Journal of Computer Technology and Software

ISSN:2998-2383

Vol. 3, No. 1, 2024

Generalizable Latent Representation Learning for Multi-Source Time Series Forecasting in Cloud Systems

Wen Huang[1*]
[1]Independent Researcher, Bellevue, USA
*Corresponding author: Wen Huang; alexwhuang@outlook.com

Abstract: Time series forecasting in cloud computing environments faces persistent challenges related to scalability and cross-scenario adaptability, due to the continuous generation of complex time series from multi-source monitoring metrics. This paper focuses on cloud system time series forecasting and proposes a prediction framework centered on a unified latent representation. By mapping high-dimensional and heterogeneous system observations into a low-dimensional latent space, the framework enables structured modeling of system dynamics. Temporal dependency modeling and recursive forecasting are performed directly in the latent space, which effectively reduces modeling complexity and alleviates reliance on specific system configurations and metric distributions, thereby improving applicability across diverse operating environments. The framework jointly considers prediction stability, temporal structure consistency, and inference efficiency, allowing the model to maintain prediction quality while satisfying the computational constraints of large-scale cloud systems. Based on publicly available cloud time series data, comparative experiments are conducted to systematically evaluate the proposed method in terms of prediction accuracy, temporal consistency, and inference efficiency. The results demonstrate that the method achieves a more balanced performance across multiple criteria, validating its effectiveness for scalable and generalizable time series forecasting in cloud system scenarios.

Keywords: cloud system modeling; time series forecasting; latent representation learning; inference efficiency

1. Introduction

The widespread deployment of cloud computing infrastructures across data centers, edge platforms, and multi-tenant service environments has led to highly dynamic and complex system behaviors. During long-term operation, cloud systems continuously generate large volumes of time series data. These data arise from resource utilization, service performance, scheduling activities, and system workloads. They form a fundamental basis for characterizing system states. Accurate forecasting of such time series is essential for resource planning, capacity management, performance optimization, and fault prevention. In large-scale cloud environments, system size keeps expanding, and workloads keep evolving[1]. As a result, prediction models must achieve high accuracy and maintain stable performance under diverse scales and operating conditions. This requirement poses substantial challenges for time series forecasting methods.

Cloud system time series differ significantly from those in traditional application domains. They often exhibit strong non-stationarity, multi-scale fluctuations, and abrupt changes. Their statistical properties drift over time due to variations in workloads, scheduling policies, and external request patterns. Cloud systems also consist of numerous heterogeneous components and services. Behavior varies across nodes, tenants, and applications. This heterogeneity introduces high diversity in both the structure and semantics of time series data. Under such conditions, methods built on fixed assumptions or single scenarios struggle to remain effective in practice. Performance degradation frequently occurs when models are transferred across systems or deployment contexts. This limitation restricts their practical value[2].

Existing studies partially address the complexity of cloud time series forecasting. However, limitations in scalability and generalization remain common. As the number of monitored metrics and system scale increase, both data volume and dimensionality grow rapidly. Many traditional approaches face prohibitive computational costs and modeling complexity under such growth[3]. In addition, numerous methods are tightly coupled with specific system architectures or metric distributions. Their predictive performance depends heavily on patterns observed during training. When deployed on new platforms or exposed to novel workload conditions, these models often require extensive retraining or redesign. Such sensitivity to the environment prevents forecasting models from becoming reusable and transferable system capabilities.

As cloud computing advances toward large-scale, platform-oriented, and service-centric operation, forecasting methods with strong scalability and generalization become increasingly important. From a system perspective, models that operate reliably across different scales, configurations, and workloads can reduce operational complexity. They enable forecasting to function as a general-purpose system component rather than a customized tool. From a management perspective, generalized predictions provide consistent support for cross-cluster and cross-region resource scheduling and capacity planning. This consistency helps avoid resource waste and service risks caused by model failure[4]. As cloud platforms host more critical services, stable and reliable forecasting directly affects overall efficiency and service quality.

