2025 International Conference on Intelligent Computing
July 26-29, Ningbo, China
https://www.ic-icc.cn/2025/index.php

CAAT: Channel-Aggregated Attention Transformer for Efficient Multivariate Time Series Forecasting

Wenhao Tang[1][0009-0002-8018-0154] , Wanli Zhao[2][0009-0007-3661-3369] , Fang Mei

[1]✉[0009-0006-3051-4610]

[1]Jiangxi Agricultural University, School of Computer and Information Engineering, Nanchang 330045, China
[2]Beijing Forestry University, School of technology, Beijing 100083, China

✉ Corresponding author: meifang@jxau.edu.cn
&These authors contributed equally to this work and should be considered co-first authors.

Abstract. Multivariate Time Series Forecasting (MTSF) holds significant application value in finance, energy, transportation, and other domains. Existing Transformer-based approaches typically face two critical challenges: (1) The computational complexity of cross-channel modeling grows quadratically with the number of channels, and (2) Noisy interference in cross-channel information leads to inefficient dependency modeling. This paper proposes a lightweight model named CAAT (Channel-Aggregated Attention Transformer) that achieves efficient forecasting through a Channel-Aggregation Module (CAM) and spatio-temporally decoupled attention mechanisms. The CAAT framework first compresses multivariate sequences into latent representations via MLPs, followed by saliency-based probabilistic sampling to select high signal-to-noise ratio channel features. Subsequently, the aggregated channel features are injected into the temporal dimension, enabling joint modeling of cross-temporal and cross-channel dependencies through temporal-axis attention mechanisms alone. Experimental results demonstrate that CAAT achieves significant improvements in prediction accuracy compared to baseline methods.

Keywords: MTSF, Transformer, Spatiotemporal Decoupled Attention, Channel Aggregation Module.

1 Introduction

Multivariate Time Series Forecasting (MTSF) is a core task in data mining and forecasting, aiming to accurately predict future trends using multiple historical time series. It plays an important role in applications such as financial market prediction, building energy management systems, and traffic flow control. In recent years, Transformer models [1] have excelled on large-scale datasets and high-channel dimensional scenarios owing to their powerful long-range dependency modeling capabilities. However, a

key challenge in current research is how to achieve efficient modeling of both temporal and cross-channel dependencies while retaining the advantages of Transformer models.

Existing Transformer methods mainly focus on temporal modeling while often overlooking the rich inter-channel dependencies present in multivariate series. In fact, cross-channel information is crucial for enhancing model generalization and forecasting accuracy. Methods that solely focus on temporal dependencies struggle to effectively exploit these inter-channel relationships, whereas approaches that attempt to model cross-channel interactions often suffer from data heterogeneity and low signal-to-noise ratios, leading to reduced generalization ability. Moreover, when the number of channels is large, models such as iTransformer [2] and PatchTST [3] exhibit computational complexity that grows quadratically with the number of channels, making efficient cross-channel modeling in practical applications challenging.

To address the aforementioned issues, we propose a novel model—CAAT. CAAT retains the long-range modeling advantages of Transformers while achieving efficient modeling of both cross-temporal and cross-channel dependencies through a lightweight Channel Aggregation Module (CAM). The main contributions of this work can be summarized as follows:

1.Channel Aggregation: We propose an efficient channel aggregation method that addresses the heterogeneity and low signal-to-noise ratio issues in multivariate series, achieving high-quality extraction of cross-channel information.

2.Joint Cross-Temporal and Cross-Channel Dependency Modeling: By injecting the aggregated cross-channel information into the temporal dimension, the model effectively leverages the attention mechanism's strength in long-range dependency modeling, thereby resolving the problem of rapidly increasing computational complexity with the number of channels found in traditional methods.

3. Experiments conducted on seven widely-used real-world multivariate time series datasets demonstrate that CAAT achieves significant improvements in both forecasting accuracy and model generalization, validating its effectiveness and superiority in multivariate time series forecasting tasks.

2 Related Work

In recent years, numerous innovative studies have emerged in the field of time series forecasting, primarily focusing on optimizing model architectures and innovating data representation methods. These studies have not only enriched time series modeling approaches but also significantly improved forecasting performance in practical applications.

