Artificial Intelligence and Computing Innovations

Intelligence Science Publishing

https://www.ispublishing.com

ISSN: 3070-4944

Vo.5, No.2, 2025

Spatiotemporal Transformer-Based Latency Prediction for Large-Scale Cloud Computing Clusters

Zijian Xiao[1], Kris Ying[2]*, Bharat Venkatesan[2]

[1]University of Pennsylvania, Philadelphia, USA

[2]Boston University, Boston, USA

**Corresponding author: krisssying@gmail.com*

Abstract: This study addresses the key problem of latency prediction in cloud clusters and proposes a spatiotemporal Transformer-based modeling method to overcome the limitations of traditional approaches in capturing complex dependencies and achieving high accuracy in large-scale dynamic environments. The research first analyzes the spatiotemporal characteristics of latency, noting that latency shows both periodic fluctuations and sudden spikes in the time dimension, while in the spatial dimension, it is influenced by the coupling of topology structures and resource competition among nodes. To address this, a unified spatiotemporal Transformer framework is constructed, where temporal self-attention captures long-range dependencies, and spatial self-attention with adjacency matrices models node interaction features, enabling a comprehensive representation of dynamic latency patterns. The method design integrates an embedding module for high-dimensional representation of multi-source inputs, residual connections, and normalization to enhance stability and generalization, and a linear mapping output layer for precise latency predictions. Experiments are conducted on the public Google Cluster Trace dataset with comparisons against mainstream models and multidimensional sensitivity analyses. Results show that the proposed method achieves superior performance in MSE, RMSE, MAE, and R^2, significantly outperforming Transformer, LSTM, TimeMixer, and ITransformer. Furthermore, the method maintains stable performance under hyperparameter sensitivity tests, environmental disturbances, and data missing scenarios, demonstrating strong robustness and adaptability in complex distributed environments. In conclusion, the proposed framework achieves high accuracy and reliability in latency prediction tasks and provides effective technical support for performance optimization and resource management in cloud computing systems.

Keywords: Cloud computing cluster; latency prediction; spatiotemporal modeling; Transformer

1. Introduction

Cloud computing clusters bring great convenience to modern information society, but latency has become a core bottleneck restricting system performance and user experience. As the scale of distributed architectures continues to expand, resource competition, network congestion, and task scheduling conflicts frequently occur among nodes in the cluster. This makes latency prediction an essential part of ensuring stable system operation[1]. Latency not only affects application response speed and interaction quality but also has long-term impacts on the reliability and scalability of higher-level services. Therefore, how to achieve accurate modeling and prediction of latency in complex multi-tenant environments and under dynamic resource allocation has become a key research direction in cloud computing[2].

Traditional approaches to latency prediction usually rely on statistical models or rule-based analysis. These methods can be effective in small-scale and low-complexity scenarios, but they often fail to maintain accuracy and stability in large-scale cloud clusters where temporal non-stationarity, spatial heterogeneity, and multi-dimensional features are intertwined. With the diversification of application demands and the explosive growth of data, latency has shown significant spatiotemporal patterns. On the one hand, latency in the time dimension presents periodic fluctuations and sudden spikes[3]. On the other hand, in the spatial dimension, latency is affected by node topology, bandwidth utilization, and load distribution. This complex coupling makes it difficult for traditional linear models to capture global and local dynamics, creating an urgent need for stronger spatiotemporal modeling methods.

In recent years, deep learning has provided new opportunities for latency prediction. Transformer structures, with their self-attention mechanism, show clear advantages in capturing long-range dependencies and modeling global relations[4]. Compared with traditional recurrent neural networks, Transformers align sequential features more flexibly and extract temporal patterns more efficiently. When combined with spatial topology modeling, they can jointly capture interactions among nodes and temporal evolution within a unified framework. Thus, spatiotemporal Transformer-based latency prediction has emerged as a frontier direction, offering precise support for resource scheduling, load balancing, and anomaly detection.

From the application perspective, latency prediction is not only important for performance optimization but also for ensuring service quality and controlling costs. In enterprise-level scenarios, even small fluctuations in latency may significantly affect critical tasks such as transactions, recommendations, and retrieval[5]. Accurate prediction provides task schedulers with prior knowledge, enabling systems to maintain low latency under high concurrency and complex dependencies. At the same time, in large-scale cluster operations, latency prediction reduces unnecessary resource expansion and supports efficient allocation of computing power and energy. This is of great value for building green and efficient cloud computing systems.

