Transactions on Computational and Scientific Methods | Vo. 4, No. 5, 2024
ISSN: 2998-8780
https://pspress.org/index.php/tcsm
Pinnacle Science Press

Intelligent Anomaly Detection in Kubernetes Clusters Through Transformer-Based Sequence Modeling

Feng Chang[1*]
[1]North Dakota State University, Fargo, USA
*Corresponding author: fchangprivate@gmail.com

Abstract: To address the challenges of high-dimensionality, rapid state evolution, complex anomaly patterns, and long-term temporal dependencies in Kubernetes cluster runtime environments, a Transformer-based anomaly detection method for Kubernetes clusters is proposed. This method uses multivariate time series generated during cluster operation as the modeling object. First, monitoring metrics from different sources are uniformly organized and temporally aligned, transforming the original runtime state into a continuous sample representation suitable for deep sequence modeling. Then, heterogeneous monitoring features are projected onto a unified latent space through linear mapping, and positional encoding is used to preserve temporal order information, enabling the model to perceive the dynamic evolutionary features during anomaly formation. Building upon this, a multi-head self-attention mechanism is introduced to model the global dependencies between different variables at different times, enhancing the ability to express complex contextual relationships and potential anomaly precursors. Furthermore, a feedforward network and hierarchical encoding structure are combined to refine the temporal semantics layer by layer, resulting in a more stable and discriminative global state representation. Finally, sequence-level representation is used to complete anomaly identification, effectively distinguishing between normal and abnormal states of the Kubernetes cluster. The proposed method is well adapted to the characteristics of cloud-native environments, such as multi-component linkage, complex anomaly propagation paths, and the interplay of local fluctuations and global instability. It has strong applicability in anomaly feature mining, temporal correlation modeling, and overall state discrimination, and can provide effective support for intelligent monitoring and anomaly identification of Kubernetes clusters.

Keywords: Cloud-native operations and maintenance; multivariate time series analysis; self-attention mechanism; abnormal state identification

1. Introduction

With the rapid development of cloud-native technologies, Kubernetes has become a core infrastructure for container orchestration and service deployment, widely used in internet platforms, enterprise information systems, and edge computing environments. Through mechanisms such as resource scheduling, service orchestration, automatic scaling, and fault recovery, it significantly improves application delivery efficiency and system elasticity[1,2]. However, Kubernetes clusters are characterized by numerous components, complex interactions, and frequent dynamic state changes. During operation, they continuously generate a large amount of logs, metrics, events, and call information, resulting in high-dimensional, strongly coupled, and non-stationary system behavior. In this context, cluster anomalies often no longer manifest as sudden changes in a single node or metric, but rather as complex evolutionary processes across components, layers, and time ranges, posing new challenges to anomaly identification and operation maintenance.

Traditional anomaly detection methods often rely on static thresholds, manual rules, or shallow machine learning models. While effective in simple, stable system environments, they often struggle to effectively

model complex temporal dependencies and contextual relationships when faced with frequently changing service topologies, dynamically scaling load patterns, and multi-source heterogeneous monitoring data in Kubernetes clusters. On the one hand, abnormal behavior may accumulate gradually over a long period, and its precursor signals are usually weak and scattered, making them difficult to capture by local features. On the other hand, there are significant linkage effects between different components, and information from a single dimension cannot fully reflect the anomaly formation mechanism. Therefore, how to extract key patterns from multi-source operational data, characterize the anomaly propagation process, and improve detection accuracy and generalization ability in complex scenarios has become an important issue in Kubernetes cluster intelligent operation and maintenance research[3].

The Transformer model demonstrates superior global dependency capture capabilities and parallel computing advantages in the field of sequence modeling, providing a new research approach for Kubernetes cluster anomaly detection[4]. Compared to traditional time-series modeling methods, Transformer can model the correlation between time steps on a larger scale and has a stronger expressive power for long-term dependencies, contextual coupling, and multivariate interactions. This characteristic makes it more suitable for handling complex behavioral patterns in Kubernetes clusters composed of resource fluctuations, service imbalances, network jitter, scheduling anomalies, and component failures. Meanwhile, by leveraging attention mechanisms, the model can focus on more discriminative temporal segments and feature dimensions from a large number of dynamic observations, thus providing a more structured representation foundation for anomaly identification and offering a new technical path for understanding the changing patterns of system operating states.

