i \quad (4)$$

The Feedforward Neural Network Layer comes after the Multi-Head Self-Attention Layer. The representation at each position is further processed through a fully connected feedforward neural network. This network typically consists of two fully connected layers, and its output is added to the input, creating a residual connection. This helps alleviate the gradient vanishing problem during training.

Residual connections and Layer Normalization are employed between the input and output of each sub-layer. This inclusion of residual connections makes it easier for the model to learn identity mappings, facilitating the training of deep networks. Additionally, the output of each sub-layer undergoes layer normalization to stabilize the training process.

E. Loss Function

We chose MSE (Mean Squared Error) and MAE (Mean Absolute Error) losses to evaluate the disparity between the model's forecastings and the actual values. MSE measures the model performance by calculating the average of the squared differences between the predicted and actual values:

$$\mathrm{MSE} = \frac{1}{h}\sum_{i=1}^{h}(y_i - \hat{y}_i)^2 \quad (5)$$

MAE assesses the model performance by computing the average of the absolute differences between the predicted and actual values:

$$\mathrm{MAE} = \frac{1}{h}\sum_{i=1}^{h}|y_i - \hat{y}_i| \quad (6)$$

We have introduced the main structure of MCformer. Algorithm 1 summarizes the training process of MCformer.

IV. EXPERIMENTS

A. Dateset

We used five datasets containing multivariate time series data to evaluate our model, including Electricity, Traffic, Weather [16], Solar-Energy [32], PEMS [40]. These datasets are widely used as benchmarks for multivariate time series forecasting, and details about the datasets can be found in Table I. It is worth noting that datasets like ETT (ETTh1, ETTh2, ETTm1, ETTm2) [14] and ILI have only 7 channels of data. Due to the fewer number of channels, they were not used for model evaluation.

Weather: The weather dataset was collected at approximately 1, 600 locations across the United States between 2010 and 2013, with a sampling frequency of one record every ten minutes. This dataset contains 21 channels.

Solar-Energy: The Solar-Energy dataset documents the solar power generation of a photovoltaic power station in Alabama in 2006, with readings captured every 10 minutes. Data from a total of 137 channels were collected.

TABLE I
THE DETAILED INFORMATION OF BENCHMARK DATASETS USED FOR TESTING

Dataset / Dataset Detail	Electricity	Traffic	Weather	Solar-Energy	PEMS
features	321	862	21	137	358
frequency	1h	1h	10m	10m	5m
length	26211	17451	52603	52179	21352

Electricity: The Electricity dataset captures the hourly electricity consumption (measured in kilowatt-hours) of 321 customers from 2012 to 2014.

Traffic: The Traffic dataset encompasses road occupancy data recorded by sensors on San Francisco Bay area freeways from 2015 to 2016. Readings are logged on an hourly basis, ranging from 0 to 1. A total of 862 sensor channels are included.

PEMS: The PEMS dataset is collected by the California Department of Transportation's Performance Measurement System (PeMS) in the California region. There are 358 channels of data recorded

B. Baseline

In the realm of time series forecasting, deep learning models have made remarkable strides, surpassing traditional approaches in a multitude of tasks. To assess the performance of our proposed methodology, we have meticulously selected a cohort of state-of-the-art (SOTA) multivariate time series forecasting models. Transformer-based models have exhibited exceptional performance in time series forecasting tasks. We have chosen the most representative models from this class, including InFormer [14], AutoFormer [16], FEDFormer [17], CrossFormer [21], and PatchTST [24]. Additionally, given the promising results recently achieved by MLP-based models, we have opted to include the most notable representatives, namely DLinear [23] and TiDE [29]. CNN-based models possess distinct advantages in multivariate feature extraction. Consequently, we have incorporated TimesNet [35] into our evaluation.

C. forecasting and baseline comparison

Setup We followed the experimental settings of TimesNet, where the look-back window length and forecasting window length for the Solar-Energy, Weather, Traffic, and Electricity datasets were set to 96. The forecasting window length was chosen as $h \in \{96, 192, 336, 720\}$. For the PEMS dataset, the look-back window length was 96, and the forecasting window length was $h \in \{12, 24, 48, 96\}$.