In summary, latency prediction in cloud clusters is both a theoretical challenge and a practical necessity, making it a key issue for the sustainable development of modern computing infrastructures. Spatiotemporal Transformer-based methods provide new ideas for latency modeling and open up new paths for system optimization in dynamic environments. This research not only advances the frontiers of cloud computing but also supports reliability, intelligent operations, and service quality assurance in distributed systems, carrying significant academic and practical value.

2. Related work

Existing studies on latency prediction in cloud clusters mainly focus on traditional modeling and statistical methods. These methods often rely on the mean, variance, or autoregressive models fitted from historical data. They were effective in characterizing latency distribution in early small-scale clusters, but their adaptability declined as computing scale expanded. The reason is that latency is influenced not only by internal fluctuations in time series but also by interactions among multiple factors such as resource competition, task migration, and network topology. Traditional methods cannot capture these

multidimensional, dynamic features. As a result, single statistical models often suffer from large prediction errors and poor generalization, which laid the groundwork for the introduction of machine learning and deep learning methods[6].

With the development of machine learning, supervised learning methods for latency prediction have gained attention. Researchers have attempted to use support vector machines, decision trees, and random forests for classification or regression tasks. These methods can improve accuracy when feature engineering is sufficient. However, limited by model expressiveness, they struggle to fully capture complex spatiotemporal dependencies. Under high-dimensional inputs and large-scale data, they are prone to overfitting or poor generalization. At the same time, the diversity of task types in heterogeneous environments makes static feature extraction inadequate to represent system states, which further limits the effectiveness of traditional machine learning approaches.

The introduction of deep learning has brought breakthroughs for latency prediction. Convolutional neural networks and recurrent neural networks have been widely used for temporal feature extraction and dependency modeling. By automatically learning high-dimensional feature representations, these models enhance the ability to capture complex patterns[7]. On this basis, hybrid structures have emerged, such as models combining convolution and recurrence. They can extract local patterns while maintaining long-term dependencies. However, these models face limitations when handling very long sequences and global dependencies. The sequential nature of recurrent structures reduces training and inference efficiency, while convolutional structures have limited capacity in modeling global relations. Thus, how to balance expressive power and efficiency has become a bottleneck for deep learning methods in latency prediction.

In recent years, Transformer-based spatiotemporal modeling methods have become a research focus. Through self-attention mechanisms, the model can capture long-term temporal dependencies and spatial topology within a unified framework. This overcomes the limitations of traditional methods in global modeling and scalability. In cloud cluster latency prediction, the introduction of Transformers enables effective integration of multi-source data and dynamic modeling of interactions among nodes, while maintaining strong robustness and generalization. In addition, continuous improvements have been proposed, including lightweight structures, optimized positional encoding, and combinations with graph neural networks or temporal decomposition mechanisms. These advances further drive the intelligent and efficient development of latency prediction methods and provide new solutions for performance optimization in large-scale distributed computing environments.

Recent studies on cloud forecasting, elastic control, distributed security, and operational representation learning further extend the methodological basis for spatiotemporal latency prediction. Generalizable latent representation learning for multi-source time series forecasting in cloud systems suggests that heterogeneous monitoring streams can be mapped into transferable hidden states for robust prediction across changing workloads[8]. Risk-constrained reinforcement learning for SLO-driven cloud-native elastic scaling links latency-aware prediction with service-level control and resource adjustment under operational constraints[9]. Secure parameter aggregation and distributed anomaly detection in cross-cloud data center networks highlight the need to preserve privacy and reliability when prediction models operate across multiple administrative domains[10]. Transformer-based semantic log modeling strengthens explainable cloud security analytics by extracting higher-level operational semantics from log sequences[11], while multi-scale Transformer forecasting for CPU utilization shows how hierarchical temporal patterns can improve cloud resource prediction[12]. Graph neural modeling of traffic propagation and topological interactions provides a structural basis for representing latency diffusion among cluster nodes[13]. Reinforcement learning-driven hotspot-aware cache management also offers an operational view of adaptive resource control under workload shifts[14]. Together, these studies support the proposed spatiotemporal Transformer framework by connecting multi-source forecasting, topology-aware dependency modeling, semantic operational evidence, and deployment-oriented control.

