

Article

Not peer-reviewed version

Deep Learning Framework for Change-Point Detection in Cloud-Native Kubernetes Node Metrics Using Transformer Architecture

[Cancan Hua](#), [Ning Lyu](#), Chen Wang, Tingzhou Yuan *

Posted Date: 16 December 2025

doi: 10.20944/preprints202512.1278.v1

Keywords: Kubernetes node monitoring; multi-dimensional time series modeling; change point detection; and attention mechanism



Preprints.org is a free multidisciplinary platform providing preprint service that is dedicated to making early versions of research outputs permanently available and citable. Preprints posted at Preprints.org appear in Web of Science, Crossref, Google Scholar, Scilit, Europe PMC.

Copyright: This open access article is published under a [Creative Commons CC BY 4.0 license](#), which permit the free download, distribution, and reuse, provided that the author and preprint are cited in any reuse.

Disclaimer/Publisher's Note: The statements, opinions, and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions, or products referred to in the content.

Article

Deep Learning Framework for Change-Point Detection in Cloud-Native Kubernetes Node Metrics Using Transformer Architecture

Cancan Hua ¹, Ning Lyu ², Chen Wang ³ and Tingzhou Yuan ^{4,*}

¹ University of Southern California, Los Angeles, USA

² Carnegie Mellon University, Pittsburgh, USA

³ University of Missouri–Kansas City, Kansas City, USA

⁴ Boston University, Boston, USA

* Correspondence: tzyuan15@bu.edu

Abstract

This study proposes a Transformer-based change-point detection method for modeling and anomaly detection of multidimensional time-series metrics in Kubernetes nodes. The research first analyzes the complexity and dynamics of node operating states in cloud-native environments and points out the limitations of traditional single-threshold or statistical methods when dealing with high-dimensional and non-stationary data. To address this, an input representation mechanism combining linear embedding and positional encoding is designed to preserve both multidimensional metric features and temporal order information. In the modeling stage, a multi-head self-attention mechanism is introduced to effectively capture global dependencies and cross-dimensional interactions. This enhances the model's sensitivity to complex patterns and potential change points. In the output stage, a differentiated scoring function and a normalized smoothing method are applied to evaluate the time series step by step. A change-point decision function based on intensity scores is then constructed, which significantly improves the ability to identify abnormal state transitions. Through validation on large-scale distributed system metric data, the proposed method outperforms existing approaches in AUC, ACC, F1-Score, and Recall. It demonstrates higher accuracy, robustness, and stability. Overall, the framework not only extends attention-based time-series modeling at the theoretical level but also provides strong support for intelligent monitoring and resource optimization in cloud-native environments at the practical level.

Keywords: Kubernetes node monitoring; multi-dimensional time series modeling; change point detection; and attention mechanism

I. Introduction

In modern cloud-native architecture, Kubernetes has become the core platform for application deployment and resource scheduling [1,2]. With the increasing scale and complexity of containerized applications, the operational states and performance metrics at the node level show strong multidimensional and temporal characteristics. Metrics such as CPU utilization, memory consumption, disk I/O, and network throughput are not only correlated but also change dynamically over time. Detecting anomalies or abrupt shifts in such massive, multidimensional, and dynamic metric streams has become a key challenge for ensuring service reliability and system stability. Change-point detection, as a technique to identify turning points in system states, provides an effective means to understand performance evolution, prevent potential failures, and assist in elastic scaling [3].

In complex distributed environments, traditional monitoring and alert mechanisms often rely on static thresholds or single-dimensional metrics, which are insufficient for handling the dynamic and uncertain nature of microservice architectures across nodes and dimensions. The inherent properties of multidimensional time-series data mean that abnormal states usually emerge as complex joint variations, rather than isolated spikes in single metrics. Without effective change-point detection, systems may suffer from delayed recognition or misjudgments, leading to imbalanced resource scheduling, degraded application performance, or even service outages. Therefore, exploring advanced modeling approaches to capture subtle transitions in multidimensional time-series metrics has significant practical value [4].

In recent years, deep learning has opened new possibilities for time-series modeling [5]. Transformer architectures, especially those based on attention mechanisms, have shown strong advantages in capturing long-term dependencies and feature interactions [6–8]. Unlike traditional methods, Transformers do not depend on strict sequential assumptions and can model global correlations among metrics. This capability makes them well-suited for the complex and high-dimensional metric streams of Kubernetes nodes, and it helps improve both the accuracy and robustness of change-point detection.

