IProbeTrans: A Long-Term Series Forecasting Method Based on Self-Supervised Learning

1st Xumin Zuo#
College of Big Data and Internet
Shenzhen Technology University
Shenzhen, China
2110416016@email.szu.edu.cn

2nd Jiayi Wu#
College of Big Data and Internet
Shenzhen Technology University
Shenzhen, China
2210413001@email.szu.edu.cn

3rd Xiaoxing Yang
College of Big Data and Internet
Shenzhen Technology University
Shenzhen, China
yangxiaoxing@sztu.edu.cn

4th Heng Zhao*
College of Big Data and Internet
Shenzhen Technology University
Shenzhen, China
zhaoheng@sztu.edu.cn

5th Bingding Huang*
College of Big Data and Internet
Shenzhen Technology University
Shenzhen, China
huangbingding@sztu.edu.cn

2024 8th Asian Conference on Artificial Intelligence Technology (ACAIT) | 979-8-3315-1709-0/24/$31.00 ©2024 IEEE | DOI: 10.1109/ACAIT63902.2024.11021826

***Abstract*—Long-term time series forecasting (LTSF) is essential for domains like meteorology and finance, requiring accurate predictions over many time points. While Transformer-based models have shown promise in capturing long-term dependencies, they face challenges due to potential information loss and quadratic scaling of time complexity, impacting efficiency. Thus, this work proposes a forecasting method called iProbTrans, which uses inverted embedding and probsparse self-attention to capture temporal representations and correlations between variables. It outperforms transformer-based methods on multivariate datasets, reducing MSE and MAE by 9.6% and 6.6%, respectively, compared to the FEDformer baseline. A self-supervised learning approach is proposed, decomposing time series into patches and applying random masking to reduce computational costs. This model, trained on mean squared error loss for mask reconstruction, improves upon supervised learning by reducing MSE and MAE by 26.3% and 27.5%. It excels in capturing long-term dependencies, enhancing generalization, and offers a solution to the limitations of Transformer-based methods in LTSF.**

Keywords—long-term time series forecasting, deep learning, transformer; self-supervised learning

I. INTRODUCTION

Long-term series prediction is a challenging problem involving large data volumes, high dimensions, and long-term dependence. The main research directions of time series data are classification[1-3], anomaly detection[4-7], event prediction[8-10], and time series prediction [11-15]. With the development of data collection technology, the task has gradually evolved to use more historical data to predict the long-term future, i.e., Long-term Time Series Forecasting (LTSF) [16, 17]

In recent years, contrastive learning and Transformer-based models [18] have achieved good results in many long-term sequence prediction tasks. However, there are still some problems with the existing methods: (1) The time complexity of the Transformer model is O(L2), which means that the computational cost of the model increases dramatically as the length of the sequence increases, leading to a decrease in the efficiency of training and inference. (2) Although the self-attention mechanism of the Transformer can deal with long-distance dependencies to a certain extent, for long sequences, the problem of information loss may occur. (3) Transformer typically employs an encoder-decoder architecture, which may limit the model's ability to handle complex temporal dynamics and long-term dependencies in long-term series prediction. Therefore, new models need to be developed to deal with these problems.

Due to the Transformer's ability to capture remote dependencies, Vanilla Transformer was applied directly to time series data.[19]. However, it did not work well in long-series time-series prediction tasks because the Transformer relied on the self-attention mechanism to extract the semantic dependencies between pairs of elements. The self-attention calculation grew squarely with the input sequence length. Recently, two strategies were proposed to improve efficiency and reduce the O (L2) time and memory complexity of the Vanilla Transformer. On the one hand, LogTrans[20] and Pyraformer[21] introduced sparsity bias. Specifically, LogTrans uses a Log sparse mask to reduce computational complexity to O (LlogL). At the same time, Pyraformer employs pyramidal attention to capture hierarchical multi-scale time dependence in terms of time and memory complexity of O (L). On the other hand, Informer[16] and FEDformer [22] take advantage of the low-rank nature of the self-attention matrix. Informer proposes the ProbSparse self-attention mechanism to reduce the complexity to O (LlogL). PatchTST[18] converts time-series data into Vision-like data patch form in Transformer and adopts the channel-independence method. This means mapping each variable sequence to an independent embedding model rather than merging multiple variables into one embedding model as in the original multivariate time series modeling approach, achieving significantly notable results. All these researches primarily address problem 1, while problems 2 and 3 persist unresolved within the LSTF problem. To augment prediction capabilities, we address all these constraints and accomplish enhancements surpassing mere efficiency in our proposed model.