Results As shown in the TableII, our proposed MCFormer model achieved the best performance across all datasets. In all experimental comparisons, we secured 12 first places and 8 second places in MSE, and 15 first places and 5 second places in MAE. This indicates that our method outperforms all compared methods. Notably, compared to the single-channel

TABLE II

FULL RESULTS ON THE MULTIVARIATE FORECASTING TASK. WE USED A LOOK-BACK WINDOW OF LENGTH 96 FOR ALL DATASETS, AND FOR PEMS, WE USED FORECASTING WINDOWS $h \in \{12, 24, 48, 96\}$, WHILE FOR OTHERS $h \in \{96, 192, 336, 720\}$. THE BEST RESULTS ARE HIGHLIGHTED IN **BOLD**, AND THE SECOND-BEST RESULTS ARE UNDERLINED.

Models		MCformer (Ours)		TiDE [29] (2023)		PatchTST [24] (2023)		TimesNet [35] (2023)		CrossFormer [21] (2023)		Dlinear [23] (2023)		FEDFormer [17] (2022)		AutoFormer [16] (2021)		InFormer [14] (2021)	
Metric		MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE
Weather	96	0.167	**0.213**	0.202	0.261	0.177	0.218	0.172	0.220	**0.158**	0.230	0.196	0.255	0.217	0.296	0.266	0.336	0.300	0.384
	192	0.215	**0.257**	0.242	0.298	0.225	0.259	0.219	0.261	**0.206**	0.277	0.237	0.296	0.276	0.336	0.307	0.367	0.598	0.544
	336	**0.270**	**0.297**	0.287	0.335	0.278	0.297	0.280	0.306	0.272	0.335	0.283	0.335	0.339	0.380	0.359	0.395	0.578	0.523
	720	**0.348**	**0.348**	0.351	0.386	0.354	0.348	0.365	0.359	0.398	0.418	0.345	0.381	0.403	0.428	0.419	0.428	1.059	0.741
Traffic	96	**0.433**	**0.288**	0.805	0.493	0.544	0.359	0.593	0.321	0.522	0.290	0.650	0.396	0.587	0.366	0.613	0.388	0.719	0.391
	192	**0.440**	**0.286**	0.756	0.474	0.540	0.354	0.617	0.336	0.530	0.293	0.598	0.370	0.604	0.373	0.616	0.382	0.696	0.379
	336	**0.454**	**0.292**	0.762	0.477	0.551	0.358	0.629	0.336	0.558	0.305	0.605	0.373	0.621	0.383	0.622	0.337	0.777	0.420
	720	**0.489**	**0.312**	0.719	0.449	0.586	0.375	0.640	0.350	0.589	0.328	0.645	0.394	0.626	0.382	0.660	0.408	0.864	0.472
Electricity	96	**0.163**	**0.255**	0.237	0.329	0.195	0.285	0.168	0.272	0.219	0.314	0.197	0.282	0.193	0.308	0.201	0.317	0.274	0.368
	192	**0.172**	**0.262**	0.236	0.330	0.199	0.289	0.184	0.289	0.231	0.322	0.196	0.285	0.201	0.315	0.222	0.334	0.296	0.386
	336	**0.190**	**0.279**	0.249	0.344	0.215	0.305	0.198	0.300	0.246	0.337	0.209	0.301	0.214	0.329	0.231	0.338	0.300	0.394
	720	0.229	0.311	0.284	0.373	0.256	0.337	**0.220**	**0.320**	0.280	0.363	0.245	0.333	0.246	0.355	0.254	0.361	0.373	0.439
Solar-Energy	96	**0.212**	**0.256**	0.312	0.399	0.234	0.286	0.250	0.292	0.310	0.331	0.290	0.378	0.242	0.342	0.884	0.711	0.236	0.259
	192	0.241	0.275	0.339	0.416	0.267	0.310	0.296	0.318	0.734	0.725	0.320	0.398	0.285	0.380	0.834	0.692	**0.217**	**0.269**
	336	0.258	0.287	0.368	0.430	0.290	0.315	0.319	0.330	0.750	0.735	0.353	0.415	0.282	0.376	0.941	0.723	**0.249**	**0.283**
	720	0.264	**0.296**	0.370	0.425	0.289	0.317	0.338	0.337	0.769	0.765	0.356	0.413	0.357	0.427	0.882	0.717	**0.241**	0.317
PEMS	12	**0.072**	**0.180**	0.178	0.305	0.099	0.216	0.085	0.192	0.090	0.203	0.122	0.243	0.126	0.251	0.272	0.385	0.126	0.233
	24	**0.100**	**0.212**	0.257	0.371	0.142	0.259	0.118	0.223	0.121	0.240	0.201	0.317	0.149	0.275	0.334	0.440	0.139	0.250
	48	0.164	0.271	0.379	0.463	0.211	0.319	**0.155**	**0.260**	0.202	0.317	0.333	0.425	0.227	0.348	1.032	0.782	0.186	0.289
	96	0.240	0.337	0.490	0.539	0.269	0.370	**0.228**	**0.317**	0.262	0.367	0.457	0.515	0.348	0.434	1.031	0.796	0.233	0.323

