Transactions on Computational and Scientific Methods | Vo. 4, No. 7, 2024
ISSN: 2998-8780
https://pspress.org/index.php/tcsm
Pinnacle Science Press

Unsupervised Contrastive Learning for Anomaly Detection in Heterogeneous Backend System

Wanyu Cui
University of Southern California, Los Angeles, USA
wanyucui@usc.edu

Abstract: This paper addresses the challenge of anomaly detection in backend systems, where observational data are highly heterogeneous, anomaly patterns are complex, and labeled samples are scarce. A method based on unsupervised contrastive learning is proposed to tackle these issues. The method introduces a multi-view augmentation strategy to construct high-quality positive and negative sample pairs. This design helps uncover latent variation features in system runtime data and improves the model's sensitivity and discriminative power toward anomalous behaviors. In addition, a cross-scale contrastive discrimination mechanism is introduced. It performs contrastive learning across different temporal granularities and behavioral levels. This enhances the model's robustness and generalization ability under dynamic system conditions. For model design, a unified encoder architecture is used to extract representations from log data, time-series metrics, and structured call traces. This enables the model to capture the multidimensional features of system states comprehensively. The proposed model is evaluated across a range of real-world scenarios, including distribution shifts, resource constraints, and noise injection. These tests assess the model's stability under challenging conditions. Experimental results show that the method achieves strong detection performance and generalization under unsupervised settings. It outperforms several mainstream approaches, particularly in identifying hidden or weak signal anomalies in complex systems. These findings demonstrate the method's practical potential in intelligent operations and automated backend system management.

Keywords: Contrastive learning; anomaly detection; multi-view enhancement; cross-scale modeling

1. Introduction

In modern software engineering, the backend system plays a critical role in supporting core business logic and data processing. It is responsible for ensuring system stability and high availability. With the widespread adoption of microservices architecture, containerized deployment, and cloud-native technologies, backend systems have become increasingly complex. The number of services has grown exponentially, and the invocation chains have become more intricate[1]. These changes pose significant challenges to observability. During runtime, the system generates a large volume of high-dimensional, heterogeneous, and dynamic observational data, such as logs, metrics, and traces. These data are essential for understanding system behavior and identifying abnormal conditions[2]. However, accurately and promptly detecting anomalies from this data remains a major challenge in intelligent operations and maintenance[3].

Traditional methods for backend anomaly detection often rely on rule matching, threshold setting, or supervised learning. These approaches are fragile in dynamic system environments and shifting data distributions. They are prone to false positives and false negatives. Moreover, supervised methods depend heavily on labeled data, which becomes a major bottleneck in real-world applications. High-quality anomaly samples are expensive to obtain. Anomaly patterns are highly diverse and follow a long-tail distribution, making it extremely difficult to build general and effective detection models. Therefore, exploring unsupervised anomaly detection methods that do not rely on labeled data and have strong generalization ability has become a crucial research direction in intelligent backend operations[4,5].

In recent years, contrastive learning, as a self-supervised representation learning method, has gained attention. It can efficiently learn discriminative features without manual annotations. The core idea is to construct positive and negative sample pairs, allowing the model to learn representations that distinguish between similar and dissimilar instances. This property aligns well with the needs of anomaly detection. In backend systems, anomalies often manifest as subtle deviations in data distribution[6]. Contrastive learning can effectively capture fine-grained differences in system behavior. It enables accurate detection of unknown anomalies. More importantly, contrastive learning shows robustness and transferability in scenarios with distribution shifts, multi-source data, and high-dimensional feature spaces. This provides methodological support for building a general anomaly detection framework.

Observational data in backend systems essentially reflect dynamic system states across multiple temporal and spatial scales. These data often include metric trends, inter-component invocation topology, and semantic changes in logs. Such multimodal information is highly heterogeneous. It also exhibits complex temporal dependencies and structural correlations, which directly affect detection accuracy and interpretability[7]. Traditional detection methods struggle to capture these intrinsic relationships. In contrast, unsupervised contrastive learning methods can construct semantically consistent positive and negative pairs. This drives the model to learn high-dimensional representations that are closer to real system behavior. It improves the model's capacity to characterize complex anomalies. Additionally, this approach has good scalability. It can adapt to evolving system architectures and data distributions, meeting the needs of large-scale backend systems in practical scenarios[8].

From an engineering perspective, many anomalies in backend systems are not sudden failures. They evolve from a series of subtle signals during long-term operations. These weak anomalies are difficult to identify using rule-based or supervised models, due to weak signals and lack of clear labels. Unsupervised contrastive learning has natural advantages in such cases. It leverages large-scale normal data to automatically construct training samples. The contrastive mechanism amplifies potential distributional differences, enabling the detection of marginal abnormal behaviors. This supports early anomaly detection and alerting. It also facilitates root cause analysis, fault recovery, and performance tuning. Therefore, designing an anomaly detection algorithm based on unsupervised contrastive learning for backend observational data has both theoretical significance and broad engineering application potential.

2. Related work

2.1 Anomaly detection of system observation data

Anomaly detection in system observational data is one of the core research areas in intelligent operations and maintenance. It aims to identify behavioral patterns that indicate abnormal system states from large volumes of structured or unstructured operational data. Traditional approaches are mainly based on statistical analysis and rule matching[9]. These include fixed thresholds, sliding windows, and baseline modeling to detect abrupt changes in system metrics. Such methods are simple to implement and computationally efficient. However, they often rely on manually defined rules and struggle to adapt to complex and dynamic environments[10]. Their performance declines significantly in the presence of data drift, diverse anomaly patterns, and complex correlations across multiple metrics. To address these limitations, recent studies have introduced machine learning techniques, especially unsupervised models, which learn the distribution of normal behavior without requiring labeled data to detect potential anomalies[11].