To this end, this work proposed a long-time series prediction method based on self-supervised learning. We applied the Probsparse self-attention mechanism, improved the Embedding method, and conducted extensive experiments. The main content and innovations contain the following two aspects:

- We propose the IProbTrans model to predict the long-term time series and demonstrate its potential to capture multivariate features.
- We present a self-supervised approach for representation learning in the model, utilizing the window-slicing technique to break down the time series into smaller patches.

This work is supported by the Project of the Educational Commission of Guangdong Province of China (No. 2022ZDJS113).

#These authors contributed equally to this work

*Corresponding author

979-8-3315-1709-0/24/$31.00 ©2024 IEEE

Authorized licensed use limited to: Shenzhen Technology University. Downloaded on June 12,2025 at 06:01:50 UTC from IEEE Xplore. Restrictions apply.

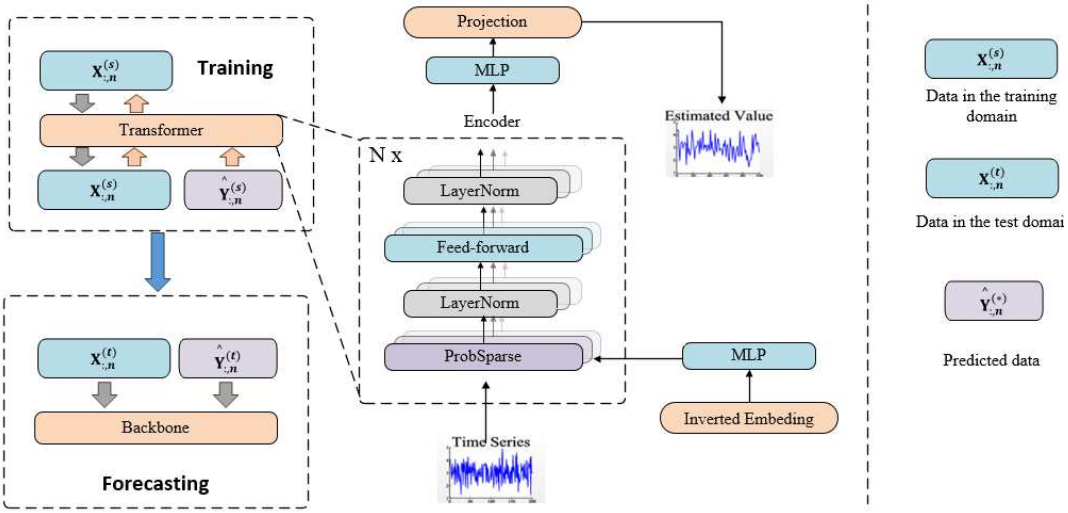

Fig. 1. Schematic diagram of IProbTrans

II. METHODOLOGY

A. IProbTrans

As shown in Fig. 1, our proposed Transformer-based model uses a simpler encoder-only architecture that includes Inverted Embedding, Transformer Encoder blocks, and Projection projections.

Inverted Embedding. For a multidimensional time series X of length T and N variables, it is assumed that denotes all variables simultaneously, and denotes the entire sequence of historical observations over the T length under a single variable. Considering that the latter has stronger semantics and relatively consistent units of measurement than the former, unlike the previous Embedding approach for, we use the Embedding layer to feature map each independently, and obtains feature representations of variables, where each variable representation implies the temporal change of the variable in the past time. Predicting a future sequence containing several specific variables from a known sequence can be simply formulated as (1).

$$\begin{aligned} \mathbf{h}_n^0 &= \text{Embedding}\ (\mathbf{X}_{:,n}), \\ \mathbf{H}^{l+1} &= \text{Encoder}\ (\mathbf{H}^l), l = 0, \cdots, L-1, \\ \hat{\mathbf{Y}}_{:,n} &= \text{Projection}\ (\mathbf{h}_n^L) \end{aligned} \tag{1}$$

Embedding and Projection are both implemented by Multi-Layer Perceptron (MLP).