In cloud-native application scenarios, change-point detection is not only about model innovation but also closely related to system operation and resource optimization. By accurately identifying state shifts at the node level, operators can locate potential risks at early stages and take preventive measures such as repair or scaling. Automated change-point detection also enables intelligent resource scheduling, keeping clusters stable and efficient under high load and sudden fluctuations. This improves system availability and service quality, while also bringing tangible benefits to enterprises in reducing operation costs, enhancing user experience, and strengthening competitiveness [9].

In summary, research on Transformer-based change-point detection for multidimensional time-series metrics in Kubernetes nodes addresses the real demand for efficient anomaly discovery in distributed systems, while also advancing the application of artificial intelligence in operations and resource management. This direction not only enhances system robustness and adaptability but also provides theoretical support and practical pathways for future intelligent operations. With the continuous expansion and increasing complexity of the cloud-native ecosystem, building frameworks that can accurately capture multidimensional temporal dynamics will play an increasingly important role in both academic research and industrial practice.

II. Related Work

Related work relevant to Transformer-based change-point detection on multidimensional metrics can be broadly divided into deep learning methods for change-point and time-series modeling, metric-driven performance prediction and representation learning, federated and privacy-aware anomaly detection, and advances in neural architectures and model design.

Deep learning approaches for change-point and sequence modeling have significantly improved the ability to capture complex temporal dynamics in multivariate data. Real-time change-point detection frameworks based on deep neural networks have been proposed to adaptively model multivariate time series and identify structural shifts through learned representations and data-driven decision rules, avoiding hand-crafted thresholds and simple statistical tests [10]. Beyond direct change-point modeling, methods that transform multidimensional time series into interpretable event sequences provide an alternative representation pipeline: continuous trajectories are converted into structured event streams, which can then be mined or classified more effectively by downstream models [11]. Recent work on improved sequence modeling architectures, such as Mamba-based models, further enhances the ability to discriminate subtle patterns in long sequences by refining the underlying state-space representation and sequence update mechanisms [12]. In the unsupervised detection setting, temporal contrastive representation learning has been applied to high-dimensional cloud environments, where contrastive objectives are designed along the temporal axis to emphasize

informative changes and suppress noise, leading to more robust anomaly detection without relying on explicit labels [13]. These developments highlight key methodological ideas—adaptive deep representations, representation transformation, advanced sequence architectures, and temporal contrastive objectives—that directly motivate the use of attention-based models and dedicated scoring functions for change-point detection in multidimensional Kubernetes node metrics.

Another line of research focuses on metric-driven performance prediction and representation learning for complex computing systems. AI-driven predictive modeling has been used to learn nonlinear mappings from system metrics to performance indicators, typically with deep architectures that integrate multiple metric dimensions and temporal dependencies into a unified model for system behavior prediction and resource management decisions [14]. Complementary approaches propose collaborative dual-branch contrastive learning for resource usage prediction, where two coupled branches learn representations under contrastive objectives to capture both global trends and local variations in metric streams [15]. From a methodological perspective, these studies show that combining multi-branch architectures, contrastive learning, and metric-conditioned modeling can significantly improve the expressiveness of learned representations for system behavior. The Transformer-based framework in this paper inherits this spirit by using multi-head self-attention and positional encoding to construct rich joint representations of multiple metrics over time, and by designing a tailored change-point scoring and smoothing mechanism rather than focusing solely on point prediction.

Federated learning has also been explored for anomaly and risk modeling in distributed environments. Federated learning frameworks have been proposed to build predictive models over sensitive data, where local models are trained on decentralized datasets and aggregated into a global model while respecting privacy constraints [16]. Differential federated learning further incorporates noise mechanisms and privacy-preserving protocols into the optimization process to improve robustness and security properties of AI systems under distributed training regimes [17]. Recent work combines federated optimization with contrastive learning objectives to enable behavioral anomaly detection across distributed systems, where local contrastive encoders are trained on behavioral traces and synchronized through federated updates, thus capturing global anomaly patterns without centralized raw data collection [18]. Although the present study adopts a centralized training paradigm on collected Kubernetes metrics, these federated and privacy-aware frameworks are methodologically relevant for future extensions of change-point detection to cross-cluster or cross-tenant settings, where decentralized metric storage and privacy constraints must be considered.