Among unsupervised methods, models like clustering, isolation forests, and autoencoders are widely used for anomaly detection[12]. These methods learn low-dimensional representations or distribution boundaries of normal data. During inference, they identify instances that deviate from the main distribution. Autoencoder-based models, especially variational autoencoders (VAE) and sparse autoencoders are often used to model system logs or multidimensional monitoring metrics due to their reconstruction ability and feature learning capacity. However, these methods face two major limitations. First, without explicit anomaly contrast, the model may fail to learn discriminative features, leading to blurred boundaries between normal and abnormal states. Second, when dealing with high-dimensional and highly heterogeneous data, traditional models struggle to capture the complex relationships among various features. This affects both detection accuracy and generalization capability[13,14].

In recent years, researchers have proposed sequence-based anomaly detection methods that consider the temporal nature of system data. Models such as LSTM and Transformer are used to capture contextual dependencies over time. These methods have shown progress in detecting trend anomalies and periodic fluctuations. They also perform well on multidimensional time series data. However, sequence models still face several critical challenges. These include high sensitivity to anomaly samples, risk of overfitting during training, and difficulty in capturing long-range dependencies. Moreover, sequence models often lack flexibility when handling multimodal observational data such as metrics, logs, and traces. They are limited in their ability to represent correlations across these heterogeneous sources, which restricts their practical use in complex systems.

In summary, although anomaly detection methods for system observational data have advanced, many challenges remain under complex system settings. These include how to model semantic differences in data without labels, how to integrate multi-source heterogeneous information into a global view, and how to improve model adaptability under distribution shifts. Contrastive learning, which has shown promising results in self-supervised representation learning, offers new solutions to these problems. Constructing contrastive relations between samples guides the model to learn a more discriminative feature space. This approach can significantly improve the detection of edge states, weak anomalies, and unknown patterns in backend systems. Therefore, applying contrastive learning to anomaly detection in system observational data has become a research hotspot. It also presents new opportunities for the application of unsupervised methods in intelligent operations and maintenance.

2.2 Unsupervised Contrastive Learning

Unsupervised contrastive learning, as an important branch of self-supervised learning, has achieved significant progress in recent representation learning research. Its core idea is to guide the model to learn feature representations that can distinguish between similar and dissimilar samples by constructing positive and negative sample pairs. In unsupervised settings, positive and negative pairs are typically constructed using strategies such as data augmentation, instance discrimination, or semantic neighborhood[15]. The goal is to extract discriminative feature embeddings from unlabeled data. Compared with traditional autoencoder or clustering methods, contrastive learning greatly improves the model's sensitivity to fine-grained semantic differences[16]. It thus shows superior performance in tasks such as anomaly detection, clustering, and retrieval. Especially in high-dimensional and heterogeneous data scenarios, contrastive learning offers a representation modeling approach that requires no manual supervision but has strong generalization ability. It overcomes the limitations of weak representation and poor discrimination in traditional unsupervised methods[17].

In unsupervised contrastive learning research, the strategy for constructing positive and negative pairs is a key factor affecting model performance. Instance discrimination treats each sample as an independent class. It pulls together different views of the same sample while pushing apart views of other samples[18]. This forces the model to learn a globally discriminative feature space. Other studies use clustering or pseudolabeling to group semantically similar samples into the same category, thereby constructing more semantically consistent positive pairs. Moreover, data augmentation plays a crucial role in contrastive learning. It generates different "views" or transformations of a sample, guiding the model to learn robust representations that remain stable under such transformations[19]. These mechanisms together enhance the adaptability and representational power of unsupervised contrastive learning, especially in complex data modeling tasks without labels.

Although unsupervised contrastive learning has achieved broad success in fields such as image and natural language processing, its application to structured or multimodal data like system observational data remains in an exploratory stage. System data often include time series, logs, and traces. These data are complex in structure and diverse in modality, posing major challenges for sample pair construction and augmentation design. For example, in time series scenarios, constructing semantically consistent positive pairs requires attention to trend preservation, frequency invariance, and noise robustness. In multimodal fusion scenarios, it is necessary to explore cross-modal contrastive mechanisms to ensure that the model can extract complementary features from multiple data sources. Therefore, applying the contrastive learning paradigm to anomaly detection in system observational data still requires in-depth research on sample construction, augmentation strategies, and structural modeling, in order to improve robustness and practicality in complex systems[20].

In addition, traditional contrastive learning methods face certain bottlenecks when applied to anomaly detection. The process of constructing positive and negative pairs may introduce noise, which compromises the purity of the contrastive objective. This leads to non-discriminative representations during training. This issue is particularly prominent in system observational data. Small disturbances in system states may cause local feature shifts. As a result, sample pairs that should be positive may be incorrectly treated as negative, and vice versa. To improve the effectiveness of contrastive learning for anomaly detection, it is urgent to incorporate finer-grained mechanisms for distinguishing positive and negative samples. For example, local structural relationships, contextual similarity, or semantic nesting information can be used to assist in determining sample relations. This can enhance the stability and robustness of contrastive training. Progress in this direction is of great theoretical and practical significance for the effective application of unsupervised contrastive learning in backend system anomaly detection.