Transformer Encoder. After undergoing Inverted Embedding, the feature representation of the time series will be applied across various layers of the Transformer module. Firstly, inter-variable information exchange is facilitated through the Probsparse self-attention mechanism. Layer normalization is employed to unify the feature distributions of different variables. Additionally, feature encoding is conducted through fully connected layers in the feedforward network. Finally, the features are projected to predict results through the projection layer. The introduction of Inverted Embeddings reverses the order of the three main modules contained within the Encoder as follows:

(1) Probsparse self-attention: The O(L2) time and memory complexity of self-attention result in significant computational costs, particularly in real-world LTSF problems. [16] introduces the Probsparse self-attention module from the Informer to address this issue.

(2) Layer normalization: In the traditional Transformer, layer normalization normalizes multiple variables simultaneously. This module was initially proposed to enhance deep neural networks' training stability and convergence. However, this operation may cause all variables within a single time step to be mixed, making them difficult to distinguish. If the collected data is not aligned in time, this operation may also introduce interaction noise between non-causal or delayed processes. In the inverted version, as in (2), layer normalization is applied to the feature representation of individual variables, allowing all variable feature channels to be under a relatively uniform distribution.

$$\text{LayerNorm}\ (\mathbf{H}) = \frac{\mathbf{h}_n - \text{Mean}\ (\mathbf{h}_n)}{\sqrt{\text{Var}\ (\mathbf{h}_n)}}\ n = 1, \cdots, \tag{2}$$

(3) Feed-forward network: In the referenced study [23], modeling temporal data using feedforward networks revealed that linear layers excel at learning the temporal characteristics inherent in time series. In response, the authors provided a plausible explanation: neurons in linear layers can learn how to extract the intrinsic properties of any time series, such as amplitude, periodicity, and even frequency spectra (the essence of Fourier transform is a fully connected mapping on the original sequence). Therefore, compared to the past approach of modeling temporal dependencies using attention mechanisms in Transformers, utilizing feedforward networks is more likely to generalize well on unseen sequences.

Unlike the intricate encoder-decoder architectures employed in preceding Transformer prediction models, IprobTrans represents an encoder-only architecture. The model comprises Embedding layers, projection layers, and N-stacked Probsparse self-attention modules. A model with only an encoder is suitable for tasks like long-time series prediction that do not require sequence generation. It focuses more on representation learning and adaptive correlation of multivariate sequences, efficiently providing feature

Authorized licensed use limited to: Shenzhen Technology University. Downloaded on June 12,2025 at 06:01:50 UTC from IEEE Xplore. Restrictions apply.

representations of input sequences. Each time series containing multiple variables is first marked using inverted Embeddings to describe the dimensions of the variables. These are then mutually represented through Probsparse self-attention and processed for long-time series prediction using feedforward neural networks. Notably, long-time series prediction in IprobTrans is essentially handed over to linear layers, which has been proven feasible in some prior related research works [24].

TABLE I. THE DATASETS USED IN THIS WORK

Dataset	Sequence lengths	Dim	Information	Frequency
ETT	174200	7	The ETT is divided into four datasets, consisting of two with a sampling frequency of 1-hour level (ETTh) and two with a sampling frequency of 15-minute level. Each dataset contains 2016.7-2018.7 power transformer load and oil temperature https://github.com/zhouhaoyi/ETDataset	15 minutes and 1 hour
Traffic	17544	862	Roadway Occupancy Recorded by Freeway Sensors in San Francisco, 2015-2016 https://github.com/thuml/Autoformer	1 hour
Electricity	321	321	Electricity consumption of 321 customers from 2011 to 2014 https://archive.ics.uci.edu/ml/datasets/ElectricityLoadDiagrams20112014	15 minutes

B. Self-supervised representation learning

To achieve excellent fine-tuning performance and advanced predictive accuracy, this chapter proposes a masked autoencoder architecture based on self-supervised learning methods for learning transferable time series representations using the Transformer model. The input time series x is first divided into non-overlapping patches. Representing the patch length as P and the patch stride, i.e., the non-overlapping region between two consecutive patches, as S, the segmentation process will generate a P×N dimensional patch sequence x_p, where N is the number of patches, . By using patches, the number of input tokens can be reduced from L to approximately L/S. This means that the self-attention graph's memory usage and computational complexity are quadratically reduced by a factor of S. Therefore, due to training time and GPU memory constraints, patch design allows the model to see longer historical sequences, which can significantly improve predictive performance.