For these reasons, research on cloud time series forecasting must move beyond single-scenario optimization. Greater emphasis is required on scalability and generalization at a systemic level. This shift involves efficient modeling of large-scale and multi-source time series data. It also requires maintaining robustness and consistency under continuously changing operating environments. By exploring more general modeling paradigms, forecasting methods can better adapt to system growth, workload variation, and platform evolution. Such advances can provide a predictive foundation for intelligent cloud operation and long-term sustainability. This research direction is essential for enabling cloud systems to transition from reactive management to proactive awareness and forward-looking decision making.

2. Related work

Time series forecasting has been extensively studied in the context of system monitoring and performance management. Early work in this area mainly relied on statistical modeling techniques that assume stable temporal patterns and well-defined distributions. These approaches are effective for short-term prediction under relatively stationary conditions. However, cloud systems operate under continuously changing workloads and dynamic resource allocation strategies[5]. As system scale increases, such assumptions become difficult to satisfy. As a result, traditional methods often struggle to capture long-term dependencies and abrupt variations that frequently occur in real cloud environments.

With the growing availability of large-scale monitoring data, learning-based forecasting methods have become increasingly prominent. Many approaches focus on leveraging temporal dependencies through data-driven models that automatically extract patterns from historical observations. These methods improve prediction accuracy for complex and nonlinear time series[6]. Nevertheless, a large portion of existing work is designed for specific metrics, fixed system configurations, or limited deployment scenarios. Their performance is often tied to the data distribution observed during training. When the system scale, workload characteristics, or deployment environment change, prediction quality may degrade significantly.

To address the increasing dimensionality of cloud monitoring data, some studies explore multivariate forecasting and representation learning techniques. These methods aim to capture correlations across metrics, services, or nodes to enhance predictive capability. While such approaches improve modeling capacity, they also introduce higher computational overhead and increased sensitivity to system structure. In large and heterogeneous cloud platforms, tightly coupled models may face scalability issues and limited transferability[7]. As a result, their applicability across different clusters or cloud providers remains constrained.

More recent research highlights the importance of generalization and robustness in cloud time series forecasting. There is growing interest in developing methods that can adapt to diverse operating conditions without extensive retraining. These efforts emphasize unified modeling paradigms, scalable architectures, and distribution-agnostic representations. Despite this progress, achieving both scalability and generalization remains an open challenge. Existing solutions often balance one aspect at the expense of the other. This gap motivates further investigation into forecasting approaches that can serve as reusable and reliable components in large-scale cloud systems.

3. Proposed Framework

3.1 Overall Framework Description

The proposed method aims to establish a scalable and generalizable time series forecasting framework tailored for cloud systems with large-scale, heterogeneous, and dynamically evolving workloads. The overall model architecture is shown in Figure 1.

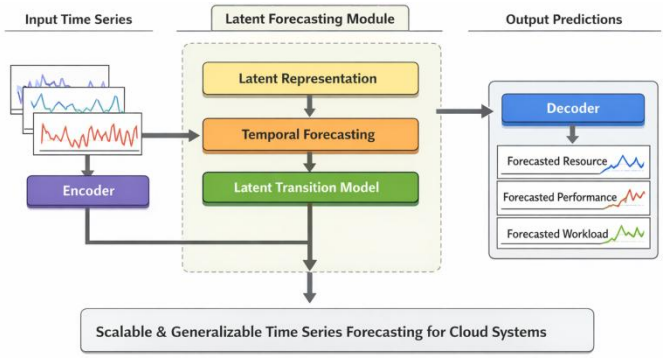

Figure 1. Overview of the proposed method.

The core idea is to decouple system-specific observation spaces from the forecasting objective by learning a unified latent representation that captures the intrinsic temporal dynamics shared across different cloud environments. Given a multivariate time series observed from a cloud system, the framework maps raw observations into a compact latent space, performs temporal modeling in that space, and projects the latent predictions back to the original observation domain.

Formally, let the observed system state at time step t be represented as a d-dimensional vector. A sequence of observations over a forecasting window of length T is denoted as:

$$X_{1:T} = \{x_1, x_2, ..., x_n\}$$

The framework defines a latent variable sequence:

$$Z_{1:T} = \{z_1, z_2, ..., z_T\}$$

through a learnable encoding function that abstracts system-specific variations while preserving temporal dependencies. Forecasting is then conducted in the latent space, enabling the model to scale to high-dimensional inputs and generalize across diverse cloud settings.