Research on Transformer detection for anomaly detection in Kubernetes clusters has significant theoretical and practical value[5,6]. From a theoretical perspective, this direction helps promote the development of complex temporal modeling methods in intelligent operation and maintenance scenarios, deepening the understanding of anomaly patterns, cross-component correlation behaviors, and dynamic evolution laws in cloud-native systems. From an application perspective, building efficient anomaly detection methods suitable for Kubernetes clusters helps improve the operational stability of cloud platforms, enhance fault early warning capabilities, reduce manual troubleshooting costs, and provide support for resource optimization scheduling and service reliability assurance[7]. With the continued penetration of cloud-native infrastructure in critical industries, research on more adaptive, robust, and intelligent anomaly detection for Kubernetes clusters has become an important requirement for ensuring the safe and stable operation of modern distributed systems.

2. Datasets and Dataset Preprocessing

2.1 Dataset

This paper selects the Sock Shop Dataset as its dataset source. This dataset is built upon the open-source microservice application Sock Shop and deployed in a Kubernetes cluster environment, effectively characterizing the state changes and anomaly processes during multi-service collaborative operation in cloud-native scenarios. The public repository provides clear data descriptions, indicating that the dataset was collected from a Kubernetes cluster containing three machines, with seven consecutive days of data collection. The first four days represent normal data, while subsequent testing phases include container-level and node-level anomalies. Corresponding anomaly injection information files are also provided, demonstrating clear open-source availability and a solid foundation for anomaly labeling.

From the perspective of the paper's thematic relevance, the Sock Shop Dataset shows strong consistency with Transformer detection research focused on anomaly detection in Kubernetes clusters. Firstly, this dataset originates from the operation of a microservice system within a Kubernetes environment, reflecting the dynamic relationships between service instances, container resources, and node states within the cluster. Secondly, relevant publicly available research indicates that Sock Shop itself is a typical open-source

microservice benchmark system, and it has been used in Kubernetes environments in conjunction with Prometheus and Grafana for metric collection to characterize service behavior and abnormal states. This makes it suitable as a data foundation for time-series modeling and anomaly identification tasks. Based on its cloud-native deployment background, open-source nature, and clearly defined anomaly scenarios, selecting this dataset can well support the task setting and problem background of Kubernetes cluster anomaly detection research.

2.2 Dataset preprocessing

In the data preprocessing stage, the raw monitoring data collected from the Sock Shop Dataset is first uniformly organized. Multi-source indicator sequences are aligned based on timestamps, and continuous time-series samples are constructed according to fixed time windows to ensure consistency of information on different service instances, container resources, and node statuses at the same observation granularity. Subsequently, missing and anomalous null values are cleaned, and numerical features are standardized to eliminate dimensional differences, mitigating the impact of inconsistent numerical ranges of different monitoring indicators on model training. For label construction, based on anomaly injection information provided in publicly available data, normal operation intervals and anomaly occurrence intervals are mapped to corresponding category labels, thus forming supervisory signals usable for anomaly detection tasks. After these processes, the raw operational data is transformed into multivariate time-series samples with a unified structure, continuous temporal sequence, and suitable for Transformer model input, providing a reliable data foundation for subsequent anomaly pattern modeling.

3. Methodology

To characterize the evolving operational state of a Kubernetes cluster, the multivariate monitoring stream is first reorganized into fixed-length temporal segments so that resource usage, service behavior, and node-level signals can be modeled within a unified representation space. This design allows the detector to preserve local fluctuations while maintaining a consistent view of long-range dependencies, which is essential because abnormal conditions in cloud-native systems often emerge through gradual interactions across multiple components rather than through isolated spikes. The overall model architecture is also presented here, as shown in Figure 1.