Transformer-based models have demonstrated strong performance in multivariate time series forecasting. For example, iTransformer [2] uses an inverted architecture by treating each variable's history as an independent token to capture inter-variable correlations via self-attention. PatchTST [3] divides a time series into patches—similar to the Vision Transformer approach—to reduce computational costs and improve generalization through self-supervised pretraining. Meanwhile, Crossformer [4] employs a dual-branch attention mechanism for modeling temporal dependencies and variable

correlations, though it suffers from high computational complexity, and FEDformer [5] combines frequency domain decomposition with a Transformer design to better capture cycles and trends.

On the lightweight side, models like DLinear [6] decompose time series into trend and residuals using simple linear layers, showing that simpler models can excel in long-term forecasting. TiDE [7] enhances forecasting accuracy with MLPs and a dynamic feature interaction module for capturing inter-variable dependencies. SCINet [8] extracts multi-scale patterns via multi-resolution downsampling, and TSMixer [9] combines MLPs with temporal mixing operations. Autoformer [10] improves long sequence forecasting using autocorrelation mechanisms with deep decomposition strategies, while LightTS [11] is recognized for its efficiency in resource-constrained scenarios.

Additionally, the Temporal Fusion Transformer [12] offers a compelling balance of accuracy and interpretability. TimesNet [13] introduces a multi-periodicity-aware architecture that transforms time series into 2D space to capture intra-period and inter-period variations effectively. Stationary [14] addresses non-stationarity in time series by learning stationary representations, improving generalization across different domains.

However, existing Transformer-based methods mainly focus on modeling dependencies along the temporal dimension (such as long-term dependencies and periodic patterns). Despite employing various architectural optimizations to enhance computational efficiency, they do not explicitly address the modeling of inter-channel dependencies. In multivariate time series, the interactions between channels provide rich information gains that can further improve forecasting performance when fully exploited.

3 Method

3.1 Model architecture

Our model is based on the Transformer [1] architecture and introduces two key innovations: Channel Aggregation Module (CAM) - featuring MLP-based feature compression and saliency-aware channel selection; Cross-dimensional information injection mechanism - embedding fused channel features into temporal dimension. This design enables efficient joint modeling of cross-time and cross-channel dependencies in multivariate time series, with the innovative architecture comprising the following key elements.

The model first uses a MLP to compress cross-channel sequences into a low - dimensional latent space. Secondly, based on the saliency probability of the channel aggregation module, key channel information is dynamically sampled, and cross - channel features are adaptively weighted and fused through learnable weight parameters. Finally, the fused features with a high signal - to - noise ratio are injected into the time dimension. By simply applying a lightweight attention mechanism on the time axis, long - term joint modeling of cross-temporal and cross-channel dependencies can be modeled synchronously. Specifically, given an input sequence$X \in \mathbb{R}^{C\times L}$(C is the variable dimension,L is the length of the backtracking window), it is reconstructed into a patch sequence $\mathbb{R}^{C\times N\times D}$(N is the number of blocks, D is the intra-block dimension).

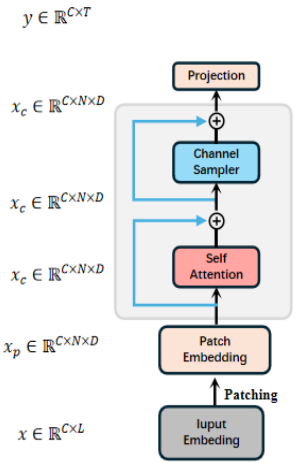

Fig. 1. The CAAT architecture demonstrates the tensor transformation process.

The embedded sequences from all channels are fed into a Transformer encoder with shared parameters, where the multi-head self-attention mechanism captures local temporal patterns, significantly reducing computational complexity. The hidden states output by the encoder are then passed through independent linear decoding layers to predict the future T steps for each channel, and the predictions are finally aggregated to form the output $Y \in \mathbb{R}^{C\times T}$. This approach, by compressing the sequence length through patching, processing channels independently, and leveraging a shared-parameter encoder, achieves efficient computation while preserving local temporal features, making it well-suited for high-dimensional long sequence forecasting scenarios.

