**Artificial Intelligence and Computing Innovations**

Intelligence Science Publishing

https://www.ispublishing.com

ISSN: 3070-4944

Vo.5, No.7, 2025

# *Dynamic Graph Structure Modeling for Anomaly Detection in Complex Cloud Platforms*

**Guoze Feng**[1*]

[1]Syracuse University, Syracuse, USA

**Corresponding author: tonyfeng0830@gmail.com*

***Abstract:*** To address the challenges of anomaly detection in complex cloud platforms, stemming from the continuous evolution of service dependencies, complex anomaly propagation paths, and high coupling of multiple operational states, this paper proposes a dynamic graph structure modeling method that integrates service topology evolution perception with deep representation learning from graph neural networks. This method first abstracts the complex cloud platform into a dynamic attribute graph at different time steps. By unifying the representation of service nodes, interaction relationships, and operational characteristics, a time-varying structural description framework for anomaly detection is constructed. Based on this, a topology evolution description mechanism is introduced to model service dependency reconstruction, relationship changes, and local structural perturbations at adjacent time steps, thereby enhancing the model's ability to perceive dynamic structural changes. Subsequently, a relationship graph aggregation strategy is used to perform deep propagation and semantic refinement of high-order dependency information between nodes, improving the identification of correlated and propagated anomalies. Furthermore, a temporal fusion mechanism is used to jointly model the current observation state and historical representation information, enhancing the model's ability to distinguish between persistent anomalies, gradual anomalies, and short-term fluctuations. Finally, the constructed method can effectively characterize and accurately identify anomaly risks in the dynamic environment of complex cloud platforms. Comparative experimental results show that the proposed method outperforms the other method in overall detection performance, verifying the effectiveness and applicability of the proposed dynamic graph structure modeling framework in anomaly detection tasks on complex cloud platforms.

***Keywords:*** Time-varying service dependency; structural evolution awareness; relational representation aggregation; intelligent operation and maintenance anomaly identification.

## 1 Introduction

As cloud computing continues to evolve towards large-scale, distributed, and service-oriented architectures, complex cloud platforms have become crucial infrastructure supporting key business

scenarios such as government affairs, finance, industrial internet, intelligent manufacturing, and online services[1,2]. Compared to traditional monolithic information systems, modern cloud platforms typically consist of numerous interconnected microservices, container instances, virtualized resources, network components, and middleware, exhibiting significant characteristics such as high concurrency, strong dynamism, cross-layer linkage, and heterogeneous collaboration. In this context, fluctuations in the state of any local component within the platform can rapidly propagate along service dependency chains, resource sharing chains, and call propagation chains, leading to performance degradation, service jitter, link blockage, and even system-level failures. Therefore, conducting efficient, accurate, and structurally-aware anomaly detection for complex cloud platforms has become one of the core issues for ensuring stable cloud environment operation and improving intelligent operation and maintenance capabilities[3].

However, the key challenges of anomaly detection in complex cloud platforms stem not only from the growth in data volume but also from the continuous evolution of system structure and operational behavior. On the one hand, service instances in cloud platforms change frequently due to elastic scaling, failover, version updates, and traffic scheduling. The dependencies, call intensity, and interaction patterns between services are not static, leading to significant time-dependent and dynamic reconfiguration characteristics in the system topology[4]. On the other hand, anomaly events often do not occur in isolation but are accompanied by associated disturbances between multiple nodes, links, and layers of resources, exhibiting characteristics of concealment, propagation, and non-stationarity. Traditional methods based on threshold rules, statistical features, or static structure modeling struggle to fully characterize the temporal correlations and topological dependencies of anomaly propagation in complex cloud platforms, and also fail to reliably capture deep-seated anomaly patterns in scenarios with rapidly changing structures. This has transformed anomaly detection from a simple behavioral recognition problem into a problem of collaborative modeling of temporal dynamics, structural relationships, and semantic representations[5].