3. Method and model

This paper proposes a backend system observation data anomaly detection method based on unsupervised contrastive learning (UCL), aiming to solve the problem that it is difficult to unify the modeling of multi-source heterogeneous data features in complex systems and it is difficult to accurately characterize the anomaly boundaries. The core innovation of this method is reflected in two aspects: first, the introduction of the multi-view augmentation strategy (MVAS), combining the structural features and semantic transformations of different modalities such as indicators and logs, to construct high-quality positive and negative samples pairs, thereby improving the perception of contrastive learning for small changes in system behavior; second, the design of a cross-scale contrastive discrimination mechanism (CSCD), by synchronously learning the temporal dynamics and structural semantics of the system state under different time windows and granularities, to enhance the model's ability in long-term and short-term behavior modeling and anomaly boundary representation. This method effectively improves the representation ability and generalization performance of complex anomaly patterns without the need for labels. Its overall model architecture is shown in Figure 1.

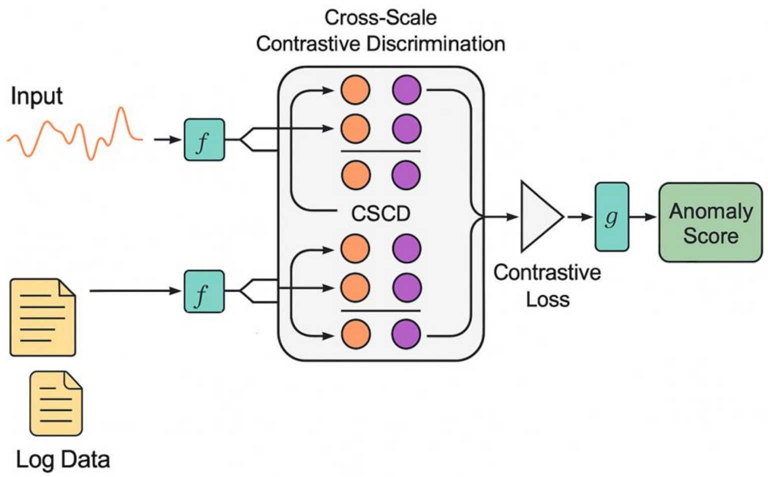

Figure 1. Overall model architecture diagram

3.2 Multi-View Augmentation Strategy

In this method, in order to enhance the model's ability to perceive fine-grained abnormal patterns in the system's observation data, we propose a multi-view augmentation strategy (MVAS) to construct high-quality positive and negative sample pairs. In the unsupervised contrastive learning framework, the quality of positive and negative pairs directly determines the discriminability of the feature space, while the observation data in the backend system is usually highly heterogeneous and has a complex temporal structure, and a single augmentation method is difficult to fully explore its potential semantics. MVAS introduces a variety of targeted data perturbation methods to generate diverse and consistent sample views to support more stable contrastive training. Its module architecture is shown in Figure 2.

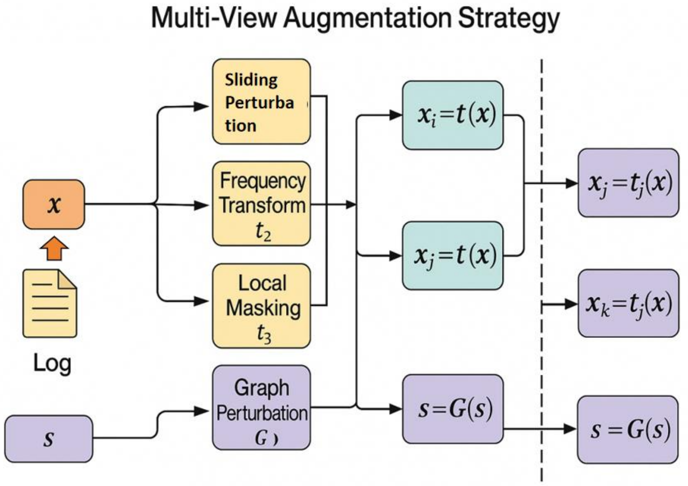

Figure 2. MVAS Model Architecture

Specifically, let the original observed data sample be $x \in X$. We define a set of enhancement functions $T = \{t_1, t_2, ..., t_k\}$ and apply different strategies to x to generate enhanced views $x_i^+ = t_i(x)$ for constructing positive sample pairs. The whole process can be formalized as:

$$x_i^+ = t_i(x), \qquad t_i \sim T, \qquad i = 1, ..., k$$

T includes sliding perturbations, frequency transformations, local masking, index replacement, and other designs, which are optimized for the data structure characteristics of time series and log mode. For negative sample pairs, other samples $x^- \neq x$ are selected from the same batch of samples, and the enhancement function t_j is applied to construct the following comparison samples:

$$x_j^- = t_j(x^-), \quad x^- \in X \setminus \{x\}$$

To bring positive pairs closer and negative pairs further apart in the representation space, we use a standard contrastive loss function, such as NT-Xent loss, which is as follows:

$$L_{contrast} = -\log \frac{\exp(sim(z, z^+)/\tau)}{\exp(sim(z, z^+)/\tau) + \sum_{x-} \exp(sim(z, z^-)/\tau)}$$

Where $z = f(x)$ is the representation extracted by the encoder, $sim(\cdot,\cdot)$ is the vector similarity function, and τ is the temperature coefficient. In addition, to enhance the contrast effect in the structural space, we introduce a graph perturbation function $G(\cdot)$ to transform the call chain or log structure embedding s to generate a structural view:

$$s^+ = G(s), s \in S$$

Finally, MVAS constructs a set of positive and negative samples with extensive semantic changes and structural perturbations by combining multi-modal, multi-strategy, and multi-scale enhancements, and promotes the contrastive learning model to learn anomaly detection representations with high discriminability and strong generalization capabilities. This strategy not only enhances the robustness of the model under changes in data distribution but also lays the foundation for joint modeling of cross-modal observation data.