Patch Embedding. Given an input feature $x_{enc} \in \mathbb{R}^{C\times N\times D}$,and an optional labeled feature $x_{mark} \in \mathbb{R}^{C\times N\times D}$, where:C is the variable dimension,N is the number of chunks, and D is the dimension of each chunk. First, if x_{mark} is non-empty, it is spliced with x_{enc} along the last dimension:

$$\mathbf{e}_{\text{in}} = \text{concat}\,(\mathbf{x}_{\text{enc}}, \mathbf{x}_{\text{mark}}, \text{dim} = -1) \tag{1}$$

where the spliced tensore$_{\text{in}} \in \mathbb{R}^{C\times N\times 2D}$ has double the intra-block dimension. The spliced feature tensor is then rearranged to accommodate subsequent processing layers. Change the shape of the tensor from $\mathbb{R}^{C\times N\times 2D}$ to $\mathbb{R}^{(C\times N)\times 2D}$:

$$\mathbf{e}_{in} = \text{rearrange}(\mathbf{e}_{in}, \, 'cnd \rightarrow (cn)d') \tag{2}$$

Next, the latent projection layer W_{proj} is applied, which maps the input tensor to the latent space via a linear transformation, generating the embedding representation $e_{out} \in R^{(C\times N)\times D_{lotent}}$:

$$e_{out} = W_{proj}(e_{in}) \tag{3}$$

Ultimately, the output contains three elements:C is the variable dimension, Nis the number of chunks, and e_{out} is the embedding representation in potential space.

Self Attention. In order to further enhance the association modeling capability between features, this method introduces a self-attention mechanism in the feature extraction module. Specifically, we adopt a multi-head self-attention structure similar to that in Transformer for weighted aggregation of the extracted image or voxel features. For the input feature sequence $W_Q, W_K, W_V \in \mathbb{R}^{d\times d'}$, the Query, Key and Value matrices are first obtained by linear transformation:

$$\mathbf{Q} = \mathbf{X}\mathbf{W}_Q, \mathbf{K} = \mathbf{X}\mathbf{W}_K, \mathbf{V} = \mathbf{X}\mathbf{W}_V \tag{4}$$

where$\mathbf{X} \in R^{N\times d}$ is the learnable parameter matrix. The attentional output is computed by scaling the dot product:

$$\text{Attention}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \text{softmax}\left(\frac{\mathbf{Q}\mathbf{K}^T}{\sqrt{d'}}\right)\mathbf{V} \tag{5}$$

The final output features are weighted aggregated attention results enhanced by residual connectivity and feed-forward networks. In this work, the Self Attention module is integrated after CAAT to enhance the information interaction between features of different spatial locations or different dimensions, thus improving the overall representation capability and downstream task performance.

3.2 Channel Aggregation Module

The Softmax function in the Transformer architecture is able to transform the raw scores into a probability distribution such that each output value is between 0 and 1 and all output values sum to 1. Based on this fact, the sampling module of CAAT relies on Softmax for the modeling of the probability of significance.

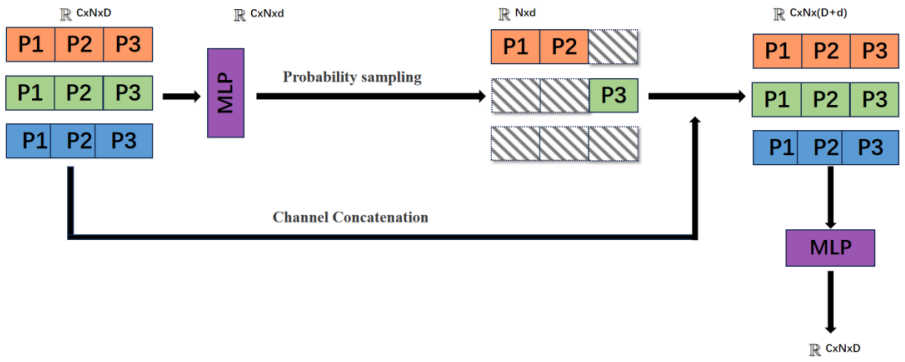

Fig. 2. Module architecture diagram.