strategies PatchTST and TiDE, our method shows significant improvements. This suggests that our approach effectively captures inter-channel dependencies, leading to additional performance gains. In comparison to the all-channel strategies TimesNet and Crossformer, our MCFormer effectively mitigates the disruption of multi-channel dependencies on long-term information.

D. Ablation study

In this section, we designed two ablation experiments to investigate the impact of different channel fusion quantities on the model's performance and the effectiveness of the channel fusion method on datasets with varying channel numbers.

Impact on Data with Different Channel Numbers We hypothesize that under a single-channel strategy, the model may experience the issue of forgetting inter-channel correlations when the number of channels is high. To validate this hypothesis, we designed experiments to test the predictive performance of the model under different channel numbers. We selected the Electricity, Traffic, Weather, Solar-Energy, and PEMS datasets for experimentation, with their characteristics detailed in Table I. We conducted experiments on different datasets with varying forecasting lengths and calculated their average performances. In order to compare the performance improvement of our model using a single-channel strategy, we contrasted TiDE and PatchTST. As depicted in Fig. 5, it is evident that MCformer exhibits a more significant enhancement in average MSE as the number of channels increases when compared to the single-channel strategy model. This indicates that our model is capable of effectively addressing the issue of inter-channel correlation forgetting and does not compromise model performance with an increase in channels.

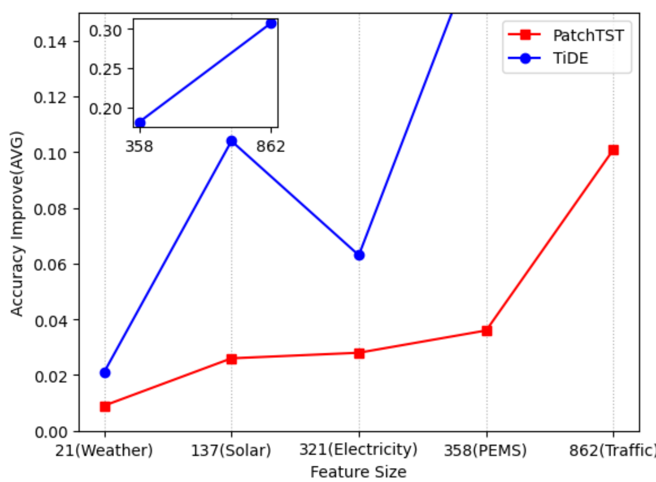


Fig. 5. Accuracy improvement versus number of features. We computed the average MSE improvement of MCformer compared to the single-channel strategies TiDE and PatchTST across different channel numbers. The results indicate that as the number of channels increases, the performance improvement of MCformer gradually becomes more significant. This suggests that MCformer can effectively capture dependencies between Multivariate data, thereby enhancing predictive performance.