From a methodological development perspective, graph structure modeling provides a new theoretical perspective and technical path for anomaly detection in complex cloud platforms. Cloud platforms can essentially be abstracted as complex graph systems composed of service nodes, resource entities, and interaction relationships[6]. Graph representation can not only characterize structural information such as service calls, deployment dependencies, resource contention, and link associations, but also provide an interpretable relational foundation for anomaly localization, impact propagation analysis, and root cause identification. Building upon this, introducing a service topology evolution awareness mechanism can further incorporate structural changes over time into the modeling process. This allows the model to focus not only on the local connectivity state at a given moment but also on capturing deep dynamic features such as dependency restructuring, hotspot migration, and interaction pattern drift during continuous system operation. Simultaneously, the deep representation learning capability of graph neural networks helps extract high-order semantic representations from multi-hop neighborhoods and complex association paths, thereby improving the ability to identify weak anomalies, combined anomalies, and propagated anomalies. Therefore, collaborative modeling based on service

topology evolution and deep graph representation learning has clear theoretical necessity and practical relevance[7].

Developing dynamic graph structure modeling for anomaly detection in complex cloud platforms not only helps advance intelligent cloud platform operation and maintenance methods from shallow indicator monitoring to structure-driven deep understanding, but also plays a significant role in improving the security, reliability, and autonomous operation and maintenance capabilities of digital infrastructure. From an academic perspective, this direction helps expand the cross-disciplinary integration of anomaly detection research in areas such as dynamic graph learning, time-series graph modeling, and complex system representation learning, promoting the development of intelligent analysis methods for open environments, dynamic structures, and multi-source coupling scenarios. From an application perspective, constructing anomaly detection methods capable of perceiving service topology evolution and learning deep correlation patterns helps improve the ability of complex cloud platforms to identify potential risks in advance, enhances the efficiency of responding to sudden anomalies and cascading failures, and reduces operation and maintenance costs and business interruption risks. With the increasing prevalence of cloud-native architectures, edge-cloud collaboration, and large-scale service orchestration, this type of research has continuous and far-reaching value in building a highly resilient, highly available, and highly intelligent next-generation cloud platform operation and maintenance assurance system.

## 2 Methodology

Complex cloud platforms are modeled as dynamically evolving service interaction systems in which runtime entities, dependency relations, and multivariate operational states change continuously with scheduling, scaling, routing, and workload migration. To preserve both structural semantics and temporal continuity, the platform at time step $t$ is represented as a directed attributed graph, where service instances, middleware components, and infrastructure units are unified as nodes, while invocation, deployment dependency, message exchange, and resource contention are encoded as typed edges. This paper also presents the overall model architecture, as shown in Figure 1.

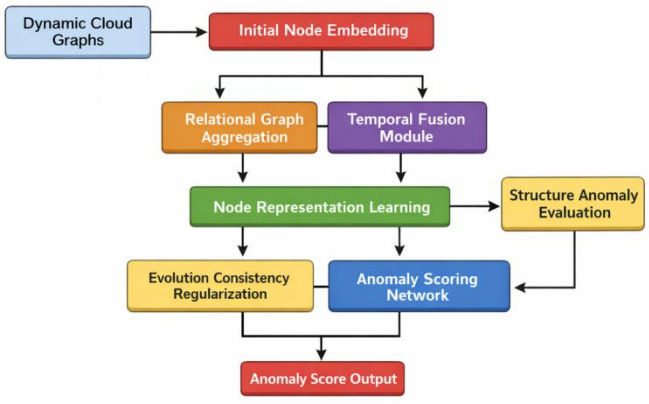



Figure 1. Overall model architecture

This formulation is introduced to avoid treating abnormal behavior as an isolated fluctuation in local metrics, since many cloud anomalies emerge through correlated disturbances propagated along service paths and cross-layer couplings. Formally, the dynamic system is defined as:

$$\mathcal{G}^{(t)} = \left(\mathcal{V}^{(t)}, \mathcal{E}^{(t)}, \boldsymbol{X}^{(t)}, \boldsymbol{A}^{(t)}\right) \tag{1}$$

where $\mathcal{V}^{(t)}$ and $\mathcal{E}^{(t)}$ denote the node and edge sets at time step $t$, while $\boldsymbol{X}^{(t)} \in \mathbb{R}^{|\mathcal{V}^{(t)}| \times d}$ and $\boldsymbol{A}^{(t)} \in \mathbb{R}^{|\mathcal{V}^{(t)}| \times |\mathcal{V}^{(t)}|}$ characterize node attributes and structural connectivity under the current service topology. Because the observed graph changes not only in edge intensity but also in relation composition, a topology evolution descriptor is further constructed to capture the magnitude and direction of structural variation between adjacent snapshots:

$$\boldsymbol{S}^{(t)} = \phi\left(\boldsymbol{A}^{(t)} - \boldsymbol{A}^{(t-1)}\right) \tag{2}$$