Unlike supervised models, where patches can overlap, each input sequence is partitioned into regular, non-overlapping tiles, ensuring that observed patches do not contain information from masked patches. Then, we randomly and uniformly select a subset of patches and zero-mask them based on these selected indices. The model is trained using MSE loss to reconstruct the masked patches.

III. EXPERIMENT

A. Long-term time series forecasting

Datasets. We have conducted experiments on six widely used benchmark datasets covering a variety of real-life applications: transportation, energy, and electricity. All of these datasets are multivariate time series, as shown in Table I.

Baselines and Experimental Settings. We selected the SOTA Transformer-based models, including FEDformer [22], Autoformer [25], Informer [16], and Pyraformer [21], for comparison in multivariate forecasting. All of the models follow the same experimental setup in the original papers. Mean Square Error and Mean Absolute Error are proposed to be used as evaluation indicators. These are standard methods for assessing the difference between predicted and observed values, frequently employed for evaluating regression tasks. The formula for MSE and MAE is shown in (3).

$$\mathrm{MSE} = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2, \mathrm{MAE} = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i| \quad (3)$$

Multivariate time series forecasting requires predicting a target variable that involves multiple relevant features. Unlike univariate time series forecasting, multivariate forecasting considers various input features. We compare the model with other centralized forecasting methods, and the specific results are shown in Table II. The results show that our proposed model's average MSE and MAE reach 0.368 and 0.399 on the four ETT datasets, improving performance by 9.6% and 6.6% compared to the benchmark model FEDformer. The average forecasting performance on the four ETT datasets reaches 0.368 and 0.399, improving the performance by 9.6% and 6.6% compared to the benchmark model FEDformer, based on the Probsparse self-attention mechanism. Informer, also based on the Probsparse self-attention mechanism, has improved the prediction performance on the ETT dataset by 81.2% on average. The model's improvement is not limited to the ETT dataset but also shows excellent performance on the transportation and energy datasets, further proving its efficacy.

B. Self-supervised representation learning

In this section, experimental results are used to demonstrate the performance of the above models by selecting four state-of-the-art time series forecasting methods for comparison, including PatchTST, Autoformer, Informer, and Pyraformer, where PatchTST uses a self-supervised approach, Autoformer, Informer and Pyraformer use supervised strategies. We study the effects of Self-supervised representation learning in Table III.

The results show that IProbTrans-Fine-tuned as a self-supervised model has lower MSE and MAE than the supervised model IProbTrans for 96, 192, 336, and 720 steps of prediction on the Traffic and Electricity datasets. The MSE and MAE of the self-supervised learning are lower than those of the supervised model IProbTrans by 26.3% and 27.5%, respectively. This result shows that even without explicit labeling, self-supervised learning can effectively capture the intrinsic structure of time series data and thus achieve comparable or even better prediction performance than supervised models. The IProbTrans-Fine-tuned MSE and MAE drop by 30.5% and 24.2% on the Electricity dataset alone, further validating the potential of self-supervised

Authorized licensed use limited to: Shenzhen Technology University. Downloaded on June 12,2025 at 06:01:50 UTC from IEEE Xplore. Restrictions apply.

learning in time series forecasting. Compared with the self-supervised benchmark model PatchTST-Fine-tuned, IProbTrans-Fine-tuned achieves lower MSE and MAE for most forecast lengths.eness of traditional supervised learning in some cases.

TABLE II. MULTIVARIATE LONG-TERM FORECASTING RESULTS WITH SUPERVISED IPROBTRANS. THE BEST RESULTS ARE IN BOLD.