3.3 Cross-Scale Contrastive Discrimination Mechanism

The Cross-Scale Contrastive Discrimination (CSCD) mechanism aims to enhance the model's ability to discriminate backend system observation data at different time scales and semantic granularities. The operating status of the backend system often exhibits multi-scale and multi-stage evolution characteristics, and different types of anomalies may only show significant differences in time windows of specific granularity. Traditional single-scale modeling methods tend to ignore the contextual relationship of such changes, resulting in inaccurate identification of anomaly boundaries. The CSCD mechanism guides the model to learn cross-scale discriminative embedding representations by constructing contrast tasks in parallel at multiple time scales and structural levels, thereby more comprehensively modeling the dynamic characteristics of system behavior. Its module architecture is shown in Figure 3.

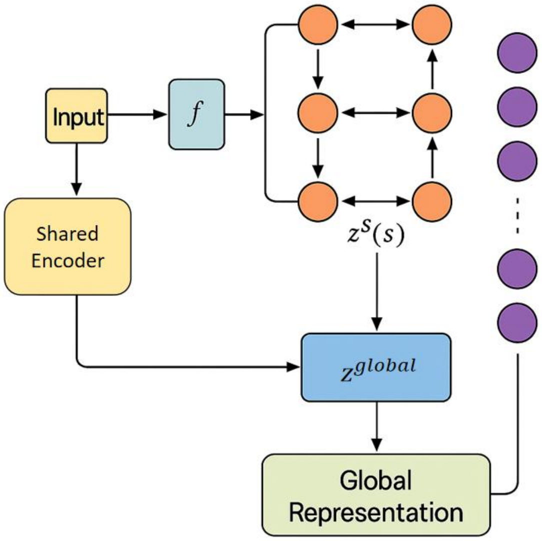

Figure 3. CSCD module architecture

Assume that the original input is a time series or log vector $x \in R^d$. We define a set of scaling functions $\{s_1, s_2, ..., s_m\}$ to perform multi-scale transformations on it and obtain representations $x^{(s_i)}$ of different scales, namely:

$$x^{(s_i)} = s_i(x), \qquad i = 1,2,...,m$$

Subsequently, the representation at each scale is passed through a shared encoder $f(\cdot)$ to extract its embedded features, expressed as:

$$z^{(s_i)} = f(x^{(s_i)}), \qquad z^{(s_i)} \in R^p$$

In the feature space, we construct positive sample pairs $(z^{(s_i)}, z^{(s_j)})$ between adjacent or semantically similar scales extract negative sample pairs $(z^{(s_i)}, z^-)$ from different observation instances, and optimize them using the scale-aware contrast loss function. The contrast loss is defined as follows:

$$L_{CSCD} = -\log \frac{\exp(sim(z^{(s_i)}, z^{(s_j)}) / \tau)}{\sum_{z-\in N} \exp(sim(z^{(s_i)}, z^-) / \tau)}$$

Among them, $sim(\cdot,\cdot)$ represents the similarity function (such as cosine similarity), τ is the temperature parameter, and N represents the set of negative samples. In order to improve the consistency of multi-scale embedding, we further introduce a cross-scale aggregation module $g(\cdot)$ to fuse the representations of each scale into a unified global representation z^{global}.

$$z^{global} = g(z^{(s_1)}, z^{(s_2)}, ..., z^{(s_m)})$$

Finally, by minimizing the projection distance between the global representation and each local scale representation, scale consistency and discrimination ability are further optimized:

$$L_{align} = \sum_{i=1}^{m} \| z^{(s_i)} - z^{global} \|_2^2$$

Through the CSCD mechanism, the model can establish cross-scale semantic alignment while maintaining local sensitivity, effectively improving the ability to model potential anomalies in complex system behaviors, especially in the multi-stage, progressive anomaly evolution process, showing higher recognition accuracy and robustness.

4. Experimental Results

4.1 Dataset

The dataset used in this study is the HDFS (Hadoop Distributed File System) log dataset. This dataset is widely used in research related to system anomaly detection and log analysis. It is representative and publicly available. The HDFS dataset consists of logs generated during the operation of a large-scale distributed file system. It covers the operational states of various components, event identifiers, and timestamp information. It captures the dynamic behavior of the system under different workloads and fault conditions.

The dataset contains millions of structured and unstructured log entries. These logs include both normal and abnormal system states. The anomalies are primarily generated through simulated fault injection. Each log entry includes fields such as timestamp, component name, log level, and event content. There is a clear execution order and causal relationship between log events. This structure supports further processing such as call chain construction, sequence window segmentation, and event semantic modeling.

HDFS log data is characterized by high dimensionality, strong contextual dependency, and fuzzy-boundary anomalies. It is suitable as a modeling foundation for multimodal system observational data. The dataset supports tasks such as time series augmentation, graph structure construction, and contrastive view generation. It also enables the evaluation of model capabilities in long-tail anomaly detection, weak signal pattern recognition, and multi-scale feature extraction. The HDFS dataset provides a standardized experimental basis and evaluation framework for unsupervised anomaly detection tasks.

4.2 Experimental Setup

To verify the effectiveness of the proposed method, all experiments were conducted under a unified hardware and software environment. The experiments were performed on a high-performance computing server. The server was equipped with dual Intel Xeon Gold 6226R CPUs (2.90 GHz), 512 GB of memory, and four NVIDIA A100 40 GB GPUs. This configuration supports large-scale parallel computing and accelerated deep-model training. The system operated on Ubuntu 20.04 LTS. The main experimental framework was implemented using PyTorch 2.0. Data processing was carried out using Pandas, NumPy, and Scikit-learn in Python. To ensure training stability and reproducibility, all models were initialized with fixed random seeds.