in which $\phi(\cdot)$ denotes a nonlinear projection used to compress local rewiring, edge birth, and edge disappearance into an evolution-aware structural signal. Rather than directly aggregating raw interactions that may be contaminated by transient burst noise, the method injects the topology evolution descriptor into the node representation process so that the model can distinguish persistent dependency drift from instantaneous metric volatility. The initial dynamic feature of node $i$ at time step $t$ is therefore written as:

$$\boldsymbol{h}_i^{(t,0)} = \sigma\left(\boldsymbol{W}_x \boldsymbol{x}_i^{(t)} + \boldsymbol{W}_s \boldsymbol{s}_i^{(t)} + \boldsymbol{b}\right) \tag{3}$$

with $\boldsymbol{x}_i^{(t)}$ denoting the original monitoring feature and $\mathbf{s}_i^{(t)}$ denoting the evolution-aware structural embedding extracted from $\boldsymbol{S}^{(t)}$, so that the encoded state already contains both runtime behavior and topology transition cues before graph propagation begins.

Dynamic resource load forecasting through representation learning further motivates constructing node states from multi-source operational observations, so that transient load changes and longer-term resource pressure can be separated within the evolution-aware embedding. This representation perspective helps preserve scalable runtime semantics before structural propagation is applied[8].

Subsequently, deep relational learning is performed over the time-varying graph in order to reveal higher-order dependency patterns that cannot be detected from single-hop interactions alone. Unlike static graph encoding, the present design assigns adaptive importance to neighboring services according to both current connectivity and topology evolution intensity, since abnormal propagation in cloud systems often follows a selective rather than uniformly diffusive path. The message passed from node $j$ to node $i$ at layer $l$ and time step $t$ is defined as:

$$\alpha_{ij}^{(t,l)} = \frac{exp\left(\psi\left(\boldsymbol{h}_i^{(t,l)}, \boldsymbol{h}_j^{(t,l)}, r_{ij}^{(t)}\right)\right)}{\sum_{k \in \mathcal{N}_i^{(t)}} exp\left(\psi\left(\boldsymbol{h}_i^{(t,l)}, \boldsymbol{h}_k^{(t,l)}, r_{ik}^{(t)}\right)\right)} \tag{4}$$

where $\psi(\cdot)$ scores the interaction relevance using source and target embeddings together with the relation state $r_{ij}^{(t)}$, thereby enabling the aggregation coefficient $\alpha_{ij}^{(t,l)}$ to respond to service dependency

asymmetry and structural perturbation. Guided by this adaptive weighting, the representation update is expressed as:

$$\boldsymbol{h}_i^{(t,l+1)} = \sigma\left(\sum_{j\in\mathcal{N}_i^{(t)}} \alpha_{ij}^{(t,l)}\, \boldsymbol{W}^{(l)} \boldsymbol{h}_j^{(t,l)} + \boldsymbol{U}^{(l)} \boldsymbol{h}_i^{(t,l)}\right) \tag{5}$$

so that neighborhood evidence and self-state preservation are jointly considered. Such a design is meaningful because cloud anomalies frequently manifest as a tension between external disturbance accumulation and local service resilience, and this balance cannot be adequately modeled when self-information is overshadowed by indiscriminate neighbor pooling.

Graph neural modeling of traffic propagation and topological interactions in cloud networks provides a complementary principle for relation-aware aggregation: messages should reflect both direct service connectivity and the wider propagation paths through which congestion or faults influence dependent components. This strengthens sensitivity to cross-layer topology disturbances in the dynamic cloud graph[9].

Meanwhile, temporal dependency is incorporated to connect the latent states of the same node across successive graph snapshots, thereby preventing the detector from overreacting to short-lived perturbations while remaining sensitive to sustained anomalous evolution. Since the runtime trajectory of a cloud service is often shaped by historical load transfer, prior fault recovery, and repeated orchestration adjustments, the representation at time step $t$ is refined through a gated temporal fusion mechanism:

$$\boldsymbol{z}_i^{(t)} = \gamma_i^{(t)} \odot \boldsymbol{h}_i^{(t,L)} + \left(1-\gamma_i^{(t)}\right) \odot \boldsymbol{z}_i^{(t-1)} \tag{6}$$

in which $\boldsymbol{h}_i^{(t,L)}$ denotes the final graph-encoded state at the current snapshot, $\boldsymbol{z}_i^{(t-1)}$ carries historical memory, and the gate $\gamma_i^{(t)}$ determines whether recent structural evidence should dominate over past behavior. By blending current and previous representations in this manner, the method preserves long-range evolution regularity while attenuating random monitoring jitter, which is essential when distinguishing gradual degradation, cascading faults, and temporary service bursts in highly elastic cloud environments.

