RESEARCH ARTICLE

Metric Interaction Learning for Performance Incident Discovery in Non-Stationary Cloud Services

Daniel Thompson[1], Rachel Bennett[2], Christopher Wilson[3*]

Department of Electrical and Computer Engineering, The University of British Columbia, Vancouver, BC V6T 1Z4, Canada

*Corresponding author: c.wilson@ubc.ca

Abstract

Large-scale cloud service platforms generate continuous performance time series from CPU utilization, memory usage, request latency, disk I/O, network throughput, queue length, and service error logs. These metrics are strongly coupled, but their distributions change frequently due to workload migration, elastic scaling, software release cycles, and diurnal user demand. This study proposes a cross-metric dependency fusion model for non-stationary cloud service performance anomaly detection. The method constructs a dynamic metric-dependency matrix through window-level mutual information and temporal attention. A multi-scale sequence encoder is used to capture short-term spikes, periodic workload changes, and long-range service degradation patterns. To reduce false alerts under normal workload shifts, a regime-conditioned residual scoring module adjusts anomaly thresholds according to recent workload states. Experiments are conducted on a cloud service monitoring dataset containing 2,460 microservice instances, 186 service groups, and 96 performance metrics collected at one-minute intervals over 120 days. The dataset contains 169 million timestamped metric records and 2,340 labeled incidents, including memory leakage, cascading latency, abnormal queue accumulation, failed deployment, and disk saturation. The proposed model reduces median incident detection delay from 14.2 minutes to 4.8 minutes compared with a static Transformer reconstruction model. The false alert volume decreases to 2.6 alerts per service group per week. The model reaches a Matthews correlation coefficient of 0.921 and a Cohen’s kappa of 0.903 across mixed workload regimes. Online inference processes 38,000 metric points per second, with a median scoring latency of 24 ms per service group. The results show that cross-metric dependency fusion can improve cloud performance anomaly detection under non-stationary service conditions.

Keywords: Cloud service monitoring; non-stationary time series; performance anomaly detection; cross-metric dependency; temporal attention; residual scoring; microservice observability.

Introduction

Cloud service platforms continuously generate large volumes of monitoring time-series data, including CPU utilization, memory consumption, request latency, disk input/output, network traffic, queue length, and error-log frequency. These metrics are strongly interconnected because a change in one system component can propagate across multiple services and infrastructure layers [1,2]. Consequently, service incidents commonly manifest as coordinated deviations across several metrics rather than as isolated changes in a single variable. Recent studies on artificial intelligence for IT operations and multivariate time-series anomaly detection have demonstrated the value of temporal representation learning for service monitoring, failure identification, and early incident warning [3,4]. Causal inference has also been incorporated into multivariate anomaly detection to distinguish direct cross-variable effects from coincidental correlations, thereby supporting more fine-grained identification of abnormal variables and their possible propagation relationships [5]. Comprehensive evaluations of time-series anomaly detection methods further indicate that reliable detection requires the joint consideration of temporal dependence, variable interaction, anomaly duration, and operational context [6].

Deep learning has substantially improved the ability of anomaly detection systems to model complex and nonlinear monitoring patterns. Transformer-based architectures can capture long-range temporal dependencies, convolution-based models are effective in extracting local fluctuations and short-duration abnormal patterns, and graph-based methods explicitly represent interactions among system metrics or service components [7,8]. These approaches have achieved promising results in identifying collective anomalies that cannot be detected through independent metric analysis [9,10]. However, many existing methods construct metric relationships from the entire training dataset or learn a fixed dependency structure during offline training. Such designs implicitly assume that the statistical distributions and interaction patterns of monitoring metrics remain relatively stable over time [11,12]. This assumption is often unsuitable for modern cloud environments. Workload migration, elastic resource scaling, software releases, service configuration changes, traffic bursts, and daily or weekly user-demand cycles can continuously alter the normal operating distributions of monitored metrics [13,14]. The strength and direction of dependencies among CPU load, memory usage, request latency, network traffic, and queue length may therefore vary across workload states [15,16]. A dependency pattern that is normal during a high-traffic period may appear anomalous when evaluated using a model trained mainly on low-load observations [17,18]. Similarly, fixed anomaly thresholds and static reconstruction models may assign high anomaly scores to legitimate workload transitions, producing excessive false alerts and increasing the burden on operations teams [19,20]. Models that adapt only their prediction errors without updating cross-metric relationships may also fail to detect incidents whose primary evidence lies in an abnormal change in metric coordination rather than in a large deviation of individual metric values. Another practical limitation is that cloud incidents occur at different temporal scales. Sudden resource exhaustion or network interruption may produce short and sharp spikes, whereas memory leakage, storage degradation, or increasing service contention may develop gradually over extended periods. Periodic workload changes can further resemble recurring anomalies if the model does not distinguish operational regimes [21,22]. An effective monitoring method must therefore capture local fluctuations, long-term trends, and periodic variations while maintaining sufficiently low computational overhead for online deployment. Detection performance should also be evaluated beyond conventional accuracy measures because delayed alarms, excessive alert volume, limited throughput, and high scoring latency can substantially reduce the operational value of a model [23,24].