For the specific implementation process, first for each tensor $x \in \mathbb{R}^{CxNxD}$ of the input feedforward neural network (C is the total number of channels of the multivariate time series, N is the total number of patches per channel, and D is the number of feature dimensions contained in each patch, which were extracted in the previous hidden layers). The deep features are then extracted at the MLP of the first layer in each FFN architecture and downsampled to obtain the core feature $F \in \mathbb{R}^{CxNxd}$ (d is the feature dimension obtained after downsampling, d<=D). Then the core features F are softmax normalized to obtain the significance weights:

$$P_{c,n,i} = \frac{\exp(F_{c,n,i})}{\sum_{j=1}^{d} \exp(F_{c,n,j})}, i = 1,2,\dots,d \tag{6}$$

where $P_{c,n} \in \mathbb{R}^{d}$ is a probability distribution denoting the significance of each feature dimension of channel c at time step n. At this point we wish to sample the significance probability of all channel information in this feature space to reduce its features from d-dimension to 1-dimension, yielding $Y \in \mathbb{R}^{CxN}$: a random selection of one of the most significant channels from the d-dimensional channels:

$$i^{*} \sim \text{Multinomial}(P_{c,n}) \tag{7}$$

Where i^{*} is the index of the most important feature drawn from the d-dimensional features, and the heaviest feature retained after polynomial sampling (randomly selecting an element of the sample according to a given probability distribution) is $Y_{c,n} = F_{c,n,i}*$ It is clear that the introduction of random generator counts helps to improve the model's generalization ability during the training phase. At this point only one most important feature is retained for each of the N patches under each time step.

$$Y_{c,n} = \sum_{i=1}^{d} P_{c,n,i} \cdot F_{c,n,i} \tag{8}$$

This space contains the features that have been extracted for all channels during this round of training. We then dynamically selected one of the most representative channels from the original D-dimensional features, thus realizing the downscaling effect of significance probability sampling. This channel was then copied into C number of copies and stacked channel-by-channel with the original channel information to obtain $\mathbb{R}^{CxNx(D+d)}$. Finally, after a second MLP downsampling, channel information of the same dimension $\mathbb{R}^{CxNxD}$ as before the input module is obtained. This method of combining the feature information enhanced by learning with the original input allows the model to learn both global and local feature information during the training process, which avoids the limitation of single reliance on the original features, and also avoids the oversimplification problem that may be caused by relying on the post-mining features alone, and effectively enhances the model's representation capability and robustness.

3.3 Cross-dimensional information fusion mechanisms

While the Channel Aggregation Module (CAM) effectively extracts high-quality inter-channel features, integrating this information into the temporal modeling pipeline remains a key design challenge. To address this, CAAT introduces a dedicated Temporal-Channel Fusion Layer, which merges the strengths of Transformer-based temporal attention with the representational advantages of CAM, achieving efficient and joint modeling across both dimensions.

Specifically, the fusion process starts with a conventional Transformer attention layer operating along the temporal axis of the input tensor $X \in \mathbb{R}^{C\times N\times D}$, where C is the number of channels, N is the number of time patches, and D is the embedding dimension. The output of this attention layer undergoes residual connection and layer normalization to yield an intermediate representation X', capturing temporal dependencies.

Next, this representation is rearranged into a format suitable for channel-wise processing and passed into the Channel Aggregation Module. The CAM produces a compact and informative cross-channel feature representation that emphasizes salient patterns across variables. These channel-wise features are then injected back into the temporal sequence through concatenation and projection, allowing their influence to propagate during subsequent temporal modeling.

A second residual connection and normalization step ensures training stability and preserves both global and local feature interactions. This fusion mechanism allows the model to simultaneously capture long-range temporal patterns and latent inter-channel structures, without incurring additional computational complexity as the number of channels increases.

This design seamlessly bridges the temporal modeling backbone and the channel aggregation branch, forming a unified architecture that enables efficient and expressive multivariate time series forecasting.

3.4 Loss Function

We choose to use the MSE as the underlying loss function, and for the i th time series, calculate the L2 paradigm squares of its prediction segment $\hat{\boldsymbol{x}}_{L+1:L+T}^{(i)}$ and its true segment$\boldsymbol{x}_{L+1:L+T}^{(i)}$:

$$\text{Loss}^{(i)} = \left\| \hat{\boldsymbol{x}}_{L+1:L+T}^{(i)} - \boldsymbol{x}_{L+1:L+T}^{(i)} \right\|_2^2 \tag{9}$$

The losses of the M time series were averaged to obtain the overall target loss:

$$\mathcal{L} = E_x \left(\frac{1}{M} \sum_{i=1}^{M} \text{Loss}^{(i)} \right) \tag{10}$$

where E_x denotes the expectation of the data distribution (or batch samples), which is often approximated by batch averaging in practical training.