Models		IProbTrans		FEDformer		Autoformer		Informer		Pyraformer	
Metric		*MSE*	*MAE*	*MSE*	*MAE*	*MSE*	*MAE*	*MSE*	*MAE*	*MSE*	*MAE*
ETTh1	96	**0.363**	**0.381**	0.376	0.419	0.449	0.459	0.865	0.713	0.664	0.612
	192	**0.382**	0.455	0.420	**0.448**	0.500	0.482	1.008	0.792	0.790	0.681
	336	**0.424**	**0.459**	0.459	0.465	0.521	0.496	1.107	0.809	0.891	0.738
	720	**0.451**	**0.501**	0.506	0.507	0.514	0.512	1.181	0.865	0.963	0.782
ETTh2	96	**0.351**	**0.351**	0.346	0.388	0.358	0.397	3.755	1.525	0.645	0.597
	192	**0.387**	**0.398**	0.429	0.439	0.456	0.452	5.602	1.931	0.788	0.683
	336	**0.455**	**0.416**	0.496	0.487	0.482	0.486	4.721	1.835	0.907	0.747
	720	0.467	0.495	**0.463**	**0.474**	0.515	0.511	3.647	1.625	0.963	0.783
ETTm1	96	**0.337**	**0.373**	0.379	0.419	0.505	0.475	0.672	0.571	0.543	0.510
	192	**0.375**	**0.392**	0.426	0.441	0.553	0.496	0.795	0.669	0.557	0.537
	336	**0.388**	**0.417**	0.445	0.459	0.621	0.537	1.212	0.871	0.754	0.655
	720	**0.405**	**0.484**	0.543	0.490	0.671	0.561	1.166	0.823	0.908	0.724
ETTm2	96	**0.182**	**0.280**	0.203	0.287	0.255	0.339	0.365	0.453	0.435	0.507
	192	**0.235**	**0.301**	0.269	0.328	0.281	0.340	0.533	0.563	0.730	0.673
	336	**0.294**	**0.264**	0.325	0.366	0.339	0.372	1.363	0.887	1.201	0.845
	720	**0.398**	**0.411**	0.421	0.415	0.433	0.432	3.379	1.338	3.625	1.451
Traffic	96	**0.358**	**0.341**	0.587	0.366	0.613	0.388	0.719	0.391	2.085	0.468
	192	**0.426**	**0.358**	0.604	0.373	0.616	0.382	0.696	0.379	0.867	0.467
	336	**0.569**	0.361	0.621	0.383	0.622	**0.337**	0.777	0.420	0.869	0.469
	720	**0.571**	**0.375**	0.626	0.382	0.660	0.408	0.864	0.472	0.881	0.473
Electricity	96	**0.178**	**0.279**	0.193	0.308	0.201	0.317	0.274	0.368	0.386	0.449
	192	**0.197**	**0.288**	0.201	0.315	0.222	0.334	0.296	0.386	0.386	0.443
	336	**0.211**	**0.305**	0.216	0.329	0.231	0.338	0.300	0.394	0.378	0.443
	720	**0.240**	**0.331**	0.246	0.355	0.254	0.361	0.373	0.439	0.376	0.445

TABLE III. MULTIVARIATE LONG-TERM FORECASTING RESULTS WITH SELF-SUPERVISED IPROBTRANS. THE BEST RESULTS ARE IN **BOLD**

Models		IProbTrans Fine-tuned		PatchTST Fine-tuned		Autoformer		Informer		FEDformer	
Metric		*MSE*	*MAE*	*MSE*	*MAE*	*MSE*	*MAE*	*MSE*	*MAE*	*MSE*	*MAE*
Traffic	96	**0.334**	**0.239**	0.352	0.244	0.613	0.388	0.719	0.391	0.587	0.366
	192	**0.347**	**0.243**	0.371	0.253	0.616	0.382	0.696	0.379	0.604	0.373
	336	**0.355**	**0.251**	0.381	0.258	0.622	0.337	0.777	0.420	0.621	0.383
	720	**0.421**	**0.264**	0.425	0.280	0.660	0.408	0.864	0.472	0.626	0.382
Electricity	96	0.136	**0.194**	**0.126**	0.221	0.201	0.317	0.274	0.368	0.193	0.308
	192	**0.139**	**0.215**	0.145	0.237	0.222	0.334	0.296	0.386	0.201	0.315
	336	**0.141**	**0.237**	0.164	0.256	0.231	0.338	0.300	0.394	0.216	0.329
	720	**0.154**	**0.269**	0.193	0.291	0.254	0.361	0.373	0.439	0.246	0.355