This study aims to develop a cross-metric dependency fusion method for anomaly detection in non-stationary cloud service monitoring. The proposed method constructs a dynamic metric-dependency matrix by combining window-level mutual information with temporal attention, allowing cross-metric relationships to be updated as service conditions change. A multi-scale sequence encoder is introduced to represent short-lived spikes, periodic workload fluctuations, and gradual service degradation within a unified framework. A regime-conditioned residual scoring module is further designed to adapt anomaly thresholds to different workload states and reduce false alerts caused by normal operational transitions. The method is evaluated using a large-scale cloud monitoring dataset containing 2,460 microservice instances, 186 service groups, 96 monitoring metrics, and approximately 169 million timestamped records. Detection delay, false alert volume, Matthews correlation coefficient, Cohen's kappa, online throughput, and scoring latency are jointly examined to assess both analytical performance and deployment feasibility. By dynamically modeling cross-metric dependencies and workload regimes, this study seeks to improve the distinction between genuine service incidents and legitimate workload shifts, providing a practical, scalable, and operationally reliable solution for anomaly detection in continuously changing cloud service environments.

Materials and Methods

2.1 Sample and Study Area

This study uses performance monitoring data from a large cloud platform. The dataset contains 2,460 microservice instances organized into 186 service groups. Each instance records 96 metrics, including CPU usage, memory usage, request latency, disk I/O, network throughput, queue length, and error logs. Data were collected at one-minute intervals for 120 consecutive days, producing 169 million timestamped records. The services are hosted across multiple data centers and virtual environments, providing diverse workload patterns, auto-scaling events, and daily demand variations. Labeled incidents include memory leaks, cascading latency, abnormal queue growth, failed deployments, and disk saturation, totaling 2,340 events.

2.2 Experimental Design and Control Groups

The experiment includes a main test group with the proposed cross-metric dependency fusion model and two control groups. The first control uses a static Transformer-based reconstruction model, capturing temporal trends without dynamic metric relations. The second control uses a multi-scale sequence encoder without temporal attention or workload-aware scoring. This setup allows evaluating the effect of metric-dependency modeling and threshold adaptation. All models are trained on 80% of the data and tested on the remaining 20%. Random shuffling ensures exposure to diverse workloads and reduces bias.

2.3 Measurement Methods and Quality Control

Metric measurements were validated against platform logs and historical incident reports. Missing values, less than 1% of the dataset, were filled using linear interpolation from adjacent timestamps and correlated metrics. Metrics were normalized to standard ranges to avoid scale differences. Anomalies were checked by comparing automated labels with recorded incidents; any inconsistencies were manually reviewed. Latency and throughput metrics were aggregated over one-minute windows to reduce noise. The scoring module was calibrated on a validation set to prevent normal workload changes from producing false alerts.