Transformer-based sequence modeling for Kubernetes anomaly detection supports encoding consecutive operational states as a structured temporal sequence, which helps distinguish persistent topology shifts from short-lived fluctuations[10]. Multi-scale Transformer forecasting of CPU utilization likewise motivates retaining both local bursts and slower workload cycles in temporal memory, improving the stability of anomaly assessment under changing cloud demand[11].

Finally, anomaly scoring is conducted on top of the fused dynamic representations so that the learned structure and evolution information can be transformed into a node-level risk signal suitable for platform-wide diagnosis. Instead of relying only on reconstruction discrepancy or metric threshold violation, the scoring function measures the deviation of each node from the learned normal relational manifold, allowing topological inconsistency and semantic mismatch to be evaluated simultaneously. The anomaly score of node $i$ at time step $t$ is computed as:

$$y_i^{(t)} = \sigma\left(\boldsymbol{w}_o^{\top} \boldsymbol{z}_i^{(t)} + b_o\right) \tag{7}$$

where $\mathbf{z}_i^{(t)}$ summarizes the integrated effect of runtime observation, topology evolution, graph dependency propagation, and temporal memory, and $y_i^{(t)}$ indicates the abnormality intensity under the current cloud state. To make the learned representation remain discriminative and temporally smooth at the same time, the training objective combines anomaly supervision with evolution consistency regularization:

$$\mathcal{L} = \mathcal{L}_{det} + \lambda \sum_{t=2}^{T} \sum_{i \in \mathcal{V}^{(t)}} \left\| \boldsymbol{z}_i^{(t)} - \boldsymbol{z}_i^{(t-1)} \right\|_2^2 \tag{8}$$

where $\mathcal{L}_{\text{det}}$ denotes the anomaly detection loss, and the second term constrains excessive representation drift unless justified by genuine structural change. Owing to this objective, the method is encouraged to detect anomalous behavior with sensitivity to evolving service topology while preserving semantic continuity across time, which is particularly important for complex cloud platforms whose fault patterns are often sparse, correlated, and dynamically propagated rather than statically localized. Finally, the overall algorithm flowchart is given, as shown in Figure 2.

**Algorithm 1** Topology-Evolution-Aware Dynamic Graph Anomaly Detection

**Require:** Dynamic graph sequence $\{\mathcal{G}^{(t)}\}_{t=1}^{T}$, propagation depth $L$, trade-off coefficient $\lambda$

**Ensure:** Anomaly scores $\{y_i^{(t)}\}$

Initialize model parameters and historical states $\mathbf{z}_i^{(0)} = \mathbf{0}$
**for** $t = 1$ to $T$ **do**
  Compute topology evolution descriptor from adjacent graph snapshots
  Generate initial node embeddings by fusing runtime features and evolution-aware structure signals
  **for** $l = 0$ to $L - 1$ **do**
    Perform relation-aware graph aggregation to update node representations
  **end for**
  Fuse current graph representations with historical states
  Predict node-level anomaly scores for all nodes at time step $t$
**end for**
Compute detection loss and evolution consistency regularization
Optimize the total objective and return anomaly scores

Figure 2. Overall algorithm flowchart

## 3 Experimental Results and Analysis

### 3.1 Dataset Introduction

This paper selects the open-source dataset GAIA as the foundation for experimental data. GAIA is a comprehensive public dataset built for AIOps scenarios. Its official documentation clearly states that the dataset can be used for tasks such as anomaly detection, log analysis, and fault localization. It contains multi-source runtime data and accompanying monitoring information from the microservice system MicroSS, and can effectively cover the characteristics of frequent service interactions, significant state evolution, and strong correlation of anomaly propagation in complex cloud platforms. Because this dataset simultaneously possesses system-level monitoring semantics, service-level runtime status, and multi-dimensional operation and maintenance observation information, it has a high degree of

compatibility with dynamic graph structure modeling tasks for anomaly detection in complex cloud platforms, and can provide reliable data support for service topology evolution awareness, node relationship modeling, and anomaly semantic representation learning.