Channel Fusion Quantity Blending channels may compro-

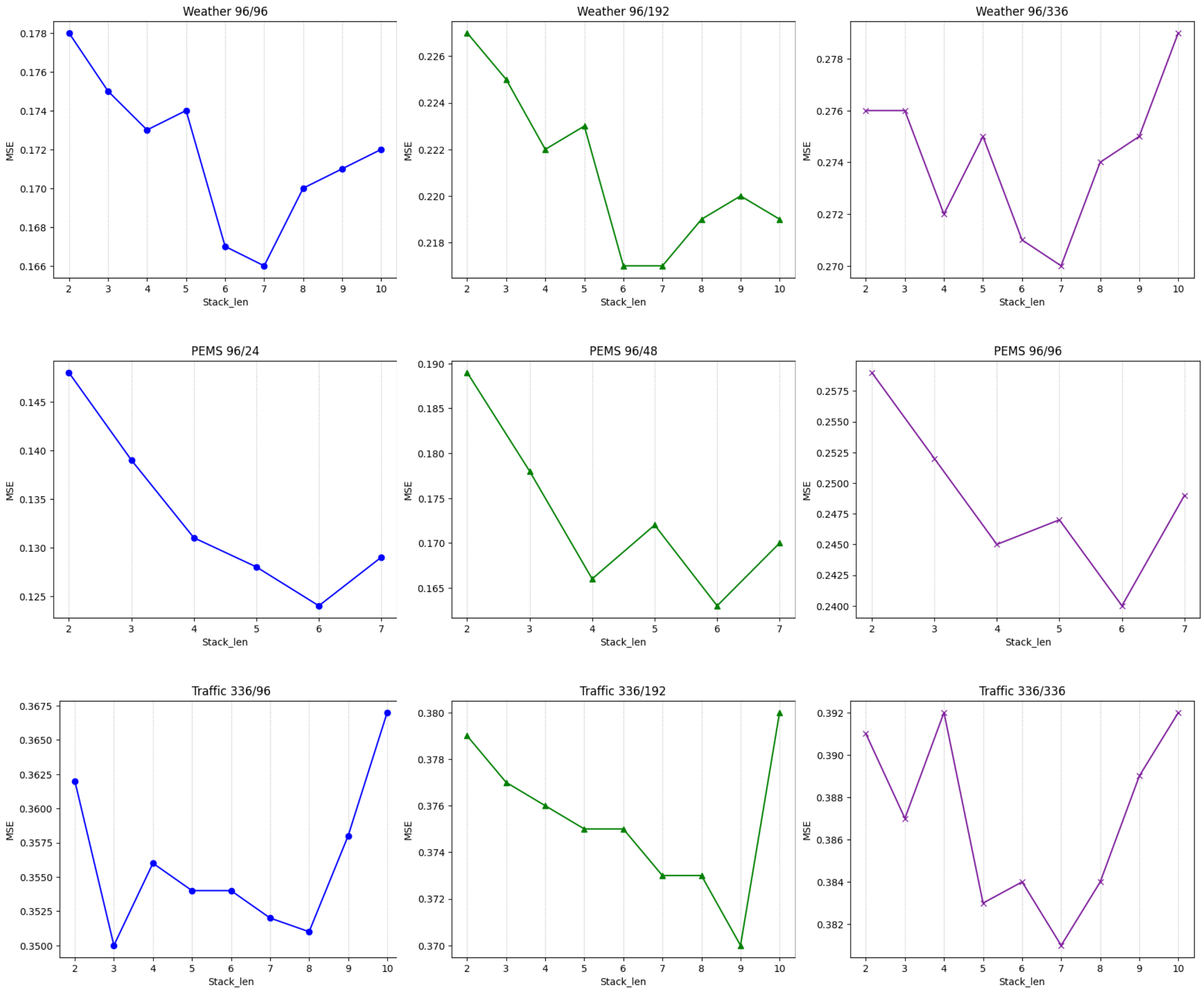


Fig. 6. Forecasting performance (MSE) of Weather, PEMS and Traffic datasets with different look-back window and forecasting window.For Traffic, the look-back window is 336, while for others, it's 96. We use forecasting lengths $h \in \{24, 48, 96\}$ for PEMS,and $h \in \{96, 192, 336\}$ for others.

mise the long-term features of the sequence, as demonstrated by PatchTST and PETFormer. Therefore, we need to investigate the impact of channel quantity on model performance. In Fig. 6, it is evident that model performance improves with a certain number of channels. However, on the Traffic, when the number of channels reaches 9, model performance starts to decline. When the number of channels reaches 10, the performance of blended channels even falls below that of the single-channel strategy. Similar trends are observed in the PEMS and Weather datasets, where the MSE initially decreases and then increases as the amount of mixed channel data grows. This suggests that our model effectively learns inter-channel correlations while minimizing the disruption of the model's ability to capture long-term features.

E. Visualiztion Analysis

Correlation analysis is a crucial method for measuring the degree of relationship between two variables, widely applied in the field of time series analysis. We attempt to analyze and evaluate the correlation of channel-wise information in the dataset by visualizing the correlation changes, which provides an explanation for the effectiveness of our model. The temporal variation of inter-channel correlation reflects changes in the correlated features among multiple channels. As shown in the Fig. 7, we utilize the same look-back window size as the forecasting to analyze the inter-channel correlation in the time series. Since the number of channels in the dataset exceeds 20, we randomly sampled channels for analysis. In the Electricity dataset, we observe that the inter-channel correlation changes over time, and there are significant variations in the correlation among some channels. This indicates that, over time, inter-channel dependency information, like long-term information, exhibits diversity and requires increased attention during modeling. The comparison between the real and predicted values shows that our model fits the correlation changes between the real values very well.