4 Experiments

4.1 Basic settings

Datasets. In this study, seven representative time series datasets are used for systematic evaluation, including three classical domain datasets (covering practical application scenarios such as meteorology, traffic flow, and power load) and four ETT benchmark datasets (ETTh1, ETTh2, ETTm1, ETTm2) [10]. The classical datasets, with larger data size and richer feature dimensions, can effectively verify the generalization ability of the model in complex real-world scenarios; while the ETT datasets, as a standard test set in the field of time series forecasting, facilitate direct performance comparison with existing studies.

Model Variants. We select a series of Transformer-based models as baselines, including iTransformer [2], PatchTST [3], Crossformer [4], TSMixer [9]and CAAT strictly follows the experimental protocols outlined in the original papers to ensure fair and consistent comparisons.

Specifically, the input sequence length T∈{96,192,336,720} is uniformly applied across all models to comprehensively evaluate performance under short-, medium-, and long-term forecasting scenarios. During training, we adopt a fixed configuration: a batch size of 128 and a maximum of 100 training epochs. All models share the same network architecture hyperparameters—hidden dimension of 512, 8 attention heads, 2 encoder layers, and 1 decoder layer. For performance evaluation, we employ MSE and MAE as the primary quantitative metrics.

4.2 Evaluation of Predictive Capability

After end-to-end supervised training, CAAT is evaluated against other baseline models on test datasets. All data are standardized, and RevIN [15] is employed to eliminate

scale discrepancies. A dynamic patch adjustment strategy is adopted to accommodate the characteristics of different datasets (For example, a patch length of 8 for the Electricity dataset and 12 for the ETT dataset). The evaluation on the validation set is conducted using MSE and MAE as performance metrics.

CAAT achieves the lowest prediction errors across all evaluated datasets, significantly outperforming advanced models such as PatchTST and Transformer, and demonstrating superior generalization capabilities. On datasets like ETT, which exhibit

Table 1. Comparison results between CAAT and other baseline models.

Models		CAAT		iTransformer		PatchTST		Crossformer		TSMixer	
Metric		MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE
ETTh1	96	0.376	0.398	0.386	0.405	0.394	0.406	0.423	0.448	0.401	0.412
	192	0.425	0.428	0.441	0.436	0.440	0.435	0.471	0.474	0.452	0.442
	336	0.467	0.447	0.487	0.458	0.491	0.462	0.570	0.546	0.492	0.463
	720	0.449	0.457	0.503	0.491	0.487	0.479	0.653	0.621	0.507	0.490
ETTh2	96	0.289	0.341	0.297	0.349	0.288	0.340	0.745	0.584	0.319	0.361
	192	0.370	0.392	0.380	0.400	0.376	0.395	0.877	0.656	0.402	0.410
	336	0.408	0.424	0.428	0.432	0.440	0.451	1.043	0.731	0.444	0.446
	720	0.420	0.441	0.427	0.445	0.436	0.453	1.104	0.763	0.441	0.450
ETTm1	96	0.322	0.361	0.334	0.368	0.329	0.365	0.404	0.426	0.323	0.363
	192	0.367	0.384	0.377	0.391	0.380	0.394	0.450	0.451	0.376	0.392
	336	0.399	0.408	0.426	0.420	0.400	0.410	0.532	0.515	0.407	0.413
	720	0.459	0.442	0.491	0.459	0.475	0.453	0.666	0.589	0.485	0.459
ETTm2	96	0.176	0.261	0.180	0.264	0.184	0.264	0.287	0.366	0.182	0.266
	192	0.242	0.303	0.250	0.309	0.246	0.306	0.414	0.492	0.249	0.309
	336	0.306	0.345	0.311	0.348	0.308	0.346	0.597	0.542	0.309	0.347
	720	0.405	0.402	0.412	0.407	0.409	0.402	1.730	1.042	0.416	0.408
Weather	96	0.166	0.208	0.174	0.214	0.176	0.217	0.158	0.230	0.166	0.210
	192	0.215	0.248	0.221	0.254	0.221	0.256	0.206	0.277	0.215	0.256
	336	0.276	0.287	0.278	0.296	0.275	0.296	0.272	0.335	0.287	0.300
	720	0.351	0.346	0.358	0.347	0.352	0.346	0.398	0.418	0.355	0.348
Electricity	96	0.143	0.236	0.148	0.240	0.164	0.251	0.219	0.314	0.157	0.260
	192	0.160	0.252	0.162	0.253	0.173	0.262	0.231	0.322	0.173	0.274
	336	0.174	0.270	0.178	0.269	0.190	0.279	0.246	0.337	0.192	0.295
	720	0.227	0.311	0.225	0.317	0.230	0.313	0.280	0.363	0.223	0.318
Traffic	96	0.388	0.260	0.395	0.268	0.427	0.272	0.522	0.290	0.493	0.336
	192	0.407	0.266	0.417	0.276	0.454	0.289	0.530	0.293	0.497	0.351
	336	0.424	0.273	0.433	0.283	0.450	0.282	0.558	0.305	0.528	0.361
	720	0.452	0.290	0.467	0.302	0.484	0.301	0.589	0.328	0.569	0.380