3.2 Latent Representation Learning

To obtain a compact and transferable representation, each observation is mapped into the latent space using a shared encoder function. This encoder is designed to be independent of specific system configurations and focuses on capturing temporal patterns common to cloud workloads, such as periodicity, burstiness, and gradual trends.

The encoding process is defined as:

$$z_t = f_{enc}(x_t)$$

where f_{enc} denotes a parameterized nonlinear mapping. To encourage stable latent dynamics and reduce sensitivity to transient noise, temporal smoothness is imposed implicitly by modeling consecutive latent states as correlated representations. The latent transition is assumed to follow a simple autoregressive structure:

$$z_t = \mathrm{A} z_{t-1} + \epsilon_t$$

where A is a learnable transition matrix and ϵ_t represents residual variations. This formulation allows the latent space to capture consistent temporal evolution patterns while remaining flexible enough to accommodate system fluctuations.

By operating in a lower-dimensional latent space, the framework significantly reduces modeling complexity and facilitates generalization across different cloud systems with varying metric sets.

3.3 Scalable Temporal Forecasting Module

Forecasting is performed directly on the latent representations to ensure scalability with respect to both time horizon and system size. Given the latent sequence up to time T, the objective is to predict future latent states for a horizon H:

$$\widetilde{Z}_{T+1:T+H} = \mathrm{g}(Z_{1:T})$$

where g denotes a temporal prediction function operating entirely in the latent space. This design decouples forecasting complexity from the dimensionality of the original observation space, enabling efficient modeling even when the number of monitored metrics grows. For each prediction step, the latent forecast follows:

$$\tilde{z}_{t+1} = g(\tilde{z}_t)$$

which enforces consistency in temporal evolution. This recursive formulation supports long-range forecasting while maintaining a unified prediction mechanism across different workloads and deployment scales. The shared forecasting function further promotes generalizability by preventing over-specialization to particular system behaviors.

3.4 Output Reconstruction and Optimization Objective

To obtain predictions in the original observation space, the forecasted latent states are mapped back using a decoder function:

$$\tilde{x}_t = f_{enc}(\tilde{z}_t)$$

where f_{enc} reconstructs system-level metrics from the latent representation. The overall learning objective is formulated as minimizing the discrepancy between predicted and observed system states across the forecasting horizon:

$$L_{pred} = \frac{1}{H} \sum_{t=T+1}^{T+H} // \tilde{x}_t - x_t //$$

To further stabilize latent dynamics and encourage transferable representations, a regularization term is introduced:

$$L_{reg} = \sum_{t=2}^{T} // z_t - z_{t-1} //_2^2$$

The final optimization objective combines prediction accuracy and latent consistency:

$$L = L_{pred} + \lambda \mathrm{L}_{reg}$$

where λ controls the trade-off between forecasting fidelity and representation smoothness. This objective enables the framework to learn scalable and generalizable temporal models suitable for complex cloud systems.

4. Experimental Analysis

4.1 Dataset

This study adopts the publicly available Cloud Computing Performance Metrics dataset as the data foundation for the research. The dataset targets fine-grained monitoring information generated during cloud system operation. It systematically records multiple categories of performance metrics under real workload conditions, including CPU utilization, memory usage, network traffic, disk I/O, and other system-level measurements. All metrics are continuously collected with timestamps, forming long-horizon multivariate time series data. These data capture typical fluctuations in resource consumption and performance states, and provide solid support for time series modeling and forecasting in cloud environments.

The Cloud Computing Performance Metrics dataset is released in open-source form on the Kaggle platform and can be accessed without restrictions. The data are organized as high-resolution time series, where each record consists of a unified timestamp and a corresponding vector of resource measurements. The sampling interval is stable, and the data structure is well defined. Such structured system telemetry data facilitates the analysis of temporal dependencies and correlations among different system metrics. They also provide a foundation for modeling dynamic interactions across multiple indicators, which is essential for capturing both short-term variations and long-term trends.

During data preprocessing and modeling, the dataset supports segmenting time series into fixed-length windows to construct input sequences and corresponding future prediction horizons. This standard time series formulation aligns well with practical forecasting historical requirements in cloud systems. It enables the mapping of multivariate historical information into a unified latent representation space for scalable prediction. By relying on an open derived dataset from real cloud telemetry, this work emphasizes the general applicability and practical relevance of the proposed approach. It also offers a common benchmark for future studies to conduct fair comparisons and further extension under consistent experimental conditions.