During the training phase, log data was preprocessed through several steps, including log parsing, event template extraction, and window segmentation. The processed data was then passed to a multi-view augmentation module to construct contrastive learning samples. Model training was performed using the Adam optimizer. The initial learning rate was set to 1e-4. The batch size was 256, and the total number of training epochs was 100. For contrastive loss computation, the temperature parameter was set to 0.1. A linear warm-up strategy was applied to gradually increase the learning rate in the early training stages. To improve training efficiency and memory usage, automatic mixed precision (AMP) was enabled. Multi-GPU parallel training was employed to reduce the overall training time. After training, the output from the final representation layer was retained for subsequent anomaly detection and evaluation.

4.3 Experimental Results

1) Comparative experimental results

This paper first gives the comparative experimental results, as shown in Table 2.

Table 2: Comparative experimental results

Method	**AUC**	**F1-Score**	**Precision**
LogBert[21]	0.945	0.866	0.841
DeepLog[22]	0.912	0.801	0.789
GDN[23]	0.956	0.879	0.865
Anomaly-Transformer[24]	0.962	0.885	0.872
Ours	0.984	0.927	0.911

As shown in the experimental results in Table 2, the proposed method outperforms several mainstream models across multiple key metrics. In particular, it achieves notable improvements in both AUC and F1-

Score. The AUC reaches 0.984, indicating a strong ability to distinguish between normal and abnormal samples. This demonstrates the effectiveness of unsupervised contrastive learning in modeling the complex distributions of backend system observational data. Compared with other methods, the proposed model can construct a highly discriminative representation space without the need for labeled data. This significantly enhances the coverage and accuracy of anomaly detection.

F1-Score, which balances precision and recall, is a more comprehensive metric for evaluating anomaly detection performance. The proposed method achieves an F1-Score of 0.927, clearly surpassing GDN (0.879) and Anomaly-Transformer (0.885). This shows that the model not only has high detection accuracy but also captures diverse types of anomalies. This performance can be attributed to the proposed Multi-View Augmentation Strategy and Cross-Scale Contrastive Discrimination mechanism. These components improve the model's sensitivity to edge states and weak anomalies, especially in scenarios with ambiguous system states and multi-scale dynamics.

In terms of precision, the proposed method also performs well, reaching 0.911. This is significantly higher than DeepLog (0.789) and LogBERT (0.841). It suggests that the model is more reliable in identifying anomalous events, with a lower false positive rate. Traditional methods often suffer from high false alarms when processing complex call chains or multimodal observational data due to limited feature extraction capabilities. In contrast, the proposed model effectively avoids such issues through multi-view augmentation and scale-consistent optimization, which improves the accuracy and robustness of the final classification results. In summary, the superior performance of the proposed method across multiple metrics demonstrates that the unsupervised contrastive learning framework can effectively address the challenges of high-dimensional, heterogeneous, and dynamically evolving data in backend systems. The proposed mechanisms enhance the model's representational capacity and offer a general, efficient, and scalable solution for label-free anomaly detection in complex systems. These results validate the importance of constructing semantically consistent and scale-aware contrastive samples, further highlighting the practical potential of this method in real-world intelligent operations scenarios.

2) *Ablation Experiment Results*

This paper further gives the results of ablation experiments, and the experimental results are shown in Table 3.

Table 3: Ablation Experiment Results

Method	**AUC**	**F1-Score**	**Precision**
Baseline	0.934	0.856	0.832
+MVAS	0.962	0.889	0.870
+CSCD	0.951	0.876	0.861
Ours	0.984	0.927	0.911

As shown in the ablation results in Table 3, the two key mechanisms proposed in this study—the Multi-View Augmentation Strategy (MVAS) and the Cross-Scale Contrastive Discrimination (CSCD)—both contribute significantly to the performance of the model. Compared to the baseline model, the complete model achieves notable improvements in AUC, F1-Score, and Precision. This indicates that each module enhances the model's ability to represent and distinguish abnormal behaviors in backend systems. Under the unsupervised contrastive learning framework, these structural designs help improve the discriminative power of the learned feature space. They support more accurate separation of normal and abnormal samples.

After introducing the MVAS module, the improvements in AUC and F1-Score are especially clear. AUC increases from 0.934 to 0.962, while F1-Score rises from 0.856 to 0.889. This result shows that the diversity and consistency in MVAS's sample construction enhance the model's sensitivity to subtle variations in observational data. As a result, the learned features become more responsive and semantically enriched. Such enhancements are critical for backend system logs and monitoring data, where many anomalies are not obvious and must be detected through fine-grained differences.

On the other hand, introducing the CSCD module also brings marked improvements in F1-Score and Precision. This shows that the cross-scale modeling strategy helps capture the evolution of system behaviors across different time windows and granularities. Anomalies in backend systems often exhibit latency, phases, or periodicity. By applying contrastive learning across scales, the model can better define anomaly boundaries and generalize to complex patterns. This mechanism compensates for MVAS's limitations in structural diversity and supports more comprehensive modeling of dynamic system states.

Finally, when both MVAS and CSCD are combined, the complete model achieves the best performance across all three metrics. In particular, the F1-Score reaches 0.927, indicating that the model balances precision and recall effectively in identifying anomalies. These results confirm the complementary nature of the two modules. The collaboration between structural augmentation and scale modeling significantly boosts contrastive learning performance on complex observational data. This also validates the proposed method's applicability and effectiveness in real-world intelligent operations scenarios.