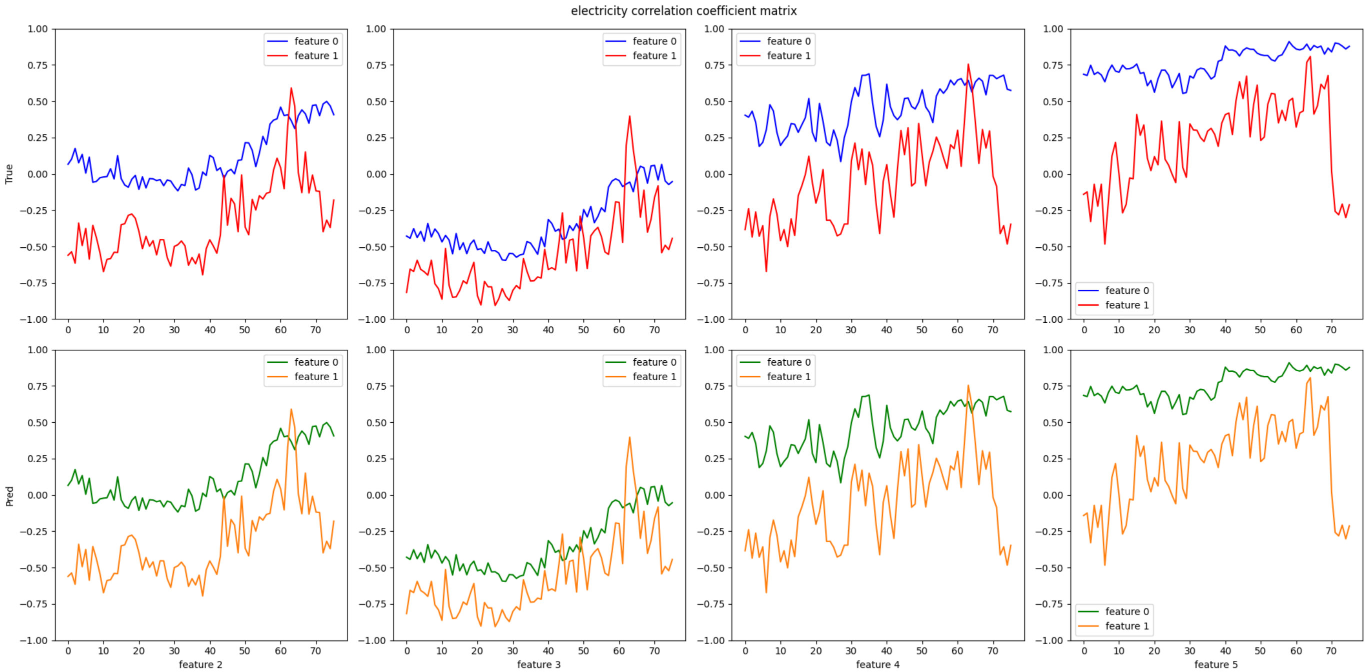


Fig. 7. Visualization of correlation variation of certain characteristics of Electricity dataset. Selected a subset of features from the Electricity dataset for temporal correlation analysis. We randomly sampled 6 features from the Electricity dataset. The four subplots above show the correlation changes between features 0 and 1 and features 2, 3, 4, and 5 in the real values. The predicted values are shown below. As can be clearly seen in the figure, the temporal correlation of different features is non-stationary. Our model, however, fits the correlation changes of the real values very well in terms of prediction values.

V. FUTURE WORK

Based on the current research, our future work can evolve in two directions to further advance the field of multi-channel time-series forecasting. Firstly, we can explore more complex Mixed Channels strategies by finely tuning the combination of channels to further enhance the model's performance in handling multi-channel time-series data. This may involve in-depth analysis of channel correlations and optimization of model architecture to achieve more accurate long-term feature modeling.

Furthermore, interpretability and explainability of the model are also crucial directions for future research. Enhancing our understanding of the model's focus on different channels and its decision-making process during forecasting can increase trust in the model's forecast results, improving its acceptability in practical applications.

In summary, future work should delve deeper into refining Mixed Channels strategies and enhancing model interpretability. This will contribute to further advancing and applying multi-channel time-series forecasting in various domains.