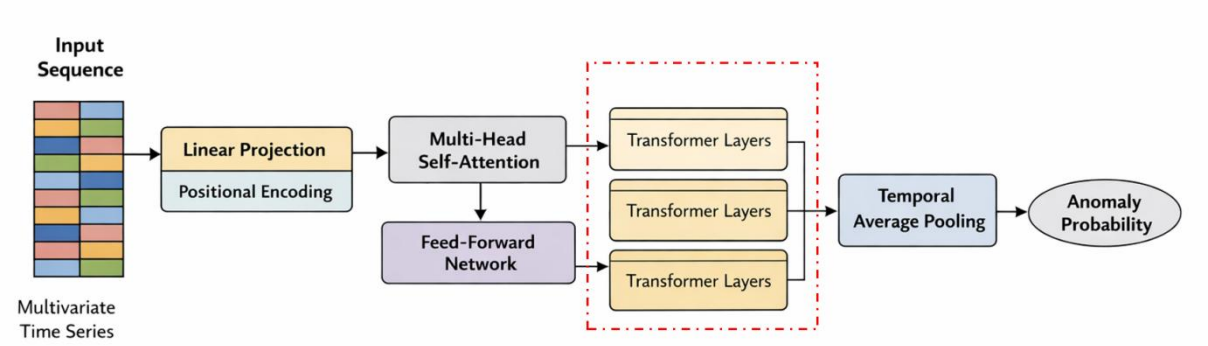

Figure 1. Overall model architecture

Given an input window with length T and feature dimension D, the raw sequence is denoted by:

$$\boldsymbol{X} = [\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_T] \in \mathbb{R}^{T \times D}$$

A linear projection is then used to map heterogeneous metrics into the latent space, where $\boldsymbol{W}_e \in \mathbb{R}^{D \times d}$ and $\boldsymbol{b}_e \in \mathbb{R}^d$ control the transformation from raw observations to hidden features:

$$\boldsymbol{H}^0 = \boldsymbol{X}\boldsymbol{W}_e + \mathbf{1}\boldsymbol{b}_e^{\top}$$

Since temporal order carries direct information about anomaly formation, positional encoding is injected into the latent sequence to retain the progression of system dynamics, and the encoded representation is defined as:

$$\boldsymbol{Z}^0 = \boldsymbol{H}^0 + \boldsymbol{P}$$

where $\boldsymbol{P} \in \mathbb{R}^{T \times d}$ denotes the positional matrix aligned with the observation window. By combining metric projection and temporal indexing at the input stage, the model obtains a structured sequence representation that is more suitable for capturing cross-time interactions in Kubernetes runtime data.

Unlike threshold-based strategies that mainly rely on instantaneous deviations, the core of the proposed method focuses on modeling contextual dependency through self-attention so that subtle precursor patterns can be amplified before a failure becomes explicit. For the lth Transformer block, the query, key, and value matrices are generated from the previous hidden state $\boldsymbol{Z}^{l-1}$ through three independent parameter groups:

$$\boldsymbol{Q}^l = \boldsymbol{Z}^{l-1}\boldsymbol{W}_Q^l, \quad \boldsymbol{K}^l = \boldsymbol{Z}^{l-1}\boldsymbol{W}_K^l, \quad \boldsymbol{V}^l = \boldsymbol{Z}^{l-1}\boldsymbol{W}_V^l$$

Global temporal correlation is subsequently quantified by scaled dot-product attention, where d_k is the key dimension and the normalization emphasizes informative interactions while suppressing irrelevant fluctuations:

$$Attn\left(\boldsymbol{Q}^l, \boldsymbol{K}^l, \boldsymbol{V}^l\right) = Softmax\left(\frac{\boldsymbol{Q}^l \boldsymbol{K}^{l\top}}{\sqrt{d_k}}\right)\boldsymbol{V}^l$$

This mechanism is especially meaningful in Kubernetes environments because an anomalous state at one time step may be strongly related to weak but correlated changes observed much earlier in containers, pods, or nodes. Consequently, the detector no longer treats each metric independently, but instead learns how distributed runtime signals cooperate over time to indicate instability.