2.4 Data Processing and Model Formulation

All metrics were normalized using min-max scaling:

$$X'_{i,t} = \frac{X_{i,t} - \min(X_i)}{\max(X_i) - \min(X_i)}$$

where $X_{i,t}$ is the raw value of metric i at time t, and $min(X_i)$,$max(X_i)$ are the minimum and maximum values for that metric. A dynamic metric-dependency matrix $M_t \in \mathbb{R}^{N \times N}$ is computed using window-level mutual information and temporal attention, with N=96N = 96N=96 metrics. The anomaly score for a service group at time t is:

$$S_t = \alpha \sum_{i=1}^{N} |X'_{i,t} - \hat{X}'_{i,t}| + (1-\alpha) D_{\mathrm{KL}}(P_t \| Q_t)$$

where $\hat{X}'_{i,t}$ is the predicted normalized value, and $D_{KL}(P_t \| Q_t)$ is the Kullback–Leibler divergence between the current metric distribution P_t and historical baseline Q_t. The weight $\alpha \in [0,1]$ balances local deviations and global distribution shifts.

2.5 Implementation and Evaluation

Models were implemented in Python with PyTorch. Training used batch size 512 and Adam optimizer with a learning rate of 0.001. Evaluation metrics included detection delay, false alert count, Matthews correlation coefficient, Cohen's kappa, online throughput, and scoring latency. Online performance was measured by feeding simulated streaming data at production rate. Control groups were evaluated using the same metrics to quantify improvements from dynamic metric-dependency modeling and workload-aware scoring.

Results and Discussion

3.1 Overall detection performance under mixed workloads

The proposed cross-metric dependency fusion method achieved clear improvements in detecting cloud service incidents under varying workloads. Compared with a static Transformer reconstruction model, the median incident detection delay decreased from 14.2 minutes to 4.8 minutes. This indicates that the method can identify abnormal changes faster, including memory leaks, queue growth, disk saturation, and cascading latency across service groups [25,26]. The false alert rate dropped to 2.6 alerts per service group per week, which is important for operational efficiency. The method also reached a Matthews correlation coefficient of 0.921 and Cohen's kappa of 0.903, showing reliable detection under different workload patterns. Transformer-based reconstruction alone captures temporal patterns but does not explicitly track how metrics interact, which explains its longer detection delay. Fig. 1 illustrates the Transformer structure used in prior studies for multivariate time-series anomaly detection.

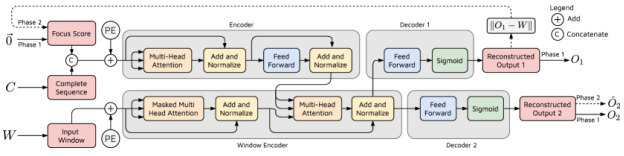

Fig. 1. Transformer framework for detecting anomalies in multiple time series metrics.

3.2 Effect of cross-metric dependency fusion

Using a dynamic metric-dependency matrix improved detection of incidents that are not evident from individual metrics. In cloud services, abnormal latency often coincides with changes in CPU load, queue length, network throughput, and error rate. Fixed metric structures cannot reflect these rapidly changing relations during scaling, migrations, or deployments. The proposed method updates metric dependencies for each window using mutual information and temporal attention [27,28]. This allows the detector to capture both individual metric deviations and abnormal metric relationships. Fig. 2 provides a reference from graph-based anomaly detection, showing learned relations and attention weights that help identify abnormal behavior. Our approach extends this idea to cloud service metrics while updating dependencies based on recent workloads.

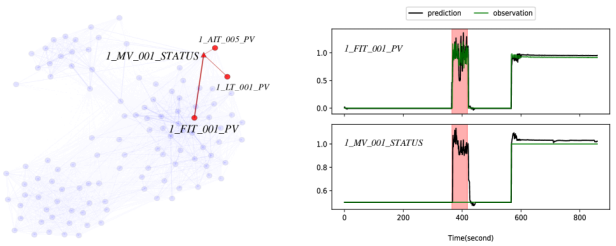

Fig. 2. Learned metric dependency graph with attention scores for identifying anomalies.

3.3 Stability under non-stationary service conditions

The regime-conditioned residual scoring module reduces false alerts caused by normal workload changes. Cloud services experience daily cycles, scaling, deployments, and service migrations, which shift metric distributions without indicating real incidents. Static thresholds are sensitive to these changes and may trigger excessive alerts. The proposed module adjusts residual thresholds based on current workload states. This allows the method to separate true incidents from normal variations. The decrease in weekly false alerts shows that workload-aware scoring is necessary for production systems. Compared with earlier studies, which mainly report point-level accuracy on public datasets, this study focuses on event delay, false alert rate, and scoring latency, which are more relevant to cloud operations [29,30].