VI. CONCLUSION

In this paper, we propose a Multivariate time series forecasting model, MCformer, which leverages a mixed Multivariate feature. MCformer effectively addresses the issue of performance degradation caused by the disruption of channel information to long-term features by incorporating a limited set of mixed multi-channel data while preserving the advantages of the single-channel strategy in expanding the dataset. In experiments, MCformer consistently outperformed other models across all datasets. Furthermore, we conducted an in-depth analysis of the impact of the number of fused channels on the model. In the future, we plan to further explore the various effects of multi-channel features on time series analysis tasks.

REFERENCES

[1] R. A. Angryk, P. C. Martens, B. Aydin, D. Kempton, S. S. Mahajan, S. Basodi, A. Ahmadzadeh, X. Cai, S. Filali Boubrahimi, S. M. Hamdi *et al.*, "Multivariate time series dataset for space weather data analytics," *Scientific data*, vol. 7, no. 1, p. 227, 2020.

[2] J. Han, H. Liu, H. Zhu, H. Xiong, and D. Dou, "Joint air quality and weather prediction based on multi-adversarial spatiotemporal networks," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 35, no. 5, 2021, pp. 4081–4089.

[3] Y. Cheng, S. Wan, K.-K. R. Choo *et al.*, "Deep belief network for meteorological time series prediction in the internet of things," *IEEE internet of things journal*, vol. 6, no. 3, pp. 4369–4376, 2018.

[4] B. Ghosh, B. Basu, and M. O'Mahony, "Multivariate short-term traffic flow forecasting using time-series analysis," *IEEE transactions on intelligent transportation systems*, vol. 10, no. 2, pp. 246–254, 2009.

[5] R.-G. Cirstea, B. Yang, C. Guo, T. Kieu, and S. Pan, "Towards spatio-temporal aware traffic time series forecasting," in *2022 IEEE 38th International Conference on Data Engineering (ICDE)*. IEEE, 2022, pp. 2900–2913.

[6] Y. Qin, H. Luo, F. Zhao, Y. Fang, X. Tao, and C. Wang, "Spatio-temporal hierarchical mlp network for traffic forecasting," *Information Sciences*, vol. 632, pp. 543–554, 2023.

[7] F. Zhou, Q. Yang, K. Zhang, G. Trajcevski, T. Zhong, and A. Khokhar, "Reinforced spatiotemporal attentive graph neural networks for traffic forecasting," *IEEE Internet of Things Journal*, vol. 7, no. 7, pp. 6414–6428, 2020.

[8] S. F. Stefenon, L. O. Seman, L. S. Aquino, and L. dos Santos Coelho, "Wavelet-seq2seq-lstm with attention for time series forecasting of level of dams in hydroelectric power plants," *Energy*, vol. 274, p. 127350, 2023.

[9] J. Han, G. H. Lee, S. Park, J. Lee, and J. K. Choi, "A multivariate-time-series-prediction-based adaptive data transmission period control algorithm for iot networks," *IEEE Internet of Things Journal*, vol. 9, no. 1, pp. 419–436, 2021.

[10] Q. Hua, D. Yang, S. Qian, H. Hu, J. Cao, and G. Xue, "Kae-informer: A knowledge auto-embedding informer for forecasting long-term workloads of microservices," in *Proceedings of the ACM Web Conference 2023*, 2023, pp. 1551–1561.

[11] K. Yi, Q. Zhang, W. Fan, S. Wang, P. Wang, H. He, N. An, D. Lian, L. Cao, and Z. Niu, "Frequency-domain MLPs are more effective learners in time series forecasting," in *Thirty-seventh Conference on Neural Information Processing Systems*, 2023.

[12] Z. Ni, H. Yu, S. Liu, J. Li, and W. Lin, "Basisformer: Attention-based time series forecasting with learnable and interpretable basis," in *Advances in Neural Information Processing Systems*, 2023.

[13] S. Li, X. Jin, Y. Xuan, X. Zhou, W. Chen, Y.-X. Wang, and X. Yan, "Enhancing the locality and breaking the memory bottleneck of transformer on time series forecasting," *Advances in neural information processing systems*, vol. 32, 2019.

[14] H. Zhou, S. Zhang, J. Peng, S. Zhang, J. Li, H. Xiong, and W. Zhang, "Informer: Beyond efficient transformer for long sequence time-series forecasting," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 35, no. 12, 2021, pp. 11 106–11 115.

[15] N. Kitaev, L. Kaiser, and A. Levskaya, "Reformer: The efficient transformer," in *International Conference on Learning Representations*, 2019.