Beyond single-head interaction modeling, multiple attention subspaces are introduced to enhance sensitivity to diverse abnormal patterns, including resource contention, service latency drift, and transient node disturbances. Multi-head aggregation is expressed as:

$$MHA\left(\boldsymbol{Z}^{l-1}\right) = Concat\left(head_1, head_2, \ldots, head_h\right)\boldsymbol{W}_O^l$$

where each $\mathtt{head}_i$ is computed by an independent attention mapping and $\boldsymbol{W}_O^l$ fuses the complementary dependencies learned from different subspaces. Residual propagation and nonlinear transformation are then employed to stabilize optimization and preserve discriminative information, yielding:

$$\boldsymbol{Z}^l = FFN\left(LayerNorm\left(\boldsymbol{Z}^{l-1} + MHA\left(\boldsymbol{Z}^{l-1}\right)\right)\right)$$

in which the feed-forward network refines token-wise semantic patterns after global interaction modeling. Such a stacked architecture increases representational depth without destroying temporal continuity, making it possible to distinguish normal operational variation from structurally meaningful anomaly evolution.

After hierarchical encoding, the hidden sequence is compressed into a global state vector so that the detector can produce an anomaly-oriented decision from the full observation context rather than from a single local signal. Temporal average pooling is adopted to summarize the encoded sequence, which gives:

$$\boldsymbol{g} = \frac{1}{T}\sum_{t=1}^{T} \boldsymbol{z}_t^L$$

with $\boldsymbol{z}_t^L$ denoting the output at time step t from the final Transformer layer. A sigmoid-based scoring function then converts the global representation into anomaly probability, where $\boldsymbol{w}_c$ and b_c are the classification parameters associated with the detection head:

$$\widehat{y} = \sigma\left(\boldsymbol{w}_c^{\top}\boldsymbol{g} + b_c\right)$$

Through this formulation, the method links sequence-level dependency learning with end-to-end anomaly discrimination, thereby enabling the model to identify whether the monitored Kubernetes segment reflects normal operation or an emerging abnormal condition. The overall design emphasizes semantic coherence, temporal sensitivity, and decision interpretability at the representation level, which together provide a principled basis for Transformer-based anomaly detection in cloud-native cluster environments.

4. Experimental Results and Analysis

To verify the effectiveness of the proposed method in Kubernetes cluster anomaly detection tasks, representative studies related to cloud-native microservice anomaly detection, multi-source time series modeling, and Transformer or graph structure modeling were selected as comparison objects. These related methods cover technical routes such as multimodal time series analysis, microservice call chain anomaly identification, federated anomaly detection, and multi-source fault diagnosis, comprehensively reflecting the mainstream research ideas and performance levels in this field. The experimental results are shown in Table 1.

Table 1. Experimental results compared with other models

Method	Acc	Precision	Recall	F1
Chen et al.[8]	92.14	91.48	90.92	91.20
Li et al.[9]	93.07	92.35	91.86	92.10
Lee et al.[10]	93.58	92.96	92.41	92.68
Hao et al.[11]	94.12	93.45	92.97	93.21
Wang et al. [12]	94.46	93.88	93.35	93.61
Wang et al. [13]	94.83	94.22	93.74	93.98
Wang et al. [14]	95.11	94.57	94.08	94.32
Ours	96.38	95.91	95.47	95.69

As shown in Table 1, the proposed algorithm demonstrates superior overall recognition performance, achieving better results in accuracy, precision, recall, and F1 score. This indicates that the proposed method can more effectively characterize key anomaly features in the operational state of Kubernetes clusters, while maintaining good stability and consistency in the discrimination between normal and anomalous samples. Because the model can learn more comprehensive contextual relationships from multi-dimensional time-series information, its ability to identify complex anomaly patterns is further enhanced, demonstrating a high level of comprehensive detection and significant practical application value.

To further examine the contributions of key components in the proposed framework, we conducted ablation experiments, removing or simplifying several modules explicitly introduced in the method design. As shown in Table 2, the complete model consistently maintains the best overall performance, while the performance degradation observed under different ablation settings indicates that positional encoding, multi-head attention mechanisms, and temporal aggregation all play significant roles in the anomaly representation and decision formation of Kubernetes cluster monitoring data.