strong periodicity and structural regularity, CAAT effectively models long-term dependencies and periodic patterns, yielding highly accurate predictions and showcasing its strong ability to extract standard temporal features.

More notably, on complex real-world datasets such as Electricity, characterized by high volatility and pronounced non-stationarity, CAAT maintains consistently stable performance.These results strongly indicate that CAAT is not only well-suited for structured sequence prediction tasks but also highly robust and adaptive in handling challenging scenarios with high variability and dynamic patterns.

4.3 Ablation analysis

Module Ablation. To assess the impact of the channel sampling module, we conducted a two-group ablation study: the control group used the full CAAT model, while the experimental group employed a version without channel sampling. In this version, input features undergo only nonlinear transformation via the FFN, without cross-channel feature selection.

Results on the ETTh1 dataset show that removing the sampling module leads to a 12.7% increase in MSE and a 9.3% increase in MAE, indicating the module's critical role in improving prediction accuracy through effective channel-wise feature filtering.

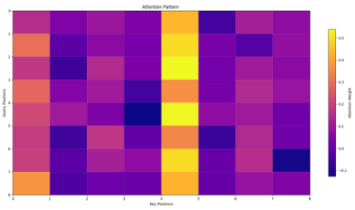

Fig. 3. Attention map of the CAAT .

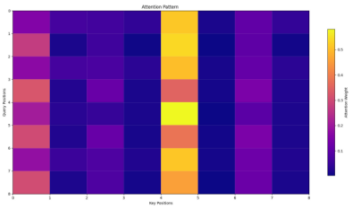

Fig. 4. Attention map of the CAAT after ablation.

Two figures compare attention maps of the full and ablated CAAT. The ablated model shows uniform, narrow attention, lacking global integration. The full model preserves periodicity and disperses attention more broadly, indicating enhanced global awareness via channel sampling.

This module enhances performance through the following approaches: (1) reinforcing key temporal patterns while suppressing noise, and (2) selecting high signal-to-noise ratio channels. It delivers feature representation gains of up to 11%, demonstrating particularly outstanding performance in modeling long-term dependencies of high-frequency signals.

Sampling Strategies Exploration. To investigate different channel sampling strategies, we designed an ablation study with three setups: uniform sampling (equal selection probability for all channels), random sampling (averaged features with random channel selection), and significance probability sampling (Softmax-based dynamic weighting).Unlike the main experiments that focus on overall performance, this study compares MAE and MSE across sampling methods. We evaluate them on ETTh1 and

ETTh2 datasets using four lookback lengths: T ∈ {96, 192, 336, 720}, to assess their effectiveness under different temporal resolutions.
Comparative analysis shows that RP-Sampling suffers from performance fluctuations due to inconsistent channel selection, while M-Sampling is more stable but lacks sensitivity to key features. In contrast, the Significance Probability Sampling used in the main model dynamically weights important features via Softmax. On the ETTh2 dataset (horizon 336), it achieves an MSE of 0.408, outperforming M-Sampling (0.422) and RP-Sampling (0.417), highlighting its ability to adaptively enhance salient features.

Table 2. The three sampling methods.