[16] H. Wu, J. Xu, J. Wang, and M. Long, "Autoformer: Decomposition transformers with auto-correlation for long-term series forecasting," *Advances in Neural Information Processing Systems*, vol. 34, pp. 22 419–22 430, 2021.

[17] T. Zhou, Z. Ma, Q. Wen, X. Wang, L. Sun, and R. Jin, "Fedformer: Frequency enhanced decomposed transformer for long-term series forecasting," in *International Conference on Machine Learning*. PMLR, 2022, pp. 27 268–27 286.

[18] M. A. Shabani, A. H. Abdi, L. Meng, and T. Sylvain, "Scaleformer: Iterative multi-scale refining transformers for time series forecasting," in *The Eleventh International Conference on Learning Representations*, 2022.

[19] S. Liu, H. Yu, C. Liao, J. Li, W. Lin, A. X. Liu, and S. Dustdar, "Pyraformer: Low-complexity pyramidal attention for long-range time series modeling and forecasting," in *International Conference on Learning Representations*, 2022.

[20] Z. Chen, D. Chen, X. Zhang, Z. Yuan, and X. Cheng, "Learning graph structures with transformer for multivariate time-series anomaly detection in iot," *IEEE Internet of Things Journal*, vol. 9, no. 12, pp. 9179–9189, 2021.

[21] Y. Zhang and J. Yan, "Crossformer: Transformer utilizing cross-dimension dependency for multivariate time series forecasting," in *International Conference on Learning Representations*, 2023.

[22] Z. Zhang, X. Wang, and Y. Gu, "Sageformer: Series-aware graph-enhanced transformers for multivariate time series forecasting," *arXiv preprint arXiv:2307.01616*, 2023.

[23] A. Zeng, M. Chen, L. Zhang, and Q. Xu, "Are transformers effective for time series forecasting?" in *Proceedings of the AAAI conference on artificial intelligence*, vol. 37, no. 9, 2023, pp. 11 121–11 128.

[24] Y. Nie, N. H. Nguyen, P. Sinthong, and J. Kalagnanam, "A time series is worth 64 words: Long-term forecasting with transformers," in *International Conference on Learning Representations*, 2023.

[25] L. Han, H. Ye, and D. Zhan, "The capacity and robustness trade-off: Revisiting the channel independent strategy for multivariate time series forecasting," *CoRR*, vol. abs/2304.05206, 2023.

[26] S. Lin, W. Lin, W. Wu, S. Wang, and Y. Wang, "Petformer: Long-term time series forecasting via placeholder-enhanced transformer," *arXiv preprint arXiv:2308.04791*, 2023.

[27] H. Wang, Y. Mo, N. Yin, H. Dai, B. Li, S. Fan, and S. Mo, "Dance of channel and sequence: An efficient attention-based approach for multivariate time series forecasting," *arXiv preprint arXiv:2312.06220*, 2023.

[28] Y. Liu, T. Hu, H. Zhang, H. Wu, S. Wang, L. Ma, and M. Long, "itransformer: Inverted transformers are effective for time series forecasting," *arXiv preprint arXiv:2310.06625*, 2023.

[29] A. Das, W. Kong, A. Leach, R. Sen, and R. Yu, "Long-term forecasting with tide: Time-series dense encoder," *arXiv preprint arXiv:2304.08424*, 2023.

[30] Z. C. Lipton, J. Berkowitz, and C. Elkan, "A critical review of recurrent neural networks for sequence learning," *arXiv preprint arXiv:1506.00019*, 2015.

[31] D. J. Bartholomew, "Time series analysis forecasting and control." 1971.

[32] G. Lai, W.-C. Chang, Y. Yang, and H. Liu, "Modeling long-and short-term temporal patterns with deep neural networks," in *The 41st international ACM SIGIR conference on research & development in information retrieval*, 2018, pp. 95–104.

[33] D. Salinas, V. Flunkert, J. Gasthaus, and T. Januschowski, "Deepar: Probabilistic forecasting with autoregressive recurrent networks," *International Journal of Forecasting*, vol. 36, no. 3, pp. 1181–1191, 2020.

[34] H. Wang, J. Peng, F. Huang, J. Wang, J. Chen, and Y. Xiao, "Micn: Multi-scale local and global context modeling for long-term series forecasting," in *The Eleventh International Conference on Learning Representations*, 2022.