Table III shows that our method IProbTrans-Fine-tuned has higher prediction accuracy, and the prediction accuracy grows steadily with the increase of the forecast sequence length, especially on the Electricity dataset where MSE and MAE improve 9.23% and 8.96%. This difference is likely attributed to the Iverted Embedding and probabilistic sparse self-attention mechanisms employed by IProbTrans-Fine-tuned, which may be more effective in dealing with multivariate complex time series data. Compared with supervised benchmark models such as FEDformer, Autoformer, and Informer, IProbTrans-Fine-tuned performs better in all cases. In particular, on the Traffic dataset,

Authorized licensed use limited to: Shenzhen Technology University. Downloaded on June 12,2025 at 06:01:50 UTC from IEEE Xplore. Restrictions apply.

IProbTrans-Fine-tuned's MSE and MAE are lower than those of these supervised benchmark models for 96, 192, 336, and 720-step prediction lengths, which suggests that self-supervised learning can meet or even exceed the effectiveness of traditional supervised learning in some cases.

TABLE IV. ABLATION STRATEGIES FOR IPROBTRANS

	SSL	Inverted	Prob-sparse
IProbTrans-Fine-tuned	✓	✓	✓
IProbTrans		✓	✓
IProbTrans(atten)		✓	
IProbTrans(Inverted)			✓

C. Ablation study

In the above experiments, it is not difficult to find that our proposed fine-tuning-based IProbTrans model performs well on all kinds of datasets for long sequence prediction and to further validate the reasonableness of the Inverted Embedding and the probabilistic sparse self-attention module. Detailed ablation experiments are provided in this section, including replacing the self-supervised learning strategy with the supervised one and replacing the Inverted Embedding with Position Embedding. The Probsparse self-attention module is replaced with a Quadratic complexity self-attention. The specific setup strategies for the ablation experiments are listed in Table IV. Table V shows the results of the ablation experiments of the IProbTrans model on the Traffic and Electricity datasets, including comparisons of the MSE and MAE, where '---' denotes an out-of-memory run failure. The experimental results show that the fully configured IProbTrans-Fine-tuned model exhibits optimal performance at all prediction lengths, with average MSE and MAE values of 0.253 and 0.239, respectively. When the self-supervised learning strategy is removed from the model, a significant increase in the MSE and the MAE is observed. In the ablation study, removing the Inverted Embedding further degraded the model's performance, especially when making long-term predictions. By replacing the probabilistic sparse self-attention with a self-attention of quadratic complexity, we observed that the model's calculation amount increases with the forecast's length. When the prediction length reaches 720, insufficient memory results in operational failure., whereas the IProbTrans model employs a probabilistic sparse self-attention mechanism, which reduces the amount of computation by focusing only on the most important parts of the sequence. In this way, the model can maintain high prediction accuracy while avoiding memory overflow problems that may occur in long-term prediction tasks.

IV. CONCLUSIONS

This work proposes an effective model using the Inverted Embedding and Probsparse self-attention mechanism, effectively capturing inter-variable correlation and long-distance data dependence and improving prediction efficiency. With patch segmentation and random masking techniques, the self-supervised model outperforms the supervised learning-based Transformer and the self-supervised learning PatchTST methods in all metrics. The ablation study provides valuable insights into the self-supervised learning strategy, Inverted Embedding, and Probsparse self-attention, which are crucial for the performance of the IProbTrans model in time series forecasting tasks. These findings help us understand the model's internal working mechanism and provide directions for future optimization and improvement.

TABLE V. ABLATION STUDY OF THE PROBSPARSE

Models		IProbTrans-Fine-tuned		IProbTrans		IProbTrans(atten)		IProbTrans(Inverted)	
Metric		*MSE*	*MAE*	*MSE*	*MAE*	*MSE*	*MAE*	*MSE*	*MAE*
Traffic	96	0.334	0.239	0.358	0.341	0.351	0.316	0.572	0.318
	192	0.347	0.243	0.426	0.358	0.383	0.322	0.565	0.402
	336	0.355	0.251	0.569	0.361	0.549	0.342	0.629	0.483
	720	0.421	0.264	0.571	0.375	---	---	0.711	0.576
Electricity	96	0.136	0.194	0.178	0.279	0.156	0.255	0.212	0.294
	192	0.139	0.215	0.197	0.288	0.184	0.295	0.238	0.308
	336	0.141	0.237	0.211	0.305	0.202	0.314	0.280	0.351
	720	0.154	0.269	0.240	0.331	---	---	0.384	0.425

REFERENCES

[1] Sun, L., et al., Few-shot class-incremental learning for medical time series classification. IEEE journal of biomedical and health informatics, 2023.