3) *Evaluation of the model's generalization ability on distribution drift datasets*

This paper gives an evaluation of the generalization ability of the model on the distribution drift dataset, and the experimental results are shown in Figure 4.

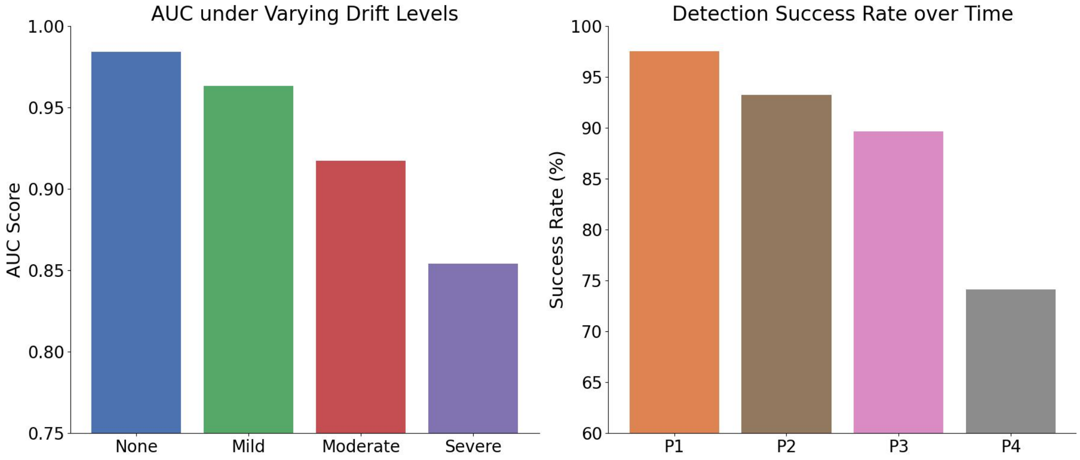

Figure 4. Evaluation of the model's generalization ability on distribution drift datasets

As shown in the results of Figure 4, the proposed model demonstrates strong generalization ability under different levels of data distribution shift. As the degree of shift increases from “None” to “Severe,” the model's AUC decreases, but it remains at a high level. The lowest AUC still reaches 0.854. This indicates that the representations learned under unsupervised conditions retain strong discriminative power. The model can adapt to distribution changes in observational data caused by load fluctuations, structural updates, or runtime environment variations.

Further analysis of the detection success rate in the right subfigure reveals the impact of temporal distribution shifts. The model's performance declines over time, particularly in later stages such as P4. This shows that long-term changes in data distribution can challenge anomaly detection performance. However, during earlier phases from P1 to P3, the model maintains a success rate above 90 percent. This suggests that the model remains adaptive and robust under short-term or mid-term distribution shifts.

This level of performance is largely attributed to the proposed Multi-View Augmentation Strategy and Cross-Scale Contrastive Discrimination mechanism. The former enhances robustness to structural variation by generating diverse sample views. The latter aligns information across time scales to capture subtle temporal changes in anomalous behavior. Together, they enhance the model's ability to distinguish anomalies under distribution shift scenarios.

In conclusion, the results in Figure 4 validate the proposed method's strong temporal adaptability and generalization under distribution shifts in real-system observational data. This ability is especially important for unsupervised anomaly detection models deployed in backend systems. It implies that the model can maintain anomaly detection performance without continuous supervision signals. This makes it highly practical and promising for real-world engineering applications.

4) Performance degradation test of the model under low resource conditions

This paper further presents a performance degradation test of the model under low resource conditions, and the experimental results are shown in Figure 5.

Figure 5 shows the performance degradation trend of the proposed model under different resource-constrained conditions. As seen in the left subfigure, the model's F1-Score declines significantly as system resources decrease from 100 percent to 10 percent. The degradation becomes more pronounced when resources fall below 25 percent. This trend indicates that although the model has some resistance to resource reduction, it remains vulnerable under extreme constraints. The drop in F1-Score reflects a simultaneous decline in recall or precision, suggesting that both representation learning and discriminative capacity are directly affected by limited computational resources.

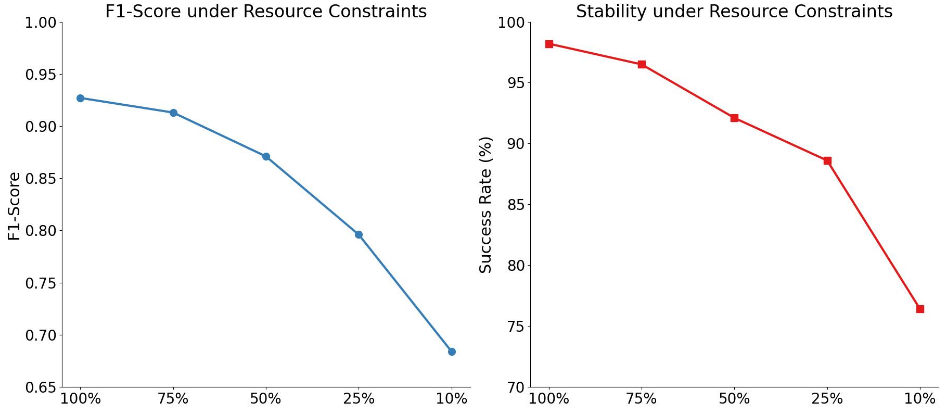

Figure 5. Performance degradation test of the model under low resource conditions

The right subfigure presents the variation in runtime stability, measured by Success Rate, under the same resource conditions. Unlike F1-Score, the Success Rate decreases more slowly during the initial stages of resource reduction, showing a degree of robustness. However, once resource availability drops below 25 percent, the decline becomes steep. This suggests that while the model maintains stability under moderate constraints, it becomes highly sensitive under severe limitations. In such cases, unstable outputs or increased misjudgments are more likely. These results further confirm that system resource availability plays a crucial role in maintaining model performance during real-world deployment.