### 3.2 Experimental Results and Analysis

To further illustrate the relationship between the proposed method and representative works in the same field, and to demonstrate the method's positioning within the research context of anomaly detection and dynamic graph structure modeling in complex cloud platforms, several representative related works are selected for cross-sectional analysis. These related works mainly focus on microservice anomaly detection, call chain graph modeling, graph neural network representation learning, and structured anomaly recognition for cloud environments. They are strongly correlated with the technical approach of this paper, which focuses on complex cloud platforms, converged service topology evolution perception, and deep graph representation learning. Their experimental results are shown in Table 1.

Table 1. Experimental results compared with other models

| Method | ACC | PRE | REC | F1 |
|---|---|---|---|---|
| Cai et al.[12] | 90.84 | 89.72 | 88.95 | 89.33 |
| Zhang et al.[13] | 92.16 | 91.48 | 90.87 | 91.17 |
| Xie et al.[14] | 91.43 | 90.65 | 90.12 | 90.38 |
| J. Chen et al.[15] | 92.74 | 91.96 | 91.51 | 91.73 |
| Tang et al.[16] | 91.88 | 91.07 | 90.44 | 90.75 |
| Ding et al.[17] | 93.21 | 92.53 | 91.94 | 92.23 |
| H. Chen et al.[18] | 93.46 | 92.88 | 92.31 | 92.59 |
| Ours | 95.18 | 94.67 | 94.21 | 94.44 |

Experimental results show that the proposed algorithm exhibits superior overall performance in complex cloud platform anomaly detection tasks, demonstrating that the method can more fully characterize the intrinsic relationship between service interaction, topological evolution features, and anomaly semantic representation. Because the model has formed a relatively complete collaborative mechanism in dynamic graph structure modeling, relation dependency aggregation, and temporal state fusion, it performs better in terms of anomaly identification accuracy, stability, and discriminative ability. This further indicates that the constructed method can more effectively adapt to the operating environment of complex cloud platforms characterized by multi-source heterogeneity, dynamic structural changes, and significant anomaly propagation correlations.

To further verify the actual contribution of each key component of the proposed method to the anomaly detection performance of complex cloud platforms, this paper conducts ablation analysis focusing on topology evolution awareness, relationship-aware graph aggregation, and temporal fusion mechanisms, and the results are summarized in Table 2. The relevant settings strictly correspond to the method design process, clearly reflecting the differences in the roles of different modules in dynamic graph structure representation, service dependency modeling, and anomaly semantic characterization.

Table 2. Ablation study results of the proposed method

| Ablation Setting | ACC | PRE | REC | F1 |
|---|---|---|---|---|
| w/o Topology Evolution Descriptor | 92.84 | 92.16 | 91.63 | 91.89 |
| w/o Relational Graph Aggregation | 93.27 | 92.58 | 92.05 | 92.31 |
| w/o Temporal Fusion Module | 93.61 | 92.97 | 92.42 | 92.69 |
| Ours | 95.18 | 94.67 | 94.21 | 94.44 |

As shown in Table 2, each component of the proposed method contributes to the overall performance, and the absence of any key module weakens the anomaly detection capability. The topological evolution descriptor plays a crucial role in characterizing structural changes between adjacent time steps; its absence significantly reduces the model's ability to perceive service dependency reconstruction and dynamic correlation perturbations. The relation graph aggregation module determines the effective transmission of high-order interaction information between nodes; its removal limits the model's depth of expression for complex propagation paths and local correlation anomalies. The temporal fusion module ensures continuous modeling between historical states and current observations, enabling the model to more stably characterize persistent and evolutionary anomalies during cloud platform operation. Overall, the complete model forms a more coordinated synergistic mechanism among dynamic graph structure representation, relation dependency modeling, and temporal state fusion, thus achieving superior comprehensive performance.