[2] Yuan, Y. and L. Lin, Self-Supervised Pre-Training of Transformers for Satellite Image Time Series Classification. IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, 2020. PP(99).

[3] Zerveas, G., et al., A Transformer-based Framework for Multivariate Time Series Representation Learning. 2021.

[4] Chen, Z., et al., Learning Graph Structures with Transformer for Multivariate Time Series Anomaly Detection in IoT. 2021.

[5] Meng, H., et al. Spacecraft Anomaly Detection via Transformer Reconstruction Error. in International Conference on Aerospace System Science and Engineering. 2019.

[6] Ruff, L., et al., A Unifying Review of Deep and Shallow Anomaly Detection. 2020.

[7] Li, G. and J.J. Jung, Deep learning for anomaly detection in multivariate time series: Approaches, applications, and challenges. Information Fusion, 2023. 91: p. 93-102.

[8] Shchur, O., et al., Neural Temporal Point Processes: A Review. 2021.

[9] Zhang, Q., et al. Self-Attentive Hawkes Process. in ICML '20. 2020.

[10] Zuo, S., et al. Transformer Hawkes Process. in International Conference on Machine Learning. 2020.

[11] Esling, P. and C. Agon, Time-series data mining. Acm Computing Surveys, 2012. 45(1): p. 1-34.

Authorized licensed use limited to: Shenzhen Technology University. Downloaded on June 12,2025 at 06:01:50 UTC from IEEE Xplore. Restrictions apply.

[12] Lim, B. and S. Zohren, Time Series Forecasting With Deep Learning: A Survey. 2020.

[13] Torres, J.F., et al., Deep Learning for Time Series Forecasting: A Survey. Big Data, 2020. 9(1).

[14] Wang, H., et al., DAFA-BiLSTM: Deep autoregression feature augmented bidirectional LSTM network for time series prediction. Neural Networks, 2023. 157: p. 240-256.

[15] Fan, J., et al., Parallel spatio-temporal attention-based TCN for multivariate time series prediction. Neural Computing and Applications, 2023. 35(18): p. 13109-13118.

[16] Zhou, H., et al. Informer: Beyond efficient Transformer for long sequence time-series forecasting. in Proceedings of the AAAI conference on artificial intelligence. 2021.

[17] Cirstea, R.G., et al., Triformer: Triangular, Variable-Specific Attentions for Long Sequence Multivariate Time Series Forecasting--Full Version. 2022.

[18] Nie, Y., et al., A time series is worth 64 words: Long-term forecasting with transformers. arXiv preprint arXiv:2211.14730, 2022.

[19] Wu, N., et al., Deep transformer models for time series forecasting: The influenza prevalence case. arXiv preprint arXiv:2001.08317, 2020.

[20] Li, S., et al., Enhancing the locality and breaking the memory bottleneck of Transformer on time series forecasting. Advances in neural information processing systems, 2019. 32.

[21] Liu, S., et al. Pyraformer: Low-Complexity Pyramidal Attention for Long-Range Time Ser ies Modeling and Forecasting.

[22] Zhou, T., et al., FEDformer: Frequency Enhanced Decomposed Transformer for Long-term Series Forecasting. 2022.

[23] Zeng, A., et al. Are transformers effective for time series forecasting? in Proceedings of the AAAI conference on artificial intelligence. 2023.

[24] Das, A., et al., Long-term Forecasting with TiDE: Time-series Dense Encoder. arXiv preprint arXiv:2304.08424, 2022

Authorized licensed use limited to: Shenzhen Technology University. Downloaded on June 12,2025 at 06:01:50 UTC from IEEE Xplore. Restrictions apply.