3. Proposed Approach

This study proposes a method for cloud computing cluster latency prediction based on a spatiotemporal Transformer, aiming to simultaneously model the temporal dependence and spatial correlation of latency. First, the raw latency series and multi-dimensional inputs such as resource utilization and network traffic are mapped into a unified high-dimensional representation space to capture the potential correlations between features. The overall model architecture is shown in Figure 1.

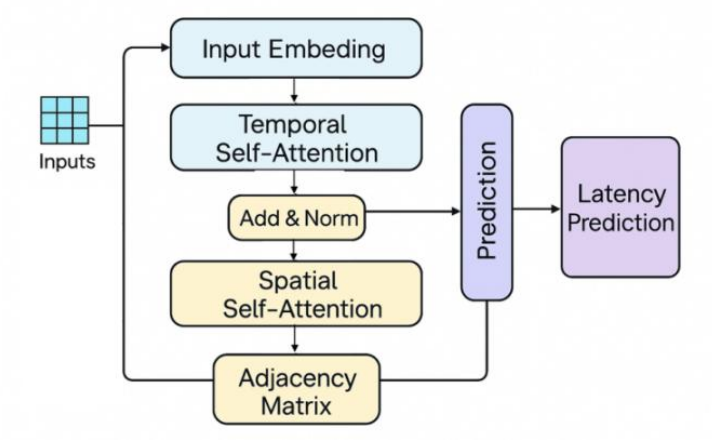


Figure 1. Overall model architecture

Assuming the input sequence is $X \in R^{T \times N \times d}$, where T represents the time step, N represents the number of nodes, and d represents the feature dimension, the input representation is obtained by embedding the mapping function:

$$H_0 = f_{embed}(X) \in R^{T \times N \times d_h} \tag{1}$$

Where d_h is the hidden layer dimension, and f_{embed} is the combination of linear transformation and positional encoding. Through this mapping, the raw data is converted into a structured feature tensor suitable for processing by the spatiotemporal Transformer.

In the temporal modeling part, the self-attention mechanism is used to capture the global dependencies between different time steps. Given an input representation H_t, the self-attention in the temporal dimension is implemented by computing the query, key, and value matrices:

$$Attention(Q, K, V) = Softmax\left(\frac{QK^T}{\sqrt{d_k}}\right)V \tag{2}$$

Where $Q = H_t W_Q, K = H_t W_K, V = H_t W_V$ and $W_Q, W_K, W_V \in R^{d_h \times d_k}$ are learnable parameters. This mechanism can dynamically adjust the weights of different time slices, allowing the model to highlight the delay patterns at critical moments.

In the spatial modeling part, the coupling effect of the topological relationship between nodes and resource competition is considered. By constructing the adjacency matrix $A \in R^{N \times N}$ to characterize the connection strength between nodes, and combining it with the self-attention mechanism to achieve spatial dependency modeling:

$$H_s = \sigma(AH_t W_s) \tag{3}$$

Where $W_s \in R^{d_h \times d_h}$ is a trainable weight and $\sigma(\cdot)$ is a nonlinear activation function. This spatial modeling module can identify potential interactions between nodes on a global scale, effectively capturing the propagation patterns of cluster latency at different locations.

To achieve unified modeling of spatiotemporal features, this study integrates temporal and spatial attention mechanisms and improves model stability through residual connections and normalization operations. The fused representation can be formalized as:

$$H = LayerNorm(H_t + H_s) \tag{4}$$

Finally, the global feature representation is input into the prediction layer, and the delay prediction value is generated using linear mapping:

$$\hat{y}_{t+1} = W_0 H + b_0 \tag{5}$$

Where $W_0 \in R^{d_h \times 1}, b_0 \in R$ is the output layer parameter. This prediction framework integrates temporal patterns and spatial structure information under a unified structure, providing a more comprehensive and accurate modeling approach for latency prediction of cloud computing clusters.

4. Performance Evaluation

4.1 Dataset

The dataset used in this study comes from the publicly available Google Cluster Trace. It collects operational logs from large-scale cloud clusters, covering thousands of servers over a long period. The dataset includes information on task scheduling, resource utilization, and system status. Its dimensions consist of core indicators such as CPU, memory, disk I/O, and network bandwidth. It also records the entire process of job submission, task execution, and completion, providing high-quality raw data for analyzing cluster characteristics and latency patterns.