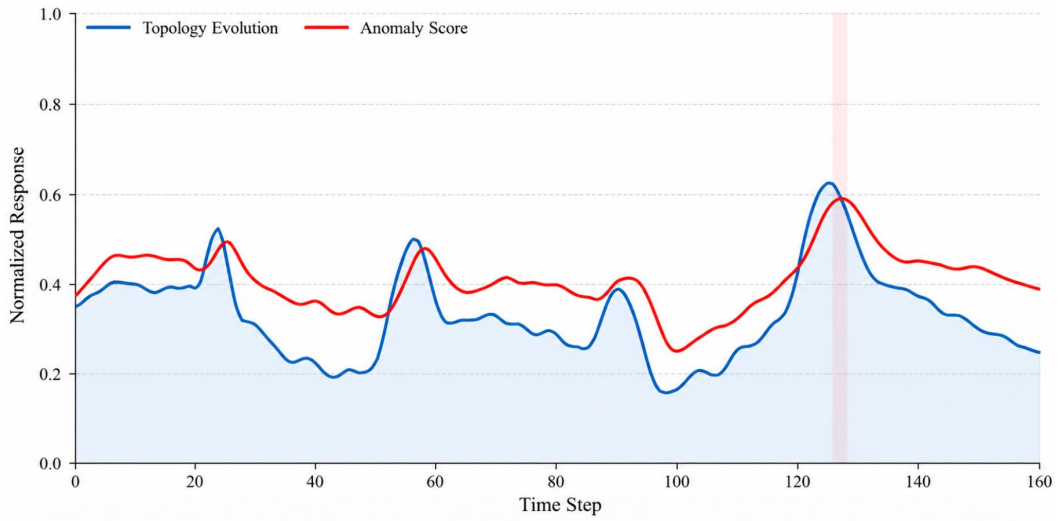



Figure 3. Temporal visualization results of anomaly score changes with service topology under the method presented in this paper.

Figure 3 shows that the anomaly scores obtained by the proposed method respond relatively continuously to changes in the service topology evolution process, exhibiting strong synchronicity and sensitivity during key structural fluctuation phases. This indicates that the constructed topology evolution awareness mechanism can effectively capture the intrinsic relationship between service dependency adjustments, interaction pattern changes, and anomaly risk accumulation in complex cloud platforms. It also demonstrates that the dynamic graph representation learning and temporal fusion process can effectively preserve the anomaly semantic information carried by structural changes. Overall, the

visualization results further demonstrate that the proposed method possesses strong anomaly awareness and temporal modeling capabilities in the dynamic environment of complex cloud platforms.

## 4 Conclusion

In summary, addressing the anomaly detection task of complex cloud platforms, this paper proposes a dynamic graph structure modeling method that integrates service topology evolution perception and deep representation learning of graph neural networks. This method, starting from the actual operating mechanism of complex cloud platforms, unifies service instances, interaction relationships, and state observations into a dynamic graph modeling framework. By introducing topology evolution description, relationship graph aggregation, and temporal fusion mechanisms, it enhances the model's ability to jointly characterize structural perturbations, dependencies, and evolutionary anomalies. Compared to modeling methods that rely solely on static structural information or local indicator changes, this method more closely reflects the real process of anomaly generation, diffusion, and accumulation in complex cloud platforms, providing a more structurally aware and dynamically adaptable solution to the anomaly detection problem in cloud environments.

From the perspective of methodological significance and application value, this work not only enriches the research paradigm of anomaly detection in complex cloud platforms but also provides practically feasible technical support for risk identification, anomaly early warning, and operational assurance in intelligent operation and maintenance scenarios. With the deepening of cloud-native architecture, microservice orchestration, elastic scheduling, and edge-cloud collaboration, the internal service relationships and resource coupling methods of platforms are becoming increasingly complex, and the manifestations of abnormal events are becoming more covert, diverse, and propagating. In this context, anomaly detection methods that can simultaneously perceive topological evolution, learn deep relational representations, and maintain temporal continuity have strong scenario adaptability and promotion potential. The dynamic graph structure modeling framework constructed in this paper helps to promote cloud platform anomaly detection from shallow monitoring and analysis to deep intelligent cognition of complex system structures, which is of positive significance for improving the stability, reliability, and autonomous operation and maintenance capabilities of digital infrastructure. It also has a certain reference value for related application areas such as cloud service management, industrial internet operation monitoring, and distributed system risk control.

Future work can be further developed in several directions. First, based on the existing dynamic graph modeling framework, the collaborative expression capability of cross-layer heterogeneous information can be further enhanced by incorporating log semantics, indicator sequences, call chain information, and resource behavior characteristics into a unified modeling process to further improve the depth of representation of complex anomaly types. Secondly, by combining real-world cloud platform scenarios with larger scales and higher dynamism, we can explore continuous learning and online update mechanisms for long-term evolutionary processes, thereby improving the model's long-term adaptability and deployment availability in open environments. Furthermore, we can further focus on the linkages between anomaly detection and downstream tasks such as root cause analysis, anomaly propagation tracing, and risk interpretation generation, pushing the methodology from single anomaly identification to intelligent operation and maintenance decision support across the entire process. As the operating

environment of complex cloud platforms continues to evolve, in-depth theoretical research and application expansion around dynamic graph structure modeling still have broad development prospects and continued research value.