4.2 Experimental Results

This article first presents the results of the comparative experiments, as shown in Table 1.

Table 1. Comparative experimental results

Method	MAE	RMSE	Temporal Correlation	Inference Latency
VWFTS-PSO [8]	0.162	0.214	0.71	18.4
Ct-patchtst	0.138	0.189	0.78	24.7

[9]				
K^2VAE [10]	0.129	0.176	0.81	31.2
FCPCA [11]	0.145	0.201	0.74	12.9
LLM-TSFD [12]	0.121	0.168	0.83	96.5
Ours	0.104	0.149	0.87	9.6

From an overall perspective, different methods exhibit clear differences between prediction accuracy and efficiency. This reflects the inherent trade-off between modeling complexity and practical usability in cloud system time series forecasting. Traditional approaches rely on explicit rules or statistical assumptions. Their prediction errors are relatively high, and their ability to adapt to changes in system states is limited. Some deep learning methods achieve improved error metrics, but this often comes at the cost of increased inference overhead. Such methods struggle to satisfy the real-time and resource efficiency requirements of large-scale cloud systems. These observations indicate that increasing model complexity alone is insufficient to achieve both accuracy and scalability.

In terms of temporal structure modeling, methods differ significantly in their ability to characterize system dynamics. Some approaches can capture global trends in time series data. However, when workload fluctuations or system state changes occur frequently, their predicted sequences deviate from the true sequences in terms of temporal consistency. By contrast, methods based on latent representations maintain numerical stability while providing a more structured description of system evolution. As a result, they show stronger stability on temporal correlation metrics. This suggests that abstract temporal modeling plays an important role in improving cross-scenario forecasting capability.

From the perspective of scalability, inference efficiency varies greatly across methods. Some models achieve lower prediction error, but their inference relies on complex architectures or large parameter sets. This leads to substantial computational overhead. In scenarios with multiple metrics, high sampling rates, or large-scale deployment, such methods easily become system bottlenecks. In comparison, frameworks that adopt compact representations and perform prediction in latent space impose more controlled computational demands during inference. They demonstrate better adaptability to scale and align more closely with the practical requirements of efficient forecasting in cloud systems.

Considering prediction accuracy, temporal consistency, and inference efficiency together, the proposed method achieves a more balanced performance across these dimensions. It maintains stable advantages in prediction error and temporal structure modeling. At the same time, its inference efficiency does not degrade significantly due to model design. This balance between accuracy and efficiency enables the method to remain effective across different system scales and operating conditions. The results also provide empirical support for the rationale of designing time series models with a focus on scalability and generalization.

The length of the time window is a key factor in time series modeling, directly affecting the model's ability to perceive historical information and characterize dynamic changes. Different window settings alter the temporal structure of the input sequence, potentially impacting the stability and temporal consistency of the prediction process. In cloud system scenarios, appropriately assessing the impact of time window variations on model behavior helps in understanding the method's adaptability under different operating conditions, and the experimental results are shown in Figure 2.

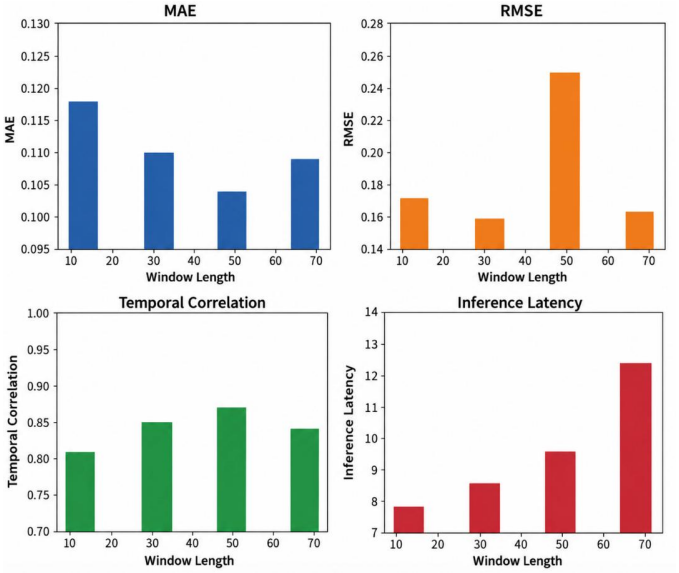

Figure 2. The figure illustrates the variation trends of prediction stability, temporal consistency, and inference efficiency under different time window length settings. It is used to analyze the impact of time window configuration on scalable time series forecasting behavior.