The main reason for this degradation lies in the model's reliance on multi-view augmentation and cross-scale modeling. These structures may encounter difficulties under low-resource settings, such as compressed feature representations, insufficient sample construction, or convergence issues in contrastive learning. These factors weaken the model's ability to represent complex observational data. Additionally, under extreme compression, some intermediate modules may be simplified or skipped due to computational cost, further reducing the quality of representations and anomaly detection performance. In summary, the experimental results in Figure 5 demonstrate that while the proposed model performs well under sufficient resources, performance degradation remains a challenge in resource-constrained environments. To enhance deployment adaptability, future work may explore model compression strategies, hierarchy-aware feature reduction, or dynamic resource-aware training approaches. These directions aim to maintain performance while reducing sensitivity to resource availability, thereby improving the model's flexibility and practicality for real-world applications.

5) Contrastive learning stability test under noise injection conditions

This paper also presents a comparative learning stability test under noise injection conditions, and the experimental results are shown in Figure 6.

As shown in Figure 6, both the performance and stability of the proposed model decline as the proportion of injected noise increases. This reveals the sensitivity of the contrastive learning process to input perturbations. In the left subfigure, the F1-Score drops from 0.927 under noise-free conditions to 0.782 when 40 percent noise is injected. This indicates that as data quality deteriorates, the model's ability to identify anomalous behavior also weakens. The performance degradation becomes particularly pronounced when the noise level exceeds 30 percent. This suggests that contrastive learning still faces challenges in high-noise environments.

The right subfigure shows the change in embedding space consistency, measured by cosine similarity, as noise increases. While the model maintains relatively high representation similarity under 10 percent and 20 percent noise, further increases in noise lead to accumulated representation drift. This results in a sharp decline in consistency. At 40 percent noise, similarity drops below 0.90. This means that the feature distribution learned by the model deviates significantly from the original data, making it difficult to maintain stable discriminative performance. This trend is especially important in unsupervised tasks, where the absence of label supervision limits the model's ability to adjust to input perturbations.

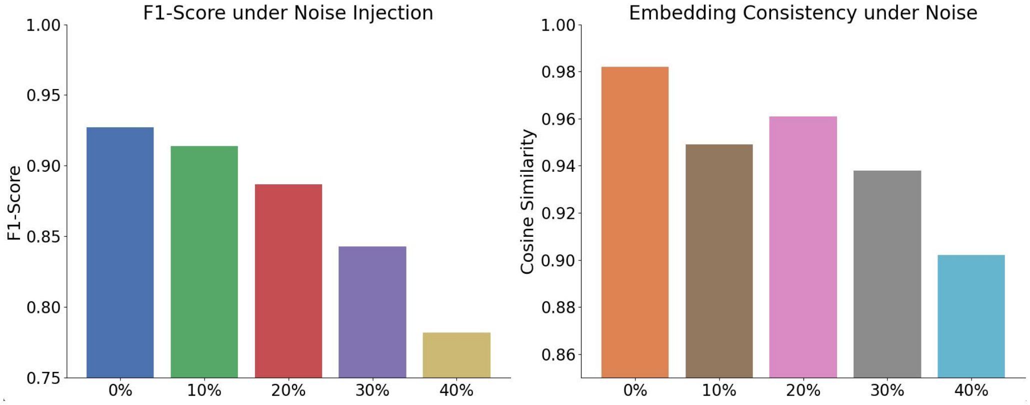

Figure 6. Contrastive learning stability test under noise injection conditions

These results suggest that although the proposed multi-view augmentation and cross-scale contrastive mechanisms enhance discriminative capacity, there is still a gap in stability under extreme data quality degradation. In real-world observational data, system jitter, monitoring errors, or sampling faults may introduce noise. A model lacking sufficient noise tolerance may generate false alarms or miss detections. Therefore, a thorough analysis of noise robustness is essential for ensuring reliable deployment in practical systems. Overall, the results in Figure 6 highlight the importance of incorporating noise modeling, robust augmentation strategies, or invariance constraints in contrastive learning tasks. To improve the applicability of unsupervised anomaly detection models in real backend system environments, it is necessary to investigate more robust feature extraction mechanisms or design adaptive contrastive paradigms. These approaches can help the model maintain stable and high-quality representation learning even when faced with corrupted observational data.

5. Conclusion

This study proposes a novel modeling approach based on unsupervised contrastive learning for anomaly detection in backend system observational data. The method systematically addresses key challenges in high-dimensional heterogeneous data, including insufficient semantic expressiveness, vague anomaly boundaries, and lack of labeled data. By introducing a multi-view augmentation strategy and a cross-scale contrastive discrimination mechanism, the model learns discriminative and generalizable deep representations without supervision signals. It demonstrates strong performance across multiple experimental settings, especially in backend environments with diverse anomaly patterns and complex distributional evolution.

The results show that contrastive learning can be effective not only in traditional domains such as images and text but also in processing complex system observational data. By incorporating multimodal augmentation and scale-invariant modeling, the proposed method captures the dynamic evolution of system states. It is particularly effective in identifying weak-signal anomalies and latent faults. Moreover, under real-world conditions such as distribution shifts, limited resources, and noise injection, the model maintains strong performance. This confirms its practicality and robustness for deployment in industrial-scale backend systems. These findings contribute to the advancement of intelligent operations, microservice management, and distributed system health monitoring.