3.4 Comparison with prior methods and practical value

Compared with Transformer reconstruction and graph-based methods, the proposed approach balances detection accuracy, speed, and practical utility. Transformers capture long-term temporal patterns but may miss changing metric relationships. Graph-based methods describe metric relations but often assume fixed graphs. The proposed method combines both advantages by updating metric dependencies over time and adjusting residual scores according to workload conditions. Online inference processed 38,000 metric points per second, with a median scoring latency of 24 ms per service group, supporting near-real-time operation. Limitations include testing on a single cloud platform and using only metric-level inputs. Future work should include logs, traces, and deployment data to improve incident interpretation and generalization across platforms.

Conclusion

The cross-metric dependency fusion approach provides a practical method for detecting cloud service performance incidents under changing workloads. By combining dynamic metric relations, multi-scale sequence encoding, and workload-aware residual scoring, the method captures both short-term spikes and long-term service degradation. Experiments show that it reduces detection delays, lowers false alerts, and maintains strong agreement with labeled incidents across various workloads. The main contribution of this study is explicitly modeling changing relationships among service metrics rather than relying on fixed thresholds or single-metric predictions. This improves understanding of how incidents propagate through interrelated system metrics. The method can be applied in cloud monitoring platforms and operations centers to support timely incident detection and faster fault response. Limitations include testing on a single cloud environment and relying mainly on metric-level data. Future work should validate the method across multiple platforms and incorporate logs, traces, deployments, and service topology to improve detection accuracy and incident explanation.

References

Qi, C., & Qiao, X. (2026, May). Design Patterns for Multi-Agent Systems in Production. In 2026 International Conference on Intelligent Systems, Automation and Control (ISAC) (pp. 198-202). IEEE.

Shamsi, N., & Helmrich, A. (2025). Interdependency classification: a framework for infrastructure resilience. Environmental Research: Infrastructure and Sustainability, 5(1), 015009.

Su, D., & Shi, X. Optimizing Family–School Collaboration to Improve Educational and Social Outcomes for Children with Intellectual Disabilities in Inclusive Public Schools.

Yahya, M. A., Moya, A. R., & Ventura, S. (2025). Deep learning for multivariate time series anomaly detection: An evaluation of reconstruction-based methods. Artificial Intelligence Review, 58(12), 400.

Chen, X., Xiao, H., Zeng, Z., Zhang, S., & Xiao, R. (2025). Fine-Grained Multivariate Time Series Anomaly Detection via Causal Inference. Knowledge-Based Systems, 114765.

Ho, T. K. K., Karami, A., & Armanfard, N. (2025). Graph anomaly detection in time series: A survey. IEEE Transactions on Pattern Analysis and Machine Intelligence.

Shao, W. (2026, March). Interpretable Ensemble Learning for Network Traffic Anomaly Detection: A SHAP-based Explainable AI Framework for Embedded Systems Security. In 2026 14th International Symposium on Digital Forensics and Security (ISDFS) (pp. 1-4). IEEE.

Gao, G., Gao, R., Lu, C., Gao, R., & Kuang, Y. (2026, March). Security Governance Methods and Quantitative Evaluation for Enterprise SMS and Verification Code Systems. In 2026 International Conference on Generative Artificial Intelligence and Information Security (GAIIS) (pp. 455-458). IEEE.

Khanizadeh, F., Ettefaghian, A., Wilson, G., Shirazibeheshti, A., Radwan, T., & Luca, C. (2025). Smart data-driven medical decisions through collective and individual anomaly detection in healthcare time series. International Journal of Medical Informatics, 194, 105696.

Xu, T., Zhu, W., & Zhang, J. (2026). A Study on the Application of Alternative Data in Credit Assessment for the Unbanked Population.

Fu, Y., Gui, H., Li, W., & Wang, Z. (2020, August). Virtual Material Modeling and Vibration Reduction Design of Electron Beam Imaging System. In 2020 IEEE International Conference on Advances in Electrical Engineering and Computer Applications (AEECA) (pp. 1063-1070). IEEE.