At the architectural level, there is a growing emphasis on automated design, structure-aware attention, and modular intelligent components. Research on the synergy between deep learning and neural architecture search demonstrates that automatically exploring architectural hyperparameters—such as depth, width, attention configurations, and connection patterns—can systematically discover architectures better aligned with specific tasks than manually crafted designs [19]. Structure-aware attention mechanisms integrated with auxiliary knowledge structures, such as knowledge graphs, have been used to inject structured relational information into attention computation, improving the interpretability and effectiveness of attention-based models [20]. In parallel, information-constrained retrieval frameworks built on large language model agents have been proposed, where agent-based pipelines orchestrate retrieval, reasoning, and decision-making under resource and information constraints [21]. These works collectively emphasize modular, adaptive, and structure-aware design patterns in modern AI systems. The method proposed in this paper follows a similar design philosophy: it introduces a Transformer-based architecture with linear embedding and positional encoding to preserve multidimensional metric and temporal information, employs multi-head self-attention to capture global and cross-dimensional dependencies, and defines a specialized scoring, normalization, and decision function tailored to change-point detection. In this way, the framework connects ideas from advanced sequence modeling, contrastive and representation learning, and modular architectural design to address the challenges of multidimensional change-point detection in Kubernetes node monitoring.

III. Proposed Approach

In terms of method design, we first need to represent the multi-dimensional time series metrics of Kubernetes nodes as a formalized input sequence. Let the observation vector of each node at time step t be $x_t \in \mathbb{R}^d$, where d represents the dimension of the metric. The entire time series can be represented as $X = \{x_1, x_2, \dots, x_T\}$, where T is the time length. Following the approach advocated by Zhou [22] for robust cross-domain optimization, we apply a linear embedding to each metric vector, enabling the model to map raw observations into a feature space that facilitates downstream learning. To further support temporal reasoning, we introduce positional encoding to each embedded vector, in line with the strategies used by Liu et al. [23] to enhance sequential pattern modeling in cloud-scale distributed systems. This combination preserves both metric diversity and temporal structure, laying a solid foundation for subsequent Transformer-based modeling. The input preprocessing can be formally defined as:

$$z_t = W_e x_t + p_t \quad (1)$$

Among them, $W_e \in \mathbb{R}^{d' \times d}$ is the learnable embedding matrix, p_t is the position encoding vector, and $z_t \in \mathbb{R}^{d'}$ is the embedded representation. This operation ensures that the model can capture both indicator features and time sequence information. The overall model architecture is shown in Figure 1.

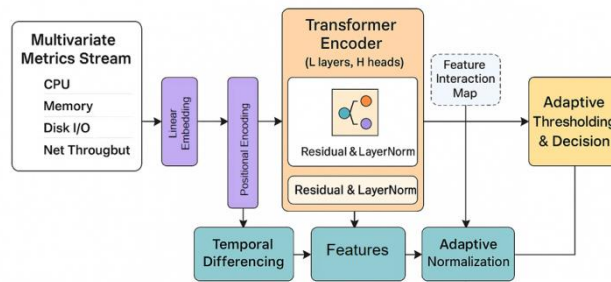


Figure 1. Overall model architecture.

Based on the embedding space, a multi-head self-attention mechanism is introduced to model the global dependencies between multi-dimensional time series. For any set of input vectors $Z \in \mathbb{R}^{T \times d'}$, the query, key, and value matrices are first obtained through linear transformation:

$$Q = ZW_Q, K = ZW_K, V = ZW_V \quad (2)$$

Among them, $W_Q, W_K, W_V \in \mathbb{R}^{d' \times d_k}$ is a trainable parameter. Then calculate the attention weight:

$$Attention(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V \quad (3)$$

This mechanism can capture complex dependencies in the time dimension and indicator dimension, and achieve multi-level feature aggregation through a multi-head mechanism.

To further improve the model's sensitivity to change points, a change point scoring function is introduced based on the representation of the Transformer encoder output. Let the encoder output be $H = \{h_1, h_2, \dots, h_T\}$, then the difference between adjacent time steps can be defined as:

$$s_t = \|h_t - h_{t-1}\|_2^2 \quad (4)$$

Where s_t represents the change point intensity score at time step t . When the score at certain moments is significantly higher than the background level, it can be considered that a change point may exist. To suppress the influence of noise, the sequence can be normalized and smoothed:

$$\tilde{s}_t = \frac{s_t - \mu}{\sigma} \quad (5)$$

Where μ and σ are the mean and standard deviation of the entire series, respectively.