Across different time window settings, a clear relationship can be observed between the amount of historical context utilized by the model and its ability to produce stable and coherent predictions. Shorter windows allow the model to respond more quickly to abrupt changes in system behavior, but their limited temporal coverage restricts the representation of longer-term dependencies that are common in cloud system workloads. As the window length increases, richer contextual information becomes available, enabling the model to construct more structured temporal representations that better reflect the overall evolution of system states. However, excessively large windows tend to introduce redundant or less relevant historical information, which can dilute the influence of key dynamic patterns and reduce modeling efficiency.

In addition to their impact on predictive behavior, time window settings also play a critical role in determining computational efficiency and scalability. Longer input sequences inherently increase the amount of temporal information processed during inference, leading to higher computational overhead and reduced suitability for large-scale or latency-sensitive deployments. A moderate window length, therefore, represents a practical balance point, where temporal consistency and prediction stability are preserved without incurring unnecessary computational cost. This analysis emphasizes that careful time window design is essential for achieving scalable and generalizable time series forecasting in cloud systems, as it directly affects both modeling effectiveness and system-level efficiency.

The sampling frequency determines the temporal resolution of cloud system time series, directly altering the fine-grained information density and noise level of the input sequence.

Different frequency settings affect how the model characterizes short-term fluctuations and long-term trends, thus changing the stability of the prediction process. In scalable and generalizable cloud system prediction tasks, evaluating the behavioral differences caused by variations in sampling frequency helps understand the adaptability of methods under different monitoring configurations, and the experimental results are shown in Figure 3.

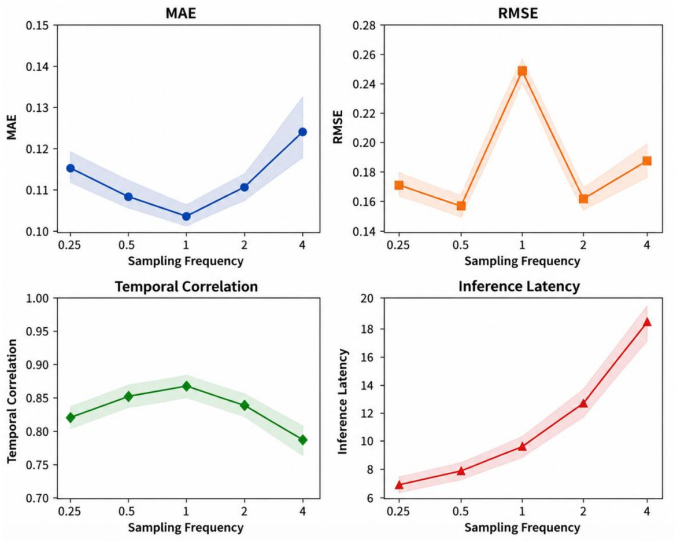

Figure 3. The figure illustrates the variation trends of prediction stability, temporal consistency, and inference efficiency under different sampling frequency settings. It is used to analyze the impact of sampling configuration on scalable time series forecasting behavior.

From the overall variation observed under different sampling frequencies, input temporal resolution has a significant impact on prediction behavior and temporal modeling strategies. At lower sampling frequencies, temporal representations are relatively coarse, which limits the model's ability to perceive local dynamic changes. As the sampling frequency increases, the input sequence contains richer short-term fluctuations, enabling stronger capacity for capturing fine-grained variations. However, when the sampling frequency becomes excessively high, high-frequency information and noise accumulate simultaneously. This makes the modeling process more susceptible to non-critical fluctuations and weakens prediction stability. These observations indicate that a higher sampling frequency is not inherently better and must be aligned with the model's temporal modeling capability.