Table 2. Ablation study of the proposed method

Ablation setting	Acc	Precision	Recall	F1
w/o positional encoding	95.42	94.87	94.31	94.59
single-head attention	95.11	94.56	93.98	94.27
last-step representation	95.26	94.71	94.16	94.43
Full model	96.38	95.91	95.47	95.69

The complete model proposed in this paper performs well across all evaluation metrics, demonstrating the clear role of its key components. The introduction of positional encoding enhances the model's ability to characterize temporal sequence information, the multi-head attention mechanism more fully captures correlation features across different dimensions, and the discrimination method based on holistic temporal representation is more conducive to forming stable anomaly detection results. This indicates that the proposed method possesses strong comprehensive capabilities in multivariate temporal modeling, context dependency mining, and global state representation, thus achieving superior anomaly detection performance.

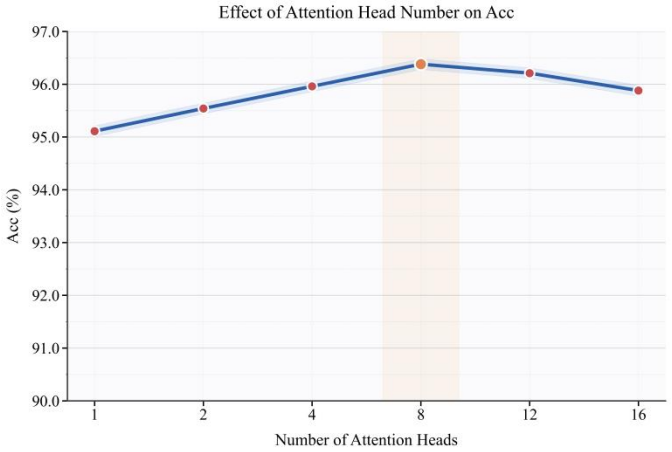

Figure 2. The effect of attention head number hyperparameter sensitivity on Acc

Figure 2 shows that the proposed algorithm maintains good recognition ability under different attention head settings, indicating that the method has strong adaptability and stability in multi-head attention modeling. A reasonable attention head configuration helps the model learn more fully the key correlation information in the multivariate temporal data of the Kubernetes cluster, thereby improving its ability to express complex abnormal states. Overall, the proposed method achieves a good balance between temporal dependency modeling, feature interaction characterization, and global anomaly detection, thus demonstrating high detection quality and good potential for practical applications.

These findings further align with recent backend reliability research that emphasizes dynamic operational reasoning, dependency-aware anomaly modeling, and evidence aggregation for cloud failure diagnosis. Risk-constrained repair planning for tool-augmented large language model agents suggests that anomaly detection outputs should support controlled operational intervention rather than isolated classification decisions [15]. Heterogeneous dynamic dependency graph learning highlights the importance of modeling changing backend relationships across services, nodes, and runtime states [16]. Temporal evidence

aggregation with causal-contrastive Transformer learning further shows that cloud failure diagnosis benefits from combining sequential evidence, causal discrimination, and robust representation learning [17].

5. Conclusion

This paper addresses the challenges of complex temporal dependencies, significant dynamic changes in runtime states, and high coupling of multi-source monitoring information in anomaly detection tasks within Kubernetes cluster environments. It presents a Transformer-based anomaly detection method. To overcome the shortcomings of traditional methods in long-distance temporal relationship modeling and global context characterization, a temporal detection framework suitable for Kubernetes cluster monitoring data is constructed. This framework achieves effective modeling of complex anomaly patterns by unifying the representation of multivariate runtime sequences, introducing temporal position information, and incorporating a multi-head attention mechanism. Related research demonstrates that this method can more fully uncover the implicit temporal correlations and structural anomaly features during cluster operation, thus providing more reliable technical support for intelligent anomaly identification in cloud-native scenarios.

From a research perspective, this work not only provides a highly expressive modeling approach for Kubernetes cluster anomaly detection but also further expands the application boundaries of Transformer in cloud-native operations and maintenance scenarios. With the increasing prevalence of containerized deployments, microservice architectures, and elastic resource scheduling, modern distributed systems are constantly improving in terms of observable data scale, system interaction complexity, and dynamic operating environments. This has transformed anomaly detection from a simple problem of identifying deviations in single metrics into a core task for understanding the state of complex systems and providing risk warnings. The advantages demonstrated by the proposed method in anomaly state representation, global context fusion, and cross-temporal dependency characterization make it highly valuable for application in cloud platform operations and maintenance, service stability assurance, resource scheduling optimization, fault warning, and intelligent operations and maintenance decision-making. It also has positive implications for improving the operational security and service reliability of large-scale cloud-native infrastructure.