This dataset features a long time span and large scale, which accurately reflects the dynamics of multi-tenant sharing and resource competition in cloud environments. Because the cluster involves many nodes and complex task types, fluctuations and anomalies in latency occur frequently. The dataset contains both regular periodic patterns and sudden latency spikes. These characteristics make it an important resource for studying spatiotemporal dependencies and system robustness, offering sufficient temporal and spatial information for building latency prediction models.

In addition, the dataset is highly representative of real-world applications, effectively simulating the operation of actual data centers. Modeling and analyzing this dataset can reveal the mechanisms behind latency formation in cloud clusters and provide practical insights for resource scheduling and system optimization. Its openness and reusability also establish a unified benchmark for different studies, enabling more objective and fair comparisons of methods. This contributes to the continuous advancement of performance prediction and optimization in cloud computing.

4.2 Experimental Results

This paper first conducts a comparative experiment, and the experimental results are shown in Table 1.

From the overall comparison, the performance differences among methods in cloud cluster latency prediction are clear. Traditional LSTM shows some advantages in capturing temporal dependencies. However, it fails to model long-range dependencies and complex spatial relations effectively. As a result, it performs worse on MSE, RMSE, and MAE, and also shows a relatively low R^2. Transformer improves

global dependency modeling and outperforms LSTM, but its lack of explicit spatial topology modeling limits prediction accuracy.

Table 1. Comparative experimental results

Method	MSE	RMSE	MAE	R^2
Transformer[15]	0.165	0.406	0.298	0.842
LSTM[16]	0.182	0.427	0.315	0.826
TimeMixer[17]	0.149	0.386	0.284	0.861
ITransformer[18]	0.141	0.375	0.276	0.869
Ours	0.118	0.344	0.251	0.892

A closer look at improved models shows that both TimeMixer and ITransformer achieve higher accuracy. This indicates that enhanced temporal modeling mechanisms are effective in capturing complex latency patterns. TimeMixer reduces noise interference through multi-scale temporal decomposition. ITransformer is more robust in global attention allocation. Therefore, both methods outperform the first two in RMSE and MAE, demonstrating stronger generalization and stability.

The method proposed in this study achieves the best results across all evaluation metrics. Both MSE and MAE are significantly lower than other methods. R^2 reaches 0.892, showing stronger capability in modeling latency fluctuations and spatial correlations. Compared with ITransformer, this method further reduces prediction error. This proves that joint spatiotemporal modeling balances local details and global dependencies better, leading to higher accuracy and stability in complex cloud environments.

These results show that in dynamic clusters with frequent multi-tenant sharing and resource competition, relying only on temporal or spatial modeling cannot achieve ideal performance. By integrating spatiotemporal Transformer design, the model captures global dependencies while also considering spatial interactions among nodes. This improves robustness in latency prediction. It not only verifies the rationality of the theoretical design but also highlights the potential for practical application in real cloud computing scenarios.

This paper further presents an experiment on the sensitivity of the number of attention heads to the experimental results, and the experimental results are shown in Figure 2.

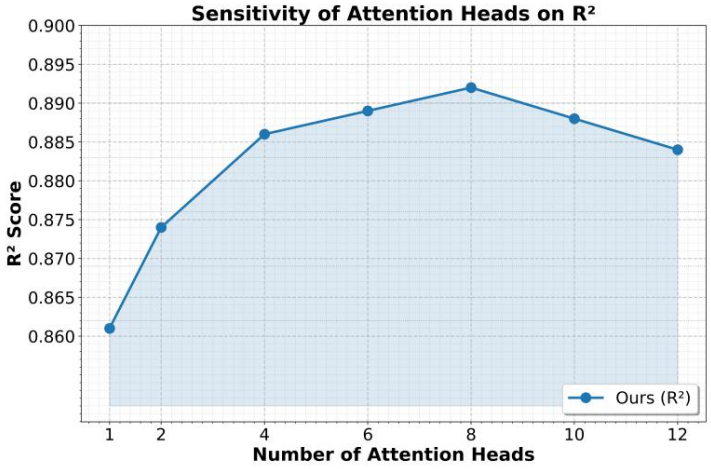


Figure 2. Sensitivity experiment of the number of attention heads to experimental results

From the overall trend, the number of attention heads has a clear impact on performance in latency prediction. As the number of heads increases from 1 to 8, the R^2 value rises steadily. This shows that more attention heads enhance the model's ability to capture complex spatiotemporal dependencies, which

improves prediction accuracy. In cloud clusters, this reflects stronger pattern recognition under conditions of intertwined multi-dimensional features and multi-node interactions.