[35] H. Wu, T. Hu, Y. Liu, H. Zhou, J. Wang, and M. Long, "Timesnet: Temporal 2d-variation modeling for general time series analysis," in *International Conference on Learning Representations*, 2023.

[36] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly *et al.*, "An image is worth 16x16 words: Transformers for image recognition at scale," in *International Conference on Learning Representations*, 2020.

[37] J. Gao, W. Hu, and Y. Chen, "Client: Cross-variable linear integrated enhanced transformer for multivariate long-term time series forecasting," *arXiv preprint arXiv:2305.18838*, 2023.

[38] T. Kim, J. Kim, Y. Tae, C. Park, J.-H. Choi, and J. Choo, "Reversible instance normalization for accurate time-series forecasting against distribution shift," in *International Conference on Learning Representations*, 2021. [Online]. Available: https://openreview.net/forum?id=cGDAkQo1C0p

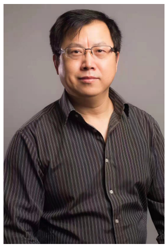
[39] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," *Advances in neural information processing systems*, vol. 30, 2017.

[40] M. Liu, A. Zeng, M. Chen, Z. Xu, Q. Lai, L. Ma, and Q. Xu, "Scinet: Time series modeling and forecasting with sample convolution and interaction," *Advances in Neural Information Processing Systems*, vol. 35, pp. 5816–5828, 2022.

Wenyong Han received B.S. degree from the School of Computer Science, University of South Chinain 2020. He has been pursuing his Master's degree in the School of Computer Science, University of South China since 2022.

His current research interests include neural networks and time series analysis.

Tao Zhu received his B.E. degree from Central South University, Changsha, China, and Ph.D. from University of Science and Technology of China, Hefei, China, in 2009 and 2015 respectively. He is currently an associate professor at University of South China, Hengyang, China. He is the principal investigator of several projects funded by the National Natural Science Foundation of China and Science Foundation of Hunan Province etc. He is now the Chair of IEEE CIS Smart World Technical Committee Task Force on "User-Centred Smart Systems". His research interests include IoT, pervasive computing, assisted living and evolutionary computation.



Liming Chen is a Professor of School of Computer Science and Technology, Dalian University of Technology, China. His current research interests include pervasive computing, data analytics, artificial intelligence and user centered intelligent systems and their applications in health care and cybersecurity. He has published over 250 papers in the aforementioned areas. Liming is an IET Fellow and a Senior Member of IEEE.

Huansheng Ning received his B.S. degree from Anhui University in 1996 and his Ph.D. degree from Beihang University in 2001. He is currently a Professor and Vice Dean with the School of Computer and Communication Engineering, University of Science and Technology Beijing and China and Beijing Engineering Research Center for Cyberspace Data Analysis and Applications, China, and the founder and principal at the Cybermatics and Cyberspace International Science and Technology Cooperation Base. He has authored several books and over 70 papers in journals and at international conferences/workshops. He has been the Associate Editor of the IEEE Systems Journal and IEEE Internet of Things Journal, Chairman (2012) and Executive Chairman (2013) of the program committee at the IEEE International Internet of Things Conference, and the Co-Executive Chairman of the 2013 International Cyber Technology Conference and the 2015 Smart World Congress. His awards include the IEEE Computer Society Meritorious Service Award and the IEEE Computer Society Golden Core Member Award. His current research interests include the Internet of Things, Cyber Physical Social Systems, electromagnetic sensing and computing.

Yang Luo attained a Bachelor's degree in Mathematics from the Department of Mathematics at Hunan Normal University in 1983. Subsequently, in 2005, he earned a Master's degree in Software Engineering from the School of Information Science and Engineering at Central South University,Hunan, China. Presently, he holds the position of Professor at University of South China, Hengyang, China , specializing primarily in the fields of image processing and software engineering.

Yaping Wan received his B.S. degree from Huazhong University of Science&Techno logy(HUST) in 2004 and his Ph.D. degree from HUST in 2009. He is currently a Professor and Dean with the School of Computer, University of South China and the International Cooperation Research Center for Medical Big Data of Hunan Province. He has authored several books and over 40 papers in journals and at international conferences/workshops. He has been the Workshop Chairman (2022) at the 16th IEEE International Conference on Big Data Science and Engineering, and the Session Chairman (2021,2022) of Asian Conference on Artificial Intelligence Technology. His current research interests include intelligent nuclear security, big data analysis and causal inference, high-reliability computing and security evaluation.