Future research can be further expanded to address more complex cloud-native operating scenarios. On one hand, multimodal observation information such as logs, metrics, events, and call chains can be combined to build more comprehensive anomaly detection models, enhancing the understanding of heterogeneous anomaly patterns and cross-level anomaly propagation processes. On the other hand, for online detection, continuous learning, and real-time warning needs, more lightweight, efficient, and adaptive time-series modeling mechanisms can be explored to improve the long-term availability and deployment capabilities of models in dynamic environments. Furthermore, with the continuous development of edge cloud, hybrid cloud, and multi-cluster collaborative scenarios, anomaly detection models need to possess stronger transferability, robustness, and generalization capabilities. Continued progress in these areas is expected to further promote the deep integration of intelligent operation and maintenance technologies with cloud-native infrastructure, and provide crucial support for building a more stable, intelligent, and sustainable digital service system for related industries.

References

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

[2] Z. Li, J. Zhao, and J. Kang, "Multi-source anomaly detection for microservice systems," in Proceedings of the 2024 IEEE/ACM 46th International Conference on Software Engineering: Companion Proceedings, 2024, pp. 414-415.

[3] M. Panahandeh, A. Hamou-Lhadj, M. Hamdaqa, et al., "ServiceAnomaly: An anomaly detection approach in microservices using distributed traces and profiling metrics," Journal of Systems and Software, vol. 209, p. 111917, 2024.
[4] 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.
[5] C. Zhao, M. Ma, Z. Zhong, et al., "Robust multimodal failure detection for microservice systems," in Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, 2023, pp. 5639-5649.
[6] M. Allam, N. Boujnah, N. E. O'Connor, et al., "Synthetic time series for anomaly detection in cloud microservices," in International Conference on Machine Learning, Optimization, and Data Science, Cham: Springer Nature Switzerland, 2024, pp. 419-433.
[7] 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.
[8] Y. Chen, M. Yan, D. Yang, et al., "Deep attentive anomaly detection for microservice systems with multimodal time-series data," in Proceedings of the 2022 IEEE International Conference on Web Services, 2022, pp. 373-378.
[9] Y. Li, Y. Lu, J. Wang, et al., "Tadl: Fault localization with transformer-based anomaly detection for dynamic microservice systems," in Proceedings of the 2023 IEEE International Conference on Software Analysis, Evolution and Reengineering, 2023, pp. 718-722.
[10] C. Lee, T. Yang, Z. Chen, et al., "Eadro: An end-to-end troubleshooting framework for microservices on multi-source data," in Proceedings of the 2023 IEEE/ACM 45th International Conference on Software Engineering, 2023, pp. 1750-1762.
[11] J. Hao, P. Chen, J. Chen, et al., "Multi-task federated learning-based system anomaly detection and multi-classification for microservices architecture," Future Generation Computer Systems, vol. 159, pp. 77-90, 2024.
[12] P. Wang, X. Zhang, Z. Cao, et al., "MADMM: microservice system anomaly detection via multi-modal data and multi-feature extraction," Neural Computing and Applications, vol. 36, no. 25, pp. 15739-15757, 2024.
[13] S. Zhang, P. Jin, Z. Lin, et al., "Robust failure diagnosis of microservice system through multimodal data," IEEE Transactions on Services Computing, vol. 16, no. 6, pp. 3851-3864, 2023.
[14] 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.
[15] Y. Tang, "Time-Varying Operational Graph Reasoning and Risk-Constrained Repair Planning for Tool-Augmented Large Language Model Agents," Transactions on Computational and Scientific Methods, vol. 4, no. 6, 2024.
[16] F. Chen, "Heterogeneous Dynamic Dependency Graph Learning for Anomaly Detection in Large-Scale Backend Systems," 2024.
[17] Z. Hu, "Temporal Evidence Aggregation with Causal-Contrastive Transformer Learning for Intelligent Cloud Failure Diagnosis," 2024.