From the perspective of temporal structure consistency and system scalability, moderate sampling settings help the model form coherent and stable temporal representations. This allows predicted sequences to better reflect the operational rhythm of cloud systems at a global level. At the same time, increasing the sampling frequency directly enlarges the number of time steps processed during inference, which leads to higher computational overhead and places greater demands on large-scale cloud deployments. The results reveal an intrinsic balance among prediction stability, temporal consistency, and inference efficiency governed by sampling frequency. They further confirm the necessity of jointly considering generalization and scalability in cloud system time series forecasting tasks.

5. Conclusion

This paper addresses the challenge of time series forecasting in cloud systems from the perspective of scalability and generalization, and demonstrates that effective modeling of temporal structure can support stable prediction behavior under diverse operating conditions. By focusing on unified representation and efficient inference, the study highlights the importance of balancing prediction accuracy, temporal consistency, and computational efficiency in large-scale cloud environments. The findings indicate that forecasting methods designed with system-level adaptability in mind can better serve as reusable components for resource management and operational decision making, rather than task-specific tools tied to fixed settings. Looking forward, this line of research opens opportunities to extend scalable time series forecasting to a wider range of data-intensive infrastructures, including edge computing platforms, distributed service systems, and intelligent operations frameworks. As cloud-based applications continue to expand in scale and complexity, robust and generalizable forecasting capabilities are expected to play a central role in enabling proactive system management, improving resource utilization, and supporting data-driven decision processes across related application domains.

References

[1] 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 Proc. AAAI Conf. Artificial Intelligence, vol. 35, no. 12, pp. 11106-11115, 2021.

[2] B. Lim, S. Ö. Arık, N. Loeff and T. Pfister, "Temporal Fusion Transformers for Interpretable Multi-horizon Time Series Forecasting," International Journal of Forecasting, vol. 37, no. 4, pp. 1748-1764, 2021.

[3] H. Wu, J. Xu, J. Wang and M. Long, "Autoformer: Decomposition Transformers with Auto-Correlation for Long-Term Series Forecasting," in Proc. Advances in Neural Information Processing Systems (NeurIPS), 2021.

[4] T. Zhou, Z. Ma, Q. Wen, X. Wang, L. Sun and R. Jin, "FEDformer: Frequency Enhanced Decomposed Transformer for Long-Term Series Forecasting," in Proc. 39th International Conference on Machine Learning (ICML), pp. 27268-27286, 2022.

[5] Y. Nie, N. H. Nguyen, P. Sinthong and J. Kalagnanam, "A Time Series is Worth 64 Words: Long-Term Forecasting with Transformers," in Proc. International Conference on Learning Representations (ICLR), 2023.

[6] Q. Wen, T. Zhou, C. Zhang, W. Chen, Z. Ma, J. Yan and L. Sun, "Transformers in Time Series: A Survey," in Proc. 32nd International Joint Conference on Artificial Intelligence (IJCAI), pp. 6778-6786, 2023.

[7] H. Ismail Fawaz, G. Forestier, J. Weber, L. Idoumghar and P.-A. Muller, "Deep Learning for Time Series Classification: A Review," Data Mining and Knowledge Discovery, vol. 33, no. 4, pp. 917-963, 2019.

[8] B. Carbonneau, K. Laframboise and R. Vahidov, "Application of Machine Learning Techniques for Supply Chain Demand Forecasting," European Journal of Operational Research, vol. 184, no. 3, pp. 1140-1154, 2008.

[9] S. M. Chen and N. Y. Chung, "Forecasting Enrollments Using High-Order Fuzzy Time Series," International Journal of Approximate Reasoning, vol. 36, no. 1, pp. 1-21, 2004.

[10] T. W. Liao, "Clustering of Time Series Data — A Survey," Pattern Recognition, vol. 38, no. 11, pp. 1857-1874, 2005.

[11] V. Chandola, A. Banerjee and V. Kumar, "Anomaly Detection: A Survey," ACM Computing Surveys, vol. 41, no. 3, Art. no. 15, 2009.

[12] A. A. Azzouni and G. Pujolle, "A Long Short-Term Memory Recurrent Neural Network Framework for Network Traffic Matrix Prediction," arXiv:1705.05690, 2017.