## References


[1] X. Yan, Y. Jiang, W. Liu, D. Yi and J. Wei, "Transforming Multidimensional Time Series into Interpretable Event Sequences for Advanced Data Mining," 2024 5th International Conference on Intelligent Computing and Human-Computer Interaction (ICHCI), pp. 126-130, 2024.

[2] G. Somashekar, A. Dutt, M. Adak, et al., "Gamma: Graph neural network-based multi-bottleneck localization for microservices applications," in Proceedings of the ACM Web Conference 2024, 2024, pp. 3085-3095.

[3] J. Huang, Y. Yang, H. Yu, et al., "Twin graph-based anomaly detection via attentive multi-modal learning for microservice system," in 2023 38th IEEE/ACM International Conference on Automated Software Engineering, 2023, pp. 66-78.

[4] Z. Xie, C. Pei, W. Li, et al., "From point-wise to group-wise: A fast and accurate microservice trace anomaly detection approach," in Proceedings of the 31st ACM Joint European Software Engineering Conference and Symposium on the Foundations of Software Engineering, 2023, pp. 1739-1749.

[5] X. Liu, Y. Liu, M. Wei, et al., "LMGD: Log-Metric Combined Microservice Anomaly Detection Through Graph-Based Deep Learning," IEEE Access, vol. 12, pp. 186510-186519, 2024.

[6] M. Fan, X. Zhang, P. Wang, et al., "Multi-modal anomaly detection for microservice system through nested graph diffusion reconstruction," Applied Intelligence, vol. 55, no. 11, p. 784, 2025.

[7] L. Akmeemana, C. Attanayake, H. Faiz, et al., "GAL-MAD: Towards explainable anomaly detection in microservice applications using graph attention networks," arXiv preprint arXiv:2504.00058, 2025.

[8] T. Xia, "Dynamic Resource Load Forecasting in Cloud Computing: A Representation Learning Perspective," 2024.

[9] X. Huang, "Capturing Traffic Propagation and Topological Interactions in Cloud Networks via Graph Neural Networks," 2024.

[10] F. Chang, "Intelligent Anomaly Detection in Kubernetes Clusters Through Transformer-Based Sequence Modeling," 2024.

[11] Z. Wang and S. Lin, "Multi-Scale Transformer for CPU Utilization Time Series Forecasting in Cloud Computing," 2025.

[12] Y. Cai, B. Han, J. Su, et al., "TraceModel: An automatic anomaly detection and root cause localization framework for microservice systems," in 2021 17th International Conference on Mobility, Sensing and Networking, 2021, pp. 512-519.

[13] C. Zhang, X. Peng, C. Sha, et al., "Deeptralog: Trace-log combined microservice anomaly detection through graph-based deep learning," in Proceedings of the 44th International Conference on Software Engineering, 2022, pp. 623-634.

[14] Z. Xie, H. Xu, W. Chen, et al., "Unsupervised anomaly detection on microservice traces through graph vae," in Proceedings of the ACM Web Conference 2023, 2023, pp. 2874-2884.

[15] J. Chen, F. Liu, J. Jiang, et al., "TraceGra: A trace-based anomaly detection for microservice using graph deep learning," Computer Communications, vol. 204, pp. 109-117, 2023.

[16] H. Tang, Y. Guo, J. Yang, et al., "Microservice anomaly diagnosis with graph convolution network based on implicit microservice dependency," in Proceedings of the 2023 9th International Conference on Computing and Artificial Intelligence, 2023, pp. 437-441.

[17] S. Ding, E. Yuepeng, J. Zhang, et al., "Trace anomaly detection for microservice systems via graph-based semi-supervised learning," in 2024 27th International Conference on Computer Supported Cooperative Work in Design, 2024, pp. 2375-2380.

[18] H. Chen, P. Chen, B. Wang, et al., "Graph neural network based robust anomaly detection at service level in SDN driven microservice system," Computer Networks, vol. 239, p. 110135, 2024.