Dataset		M-Sampling		RP-Sampling		SP-Sampling	
Metric		MSE	MAE	MSE	MAE	MSE	MAE
ETTh1	96	0.377	0.398	0.389	0.400	0.376	0.398
	192	0.428	0.431	0.426	0.429	0.425	0.428
	336	0.467	0.448	0.465	0.449	0.461	0.446
	720	0.457	0.464	0.461	0.462	0.449	0.457
ETTh2	96	0.291	0.344	0.287	0.342	0.289	0.341
	192	0.374	0.396	0.375	0.397	0.370	0.392
	336	0.422	0.431	0.418	0.429	0.408	0.424
	720	0.421	0.445	0.423	0.447	0.420	0.441

5 Conclusion

The model proposed in this paper fully exploits the Transformer's capability in modeling long-range dependencies. It introduces an innovative framework that first maps multivariate time series into a latent space via a multilayer perceptron, then applies a significance probability-based sampling strategy to extract informative cross-channel features. These selected features are subsequently injected into the temporal dimension for attention-based modeling.

This design effectively mitigates the challenges posed by complex inter-channel dependencies and low signal-to-noise ratios in traditional approaches, while also avoiding the steep computational cost associated with increasing channel dimensionality. Extensive experiments demonstrate that the proposed method consistently outperforms strong baselines in both forecasting accuracy and generalization ability. Overall, this

work offers a novel and practical solution for efficient and effective multivariate time series forecasting.

Disclosure of Interests. The authors have no competing interests to declare that are relevant to the content of this article.

References

1. Vaswani, A., Shazeer, N., Parmar, N., et al.: Attention is all you need. Advances in Neural Information Processing Systems **30**, 5998-6008 (2017)
2. Arxiv, https://arxiv.org/abs/2310.06625, last accessed 2023/10/10
3. Arxiv, https://arxiv.org/abs/2211.14730, last accessed 2022/11/27
4. Wang, W., Chen, W., Qiu, Q., et al.: Crossformer++: A versatile vision transformer hinging on cross-scale attention. IEEE Transactions on Pattern Analysis and Machine Intelligence **46**(5), 3123-3136 (2023)
5. Zhou, T., Ma, Z., Wen, Q., et al.: FedFormer: Frequency enhanced decomposed transformer for long-term series forecasting. In: International Conference on Machine Learning, pp. 27268-27286. PMLR, Cambridge (2022)
6. Zeng, A., Chen, M., Zhang, L., et al.: Are transformers effective for time series forecasting? In: Proceedings of the AAAI Conference on Artificial Intelligence, pp. 11121-11128. AAAI Press, Palo Alto (2023)
7. Arxiv, https://arxiv.org/abs/2304.08424, last accessed 2023/04/17
8. Liu, M., Zeng, A., Chen, M., et al.: SCINet: Time series modeling and forecasting with sample convolution and interaction. Advances in Neural Information Processing Systems **35**, 5816-5828 (2022)
9. Ekambaram, V., Jati, A., Nguyen, N., et al.: TSMixer: Lightweight MLP-mixer model for multivariate time series forecasting. In: Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, pp. 459-469. ACM, New York (2023)
10. Wu, H., Xu, J., Wang, J., et al.: Autoformer: Decomposition transformers with auto-correlation for long-term series forecasting. Advances in Neural Information Processing Systems **34**, 22419-22430 (2021)
11. Zhang, Yuanhan, Kaiyang Zhou, and Ziwei Liu. "Neural prompt search." IEEE Transactions on Pattern Analysis and Machine Intelligence (2024).
12. Lim, Bryan, et al. "Temporal fusion transformers for interpretable multi-horizon time series forecasting." International Journal of Forecasting 37.4 (2021): 1748-1764.
13. Wu, Haixu, et al. "Timesnet: Temporal 2d-variation modeling for general time series analysis." arXiv preprint arXiv:2210.02186 (2022).
14. Liu, Y., Wu, H., Wang, J., et al.: Non-stationary transformers: Exploring the stationarity in time series forecasting. Advances in Neural Information Processing Systems **35**, 9881-9893 (2022)
15. Kim, Y., Choi, J.: Reversible instance normalization for accurate time-series forecasting against distribution shift. Advances in Neural Information Processing Systems 34, 16748–16760 (2021)