Pelosi, D., Cacciagrano, D., & Piangerelli, M. (2025). Explainability and interpretability in concept and data drift: a systematic literature review. Algorithms, 18(7), 443.

Chen, F., Yue, L., Xu, P., Liang, H., & Li, S. (2025). Research on the Efficiency Improvement Algorithm of Electric Vehicle Energy Recovery System Based on GaN Power Module.

Liang, R., Feifan, F. N. U., Liang, Y., & Ye, Z. (2025). Emotion-Aware Interface Adaptation in Mobile Applications Based on Color Psychology and Multimodal User State Recognition. Frontiers in Artificial Intelligence Research, 2(1), 51-57.

Gajanin, R., Marcelino, C., & Nastic, S. (2025). Performance Isolation for Serverless Functions. IEEE Transactions on Services Computing.

Wang, Y., Chen, J., Arias, R., Wang, Y., & Yin, X. (2026). Development and Validation of a Patient-Friendly Digital Assessment Platform for Precision Screening of Oral Anti-Obesity Medications (AOMs).

Jiao, Y., Zhao, B., Wang, A., & Shi, T. (2026). Construction and Empirical Study of a Modularized Teaching System for Art Courses Based on a Unified Training Pathway.

Ram, A., Chakraborty, S. K., Banerjee, A., & Mahato, G. K. (2025). Formulating and analysis of traffic flow to secure software-defined network (SDN) using recursive network (RN) learning method: A. Ram et al. The Journal of Supercomputing, 81(5), 654.

Yin, J., Rao, H., & Huang, Y. (2026, March). Dynamic Modeling and Heterogeneity Analysis of Platform User Behavior Time Series. In 2026 International Conference on AI in Education Technology and Applications (AIETA) (pp. 30-33). IEEE.

Shakil, S. M., Hossain, A., & Rahman, M. M. (2025). An empirical evaluation of anomaly detection techniques in smart grid systems using real-time operational data. American Journal of Advanced Technology and Engineering Solutions, 1(02), 95-134.

Yin, J., Huang, Y., & Rao, H. (2026). A Study on the Dynamic Evolution of Learning Behavior and Outcome Prediction in Digital Educational Environments. Available at SSRN 6607139.

Yang, J. (2026). Computational Analysis of How Digital Non-Clinical Communication Tools Influence Social Participation Among Older Adults in Community-Based Elderly Care. Available at SSRN 6682838.

Tariq, S., Baruwal Chhetri, M., Nepal, S., & Paris, C. (2025). Alert fatigue in security operations centres: Research challenges and opportunities. ACM Computing Surveys, 57(9), 1-38.

Zhang, Z. (2026). A Study on Multimodal Product Understanding and Assembly Guidance Optimization in e-commerce for Complex Consumer Goods. Available at SSRN 6734559.

Bittla, S. R. (2025). Predicting Failures with AI/ML Analytics. In AI-Driven Software Testing: Transforming Software Testing with Artificial Intelligence and Machine Learning (pp. 435-454). Berkeley, CA: Apress.

Johnphill, O., Sadiq, A. S., Kaiwartya, O., & Taheir, M. A. (2026). Cross: a cloud-native approach to automated remediation and self-healing in cyber-physical systems. Cybersecurity, 9(1), 124.

Zhang, Z., Gao, Y., & Tong, Y. (2026). CausalML4CX: A Causal Inference and Machine Learning Framework for Customer Experience Optimization with Application in Banking.

Qi, C., & Qiao, X. (2026). Building and Operating a Large Scale Multi-Agent System: A Case Study from Industry. Available at SSRN 6795198.

Korostenskyi, R., & Olenych, I. (2025, October). EDGE Approach to Early Detection of Anomalies in Electricity Consumption. In 2025 IEEE 6th KhPI Week on Advanced Technology (KhPIWeek) (pp. 1-5). IEEE.

Xu, T., Zhang, J., & Zhu, W. (2026). Fairness-Accuracy Trade-offs and Calibration Mechanisms in Machine Learning-Based Subprime Credit Scoring. Available at SSRN 6893759.