Finally, to detect global change points, the smoothed score sequence is combined with the threshold mechanism. Let the detection threshold be θ , then the decision function can be expressed as:

$$y_t = \begin{cases} 1, & s_t \geq \theta \\ 0, & s_t < \theta \end{cases} \quad (6)$$

Here, $y_t = 1$ represents the change point detected at time step t , and $y_t = 0$ represents the normal state. This end-to-end Transformer-based framework enables the model to capture complex dependency patterns and key turning points in multi-dimensional time series metric streams, providing theoretical support and technical implementation for status monitoring and intelligent O&M of Kubernetes nodes.

IV. Performance Evaluation

A. Dataset

The dataset used in this study comes from the Google Cluster Trace. It is a publicly available large-scale dataset of distributed system operations that covers node-level and task-level metrics from real production environments. The dataset records resource usage of thousands of machines over a long time window, including CPU, memory, disk, and network indicators. It also contains detailed information on job scheduling, task execution, and node state changes. Its large scale and rich dimensions provide an authentic reflection of the dynamic features and complexity of nodes in cloud computing environments.

The dataset is characterized by its high dimensionality and long time-series nature, which creates sufficient scenarios for change-point detection tasks. Resource utilization at the node level shows significant non-stationarity and volatility over time. Different indicators across dimensions present potential correlations and coupling effects. This poses real challenges for models in capturing cross-dimensional dependencies and abrupt changes. At the same time, the dataset includes events such as node failures, resource anomalies, and scheduling fluctuations. These provide abundant samples for evaluating the effectiveness and robustness of change-point detection methods.

In the preprocessing stage, this study extracts core node-level metric sequences, including CPU utilization, memory usage, disk I/O, and network throughput. The raw records are processed with time alignment, normalization, and missing-value imputation. This transforms the data into standardized time-series inputs suitable for model training and inference. The procedure preserves the multidimensional dynamic features of the original data while ensuring effectiveness and stability in modeling. It provides a solid data foundation for change-point detection on multidimensional time-series metrics in Kubernetes nodes.

B. Experimental Results

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

Table 1. Comparative experimental results.

Method	AUC	ACC	F1-Score	Recall
Transformer [24]	0.921	0.894	0.876	0.861
GAT [25]	0.908	0.882	0.864	0.847
Mamba [26]	0.932	0.901	0.883	0.872
Swin-Transformer [27]	0.939	0.907	0.889	0.878
Ours	0.957	0.923	0.905	0.894

From the overall results, the proposed method outperforms all comparison models across all evaluation metrics, with clear advantages in AUC and ACC. This indicates that the Transformer-based change-point detection framework has stronger discriminative power in multidimensional time-series modeling. It can effectively capture complex dependencies and cross-dimensional dynamics, thereby improving accuracy and robustness in anomaly detection. Compared with traditional graph-based methods or simple sequence models, the proposed method demonstrates more stable and comprehensive performance. Further comparisons show that although GAT has certain advantages in modeling relationships between nodes, it is still insufficient in capturing long-range dependencies of multidimensional metrics. This limitation results in overall performance lower than other deep time-series models. Mamba and Swin-Transformer achieve relatively strong results in AUC and Recall, indicating their ability in feature extraction and sequence dependency modeling. However, they still show limitations in accuracy and balance, and fail to fully address information interactions in cross-dimensional time-series modeling. In conclusion, the proposed Transformer-based change-point detection framework effectively compensates for the shortcomings of existing methods in multidimensional time-series modeling. It not only achieves overall superiority across metrics but also excels in recall and discriminative ability, confirming its practical value in complex distributed system environments. These results show that the method can provide more intelligent and fine-grained support for cloud-native operations, thereby playing an important role in ensuring service continuity and efficient resource utilization.

This paper also presents a sensitivity experimental analysis of the change point score smoothing coefficient, and the experimental results are shown in Figure 2.