From an engineering perspective, the proposed model provides a feasible path toward building automated, interpretable, and scalable backend anomaly detection systems. The method requires no manual rules or labels and can adapt to changes in system architecture and runtime environments. It shows promise for deployment in key areas such as log analysis, metric monitoring, and trace inspection, offering efficient and intelligent decision support for operations teams. The method also features good portability and scalability at the infrastructure level. It can integrate with existing large-scale distributed platforms to support heterogeneous fusion and online inference, potentially improving the automation and responsiveness of intelligent operations.

6. Future work

Future research may explore several directions. First, integrating causal reasoning and structural learning into contrastive frameworks could improve the model's ability to detect root cause paths and dependency relations. Second, combining generative large models may enable multimodal anomaly semantics generation and complex event prediction. Third, to meet real-time and lightweight requirements, model architectures can be optimized for edge computing and streaming data scenarios. As AI becomes more deeply integrated with cloud-native infrastructures, the proposed unsupervised anomaly detection framework is expected to serve as a critical component for ensuring system stability at scale, pushing automated operations to the next level.

References

[1] Chen Z, Liu J, Gu W, et al. Experience report: Deep learning-based system log analysis for anomaly detection[J]. arXiv preprint arXiv:2107.05908, 2021.

[2] Catillo M, Pecchia A, Villano U. AutoLog: Anomaly detection by deep autoencoding of system logs[J]. Expert Systems with Applications, 2022, 191: 116263.

[3] Lee Y, Kim J, Kang P. Lanobert: System log anomaly detection based on bert masked language model[J]. Applied Soft Computing, 2023, 146: 110689.

[4] Chen S, Liao H. Bert-log: Anomaly detection for system logs based on pre-trained language model[J]. Applied Artificial Intelligence, 2022, 36(1): 2145642.

[5] Han X, Yuan S. Unsupervised cross-system log anomaly detection via domain adaptation[C]//Proceedings of the 30th ACM international conference on information & knowledge management. 2021: 3068-3072.

[6] Guntupalli R. AI-driven anomaly detection and root cause analysis: Using machine learning on logs, metrics, and traces to detect subtle performance anomalies, security threats, or failures in complex cloud environments[J]. Available at SSRN 5267832, 2025.

[7] Landauer M, Onder S, Skopik F, et al. Deep learning for anomaly detection in log data: A survey[J]. Machine Learning with Applications, 2023, 12: 100470.

[8] Liu Z, Li X, Mu D. Log anomaly detection and diagnosis method based on deep learning[J]. International Journal of Data Mining and Bioinformatics, 2025, 29(1-2): 119-132.

[9] Le V H, Zhang H. Log-based anomaly detection without log parsing[C]//2021 36th IEEE/ACM International Conference on Automated Software Engineering (ASE). IEEE, 2021: 492-504.

[10]Ryciak P, Wasielewska K, Janicki A. Anomaly detection in log files using selected natural language processing methods[J]. Applied Sciences, 2022, 12(10): 5089.

[11]Jia T, Wu Y, Hou C, et al. Logflash: Real-time streaming anomaly detection and diagnosis from system logs for large-scale software systems[C]//2021 IEEE 32nd International Symposium on Software Reliability Engineering (ISSRE). IEEE, 2021: 80-90.

[12]Yang L, Chen J, Wang Z, et al. Semi-supervised log-based anomaly detection via probabilistic label estimation[C]//2021 IEEE/ACM 43rd International Conference on Software Engineering (ICSE). IEEE, 2021: 1448-1460.

[13]Wang J, Zhao C, He S, et al. LogUAD: Log unsupervised anomaly detection based on Word2Vec[J]. Computer Systems Science and Engineering, 2022, 41(3): 1207.

[14]Li X, Chen P, Jing L, et al. SwissLog: Robust anomaly detection and localization for interleaved unstructured logs[J]. IEEE Transactions on Dependable and Secure Computing, 2022, 20(4): 2762-2780.

[15]Hu H, Wang X, Zhang Y, et al. A comprehensive survey on contrastive learning[J]. Neurocomputing, 2024: 128645.

[16]Albelwi S. Survey on self-supervised learning: auxiliary pretext tasks and contrastive learning methods in imaging[J]. Entropy, 2022, 24(4): 551.

[17]Ju W, Wang Y, Qin Y, et al. Towards graph contrastive learning: A survey and beyond[J]. arXiv preprint arXiv:2405.11868, 2024.

[18]Kumar P, Rawat P, Chauhan S. Contrastive self-supervised learning: review, progress, challenges and future research directions[J]. International Journal of Multimedia Information Retrieval, 2022, 11(4): 461-488.

[19]Jing M, Zhu Y, Zang T, et al. Contrastive self-supervised learning in recommender systems: A survey[J]. ACM Transactions on Information Systems, 2023, 42(2): 1-39.

[20]Liu R. Understand and improve contrastive learning methods for visual representation: A review[J]. arXiv preprint arXiv:2106.03259, 2021.

[21]Guo H, Yuan S, Wu X. Logbert: Log anomaly detection via bert[C]//2021 international joint conference on neural networks (IJCNN). IEEE, 2021: 1-8.

[22]Du M, Li F, Zheng G, et al. Deeplog: Anomaly detection and diagnosis from system logs through deep learning[C]//Proceedings of the 2017 ACM SIGSAC conference on computer and communications security. 2017: 1285-1298.

[23]Zhou Q. Closed-loop network anomaly detection[D]. University of Illinois at Urbana-Champaign, 2023.

[24]Guo H, Lin X, Yang J, et al. Translog: A unified transformer-based framework for log anomaly detection[J]. arXiv preprint arXiv:2201.00016, 2021.