At a low number of heads, R^2 is noticeably lower, indicating limited representational capacity of the model. In this case, the attention mechanism cannot simultaneously cover multiple dependencies across time slices and spatial nodes. As a result, the fitting degree of predictions is insufficient. This phenomenon matches the complexity of large-scale clusters, where insufficient model capacity leads to missing local features or long-range dependencies, causing higher prediction errors.

When the number of heads increases to 8, R^2 reaches its peak. This suggests that the model achieves a good balance between complexity and generalization in this setting. The model can effectively decompose spatiotemporal features and establish a proper distribution of attention between global and local aspects, which maximizes prediction ability. This result shows that moderately increasing model complexity has a positive effect on capturing dynamic latency patterns in cloud clusters.

When the number of heads further increases to 10 or 12, R^2 drops slightly. This indicates that too many heads introduce redundant information and additional computational costs, which may weaken generalization. It shows that blindly increasing model complexity does not guarantee continuous improvement. Proper structural configuration is more effective in balancing efficiency and accuracy, providing practical solutions for latency prediction in large-scale distributed systems.

This paper also presents a sensitivity experiment on the missing rate to the experimental results, and the experimental results are shown in Figure 2.

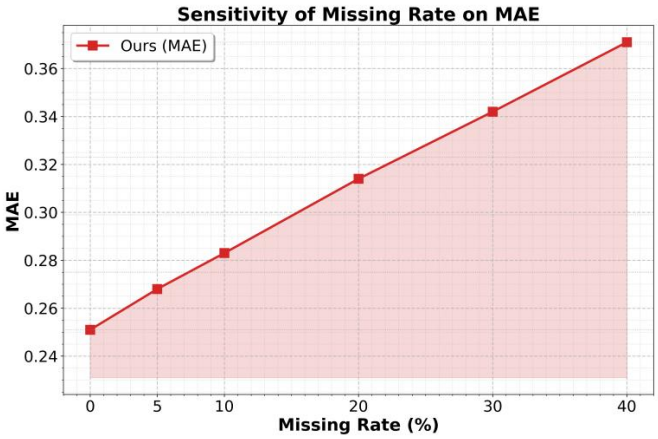


Figure 2. Sensitivity test of the missing rate to experimental results

From the overall trend, it can be observed that an increase in the missing rate has a significant impact on latency prediction performance. As the missing ratio rises from 0% to 40%, MAE shows a steady upward trend. This indicates that data integrity plays a crucial role in spatiotemporal modeling. When the missing rate is low, the model can still rely on global dependencies and redundant features to achieve high accuracy. As the missing rate increases, information loss accumulates, leading to larger prediction errors.

At low missing rates, the model error remains within a reasonable range. This shows that the attention mechanism of the spatiotemporal Transformer has some noise resistance. By weighting multiple time slices and nodes, it can partially alleviate the effect of missing information and maintain relatively good prediction performance. This property is important in real cloud clusters, where mild data loss is common, ensuring that the model remains usable under incomplete data conditions.

However, when the missing rate exceeds a certain threshold, the error grows faster. This suggests that model robustness has a limit. With the loss of key information, the attention mechanism cannot build stable dependencies at the global level. As a result, the model's representation of latency patterns

becomes distorted. This effect is more pronounced in large-scale clusters, where the loss of node correlations directly reduces prediction accuracy.

The results further show that the missing rate is a key factor affecting prediction accuracy. In real deployment, if there is no effective data cleaning or missing value handling, even optimized spatiotemporal Transformers cannot maintain stable performance. Ensuring data quality, reducing the missing ratio, or designing stronger imputation strategies will be critical paths to improving the reliability of latency prediction in cloud clusters.

5. Conclusion

This study focuses on the problem of latency prediction in cloud clusters and proposes a spatiotemporal Transformer-based modeling method. The design combines global dependency capture in the time dimension with topology modeling in the spatial dimension. This allows the model to effectively characterize dynamic latency features within a unified framework. Compared with existing methods, the proposed approach achieves higher accuracy and robustness in performance metrics. This demonstrates the feasibility and superiority of joint spatiotemporal modeling in complex distributed environments. The results enrich research directions in latency prediction and provide new insights for model design in related fields.