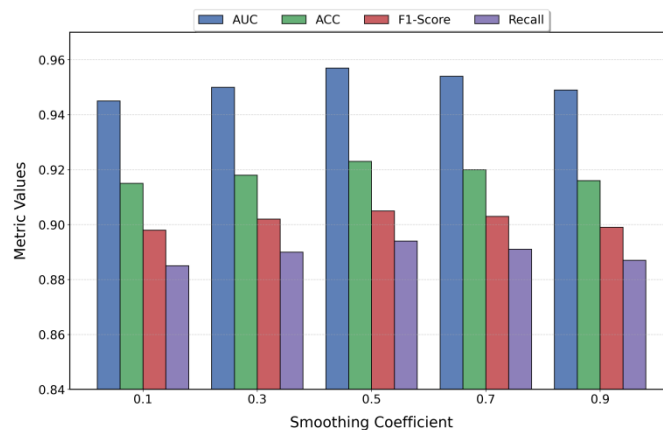


Figure 2. Sensitivity Experimental Analysis of Change Point Score Smoothing Coefficient.

As the smoothing coefficient varies, the model maintains high and stable AUC, ACC, F1-score, and Recall, indicating strong robustness to this parameter. AUC peaks at a smoothing coefficient of 0.5, while changes in either direction cause only minor performance fluctuations. F1-score and Recall remain consistently high, suggesting that smoothing effectively suppresses noise and enhances true change-point detection. Overall, a moderate smoothing level optimizes discrimination while preserving stability, which is crucial for reliable change-point detection on noisy Kubernetes node metrics.

V. Conclusion

This study focuses on the modeling and analysis of multidimensional time-series metrics in Kubernetes nodes and proposes a Transformer-based change-point detection method. Starting from the complexity and temporal characteristics of multidimensional metrics, the framework is designed with input embedding, global dependency modeling, and change-point scoring mechanisms. It effectively addresses the performance limitations of traditional methods under high-dimensional and non-stationary scenarios. By fully leveraging the attention mechanism and a differentiated scoring function during modeling, the method can more accurately capture sudden changes in node operating states and provide new insights for intelligent monitoring of distributed systems.

The experimental results show that the proposed method achieves excellent performance in key metrics such as AUC, ACC, F1-Score, and Recall. It demonstrates clear advantages in multidimensional time-series modeling and anomaly detection. The model not only achieves high accuracy but also exhibits strong robustness and stability. It maintains reliable performance under different parameter settings and environmental disturbances. These advantages enable the method to handle dynamic variations and complex fluctuations of resource usage in cloud-native environments, meeting the practical needs of efficient and stable detection mechanisms in operations.

The significance of this study lies not only in algorithmic improvement but also in its contribution to real-world applications. By enabling precise change-point detection at the node level in Kubernetes, operators can identify potential risks earlier and take timely scheduling and repair measures. This greatly reduces the probability of system failures and service interruptions. Such a mechanism directly contributes to improving cluster availability and service quality, optimizing resource allocation, and reducing operational costs. It also provides enterprises with strong technical support in a competitive environment.

Future research can proceed in several directions. One direction is to further explore multi-source metric fusion across clusters and system layers, allowing change-point detection to go beyond single nodes and capture the dynamic evolution of the entire cloud-native system. Another direction is to integrate the method with automated operations platforms, enabling seamless linkage between detection results, elastic scaling, and fault recovery strategies. With the continuous growth of cloud computing and containerized applications, the proposed method will show broader application potential in larger-scale and more complex scenarios.

References

1. H. Du and Z. Duan, "Finder: A novel approach of change point detection for multivariate time series," *Applied Intelligence*, vol. 52, no. 3, pp. 2496-2509, 2022.
2. Z. Duan, H. Du and Y. Zheng, "Trident: change point detection for multivariate time series via dual-level attention learning," *Proceedings of the Asian Conference on Intelligent Information and Database Systems*, Springer International Publishing, pp. 799-810, 2021.
3. G. Zerveas, S. Jayaraman, D. Patel et al., "A transformer-based framework for multivariate time series representation learning," *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*, pp. 2114-2124, 2021.
4. L. Deng, X. Chen, Y. Zhao et al., "HIFI: Anomaly detection for multivariate time series with high-order feature interactions," *Proceedings of the International Conference on Database Systems for Advanced Applications*, Springer International Publishing, pp. 641-649, 2021.
5. R. Liu, Y. Zhuang and R. Zhang, "Adaptive Human-Computer Interaction Strategies Through Reinforcement Learning in Complex," *arXiv preprint arXiv:2510.27058*, 2025.
6. P. Xue and Y. Yi, "Integrating Context Compression and Structural Representation in Large Language Models for Financial Text Generation," *Journal of Computer Technology and Software*, vol. 4, no. 9, 2025.
7. X. Song, Y. Liu, Y. Luan, J. Guo and X. Guo, "Controllable Abstraction in Summary Generation for Large Language Models via Prompt Engineering," *arXiv preprint arXiv:2510.15436*, 2025.

8. H. Feng, Y. Wang, R. Fang, A. Xie and Y. Wang, "Federated Risk Discrimination with Siamese Networks for Financial Transaction Anomaly Detection," 2025.
9. T. Tayeh, S. Aburakhia, R. Myers et al., "An attention-based ConvLSTM autoencoder with dynamic thresholding for unsupervised anomaly detection in multivariate time series," *Machine Learning and Knowledge Extraction*, vol. 4, no. 2, pp. 350-370, 2022.
10. M. Gupta, R. Wadhvani and A. Rasool, "Real-time change-point detection: A deep neural network-based adaptive approach for detecting changes in multivariate time series data," *Expert Systems with Applications*, vol. 209, Article 118260, 2022.
11. 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.
12. Z. Xu, J. Xia, Y. Yi, M. Chang and Z. Liu, "Discrimination of Financial Fraud in Transaction Data via Improved Mamba-Based Sequence Modeling," 2025.
13. S. Wang, "Temporal Contrastive Representation Learning for Unsupervised Anomaly Detection in High-Dimensional Cloud Environments," *Journal of Computer Technology and Software*, vol. 3, no. 5, 2024.
14. S. Han, "AI-Driven Predictive Modeling for System Performance and Resource Management in Microservice Architectures," *Journal of Computer Technology and Software*, vol. 4, no. 10, 2025.
15. G. Yao, "Collaborative Dual-Branch Contrastive Learning for Resource Usage Prediction in Microservice Systems," *Transactions on Computational and Scientific Methods*, vol. 4, no. 5, 2024.
16. R. Hao, W. C. Chang, J. Hu and M. Gao, "Federated Learning-Driven Health Risk Prediction on Electronic Health Records Under Privacy Constraints," 2025.
17. Y. Li, "Differential Privacy-Enhanced Federated Learning for Robust AI Systems," *Journal of Computer Technology and Software*, vol. 3, no. 4, 2024.
18. R. Meng, H. Wang, Y. Sun, Q. Wu, L. Lian and R. Zhang, "Behavioral Anomaly Detection in Distributed Systems via Federated Contrastive Learning," arXiv preprint arXiv:2506.19246, 2025.
19. X. Yan, J. Du, L. Wang, Y. Liang, J. Hu and B. Wang, "The Synergistic Role of Deep Learning and Neural Architecture Search in Advancing Artificial Intelligence", *Proceedings of the 2024 International Conference on Electronics and Devices, Computational Science (ICEDCS)*, pp. 452-456, Sep. 2024.
20. S. Lyu, M. Wang, H. Zhang, J. Zheng, J. Lin and X. Sun, "Integrating Structure-Aware Attention and Knowledge Graphs in Explainable Recommendation Systems," arXiv preprint arXiv:2510.10109, 2025.
21. J. Zheng, Y. Chen, Z. Zhou, C. Peng, H. Deng and S. Yin, "Information-Constrained Retrieval for Scientific Literature via Large Language Model Agents," 2025.
22. Y. Zhou, "Self-Supervised Transfer Learning with Shared Encoders for Cross-Domain Cloud Optimization," 2025.
23. H. Liu, Y. Kang and Y. Liu, "Privacy-Preserving and Communication-Efficient Federated Learning for Cloud-Scale Distributed Intelligence," 2025.
24. Y. Jeong, E. Yang, J. H. Ryu et al., "Anomalybert: Self-supervised transformer for time series anomaly detection using data degradation scheme," arXiv preprint arXiv:2305.04468, 2023.
25. Z. Liu, X. Huang, J. Zhang et al., "Multivariate time-series anomaly detection based on enhancing graph attention networks with topological analysis," *Proceedings of the 33rd ACM International Conference on Information and Knowledge Management*, pp. 1555-1564, 2024.
26. J. Liu, Q. Li, S. An et al., "EdgeConvFormer: Dynamic Graph CNN and Transformer based Anomaly Detection in Multivariate Time Series," arXiv preprint arXiv:2312.01729, 2023.
27. W. Zhang and C. Luo, "Decomposition-based multi-scale transformer framework for time series anomaly detection," *Neural Networks*, vol. 187, Article 107399, 2025.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.