From the perspective of system performance optimization, the proposed model offers more refined predictive support for resource scheduling and task allocation. In multi-tenant and resource-competitive scenarios, predicting latency trends in advance helps schedulers allocate computing resources more reasonably. This reduces response time and improves overall service quality. The prediction results also help system administrators identify potential bottleneck nodes and mitigate performance fluctuations caused by network congestion or resource shortages. Such foresight is of great value for building stable and efficient cloud computing environments.

In terms of engineering applications, the proposed latency prediction method improves resource utilization while ensuring computational efficiency. By accurately predicting latency, it avoids unnecessary redundancy and excessive scaling. This reduces operating costs and supports green and energy-efficient goals without sacrificing performance. For real-time response scenarios such as online transactions, intelligent recommendations, and large-scale data analysis, the framework enhances system reliability and stability. It further improves user experience and ensures service continuity.

Moreover, the methodological generality of this study provides references for other performance prediction tasks in distributed systems. Whether for computing latency, network throughput, or resource utilization, spatiotemporal modeling frameworks have potential transfer value. This shows that the study is not only influential in cloud computing but also offers theoretical and practical guidance for intelligent operations and distributed computing optimization in broader contexts.

References

[1] Jiang J, Chen L, Ke L, et al. A review of transformer models in drug discovery and beyond[J]. Journal of Pharmaceutical Analysis, 2025, 15(6): 101081.

[2] Wu Y, Yang H, Lin Y, et al. Spatiotemporal propagation learning for network-wide flight delay prediction[J]. IEEE Transactions on Knowledge and Data Engineering, 2023, 36(1): 386-400.

[3] Arbat S, Jayakumar V K, Lee J, et al. Wasserstein adversarial transformer for cloud workload prediction[C]//Proceedings of the AAAI Conference on Artificial Intelligence. 2022, 36(11): 12433-12439.

[4] Mouen A S F M, Zeutouo J L, Tchendji V K. Transformer-Based Model for Cold Start Mitigation in FaaS Architecture[J]. arXiv preprint arXiv:2504.11338, 2025.

[5] Li A M, Li J. A hybrid spatiotemporal LSTM and transformer network for cellular traffic prediction[J]. Computer Networks, 2025: 111669.

[6] Wei S, Yang Y, Liu D, et al. Transformer-Based spatiotemporal graph diffusion convolution network for traffic flow forecasting[J]. Electronics, 2024, 13(16): 3151.

[7] Kumar R, Mendes-Moreira J, Chandra J. Spatio-temporal parallel transformer based model for traffic prediction[J]. ACM Transactions on Knowledge Discovery from Data, 2024, 18(9): 1-25.

[8] Huang W. Generalizable Latent Representation Learning for Multi-Source Time Series Forecasting in Cloud Systems[J].2024, 3(1).

[9] Hu P, Uziel G, Ma X. Risk-Constrained Reinforcement Learning for SLO-Driven Cloud-Native Elastic Scaling[J]. 2024, 3(1).

[10] Zhang S. Secure Parameter Aggregation and Distributed Anomaly Detection in Cross-Cloud Data Center Networks[J].2024, 3(4).

[11] Yang Z. Transformer-Based Semantic Log Modeling for Robust and Explainable Cloud Security Analytics[J]. 2024, 4(5).

[12] Wang Z, Lin S. Multi-Scale Transformer for CPU Utilization Time Series Forecasting in Cloud Computing[J]. 2025, 4(7).

[13] Huang X. Capturing Traffic Propagation and Topological Interactions in Cloud Networks via Graph Neural Networks[J]. 2024, 4(5).

[14] Dang S, Chen C, Chua K S. Reinforcement Learning-Driven Dynamic Cache Management with Hotspot-Aware Replacement[J]. 2024, 4(4).

[15] Chu Z, Li B, Ying C. Transformer-based latency prediction for stream processing task[J]. Journal of King Saud University Computer and Information Sciences, 2025, 37(5): 106.

[16] Nguyen T, Do T, Le K, et al. An LSTM-based Approach for predicting resource utilization in cloud computing[C]//Proceedings of the 11th International Symposium on Information and Communication Technology. 2022: 173-179.

[17] Wang S, Wu H, Shi X, et al. Timemixer: Decomposable multiscale mixing for time series forecasting[J]. arXiv preprint arXiv:2405.14616, 2024.

[18] Liu Y, Hu T, Zhang H, et al. itransformer: Inverted transformers are effective for time series forecasting[J]. arXiv preprint arXiv:2310.06625, 2023.