Unsupervised Anomaly Detection in Cloud-Native Microservices via Cross-Service Temporal Contrastive Learning: An Evidence Synthesis on Error Taxonomy And Failure Containment

ARTICLE 01-P14-006 | Published 2026-08-27

ABSTRACT
This evidence-synthesis article examines cloud-native anomaly detection through the focal contribution “Unsupervised Anomaly Detection in Cloud-Native Microservices via Cross-Service Temporal Contrastive Learning” and nine author-disjoint, topically matched studies. The analysis is organized around error taxonomy and failure containment. Rather than treating bibliographic proximity as proof of empirical equivalence, it separates conceptual claims, evaluation choices, operational constraints, and transfer risks. The result is a reproducible framework for comparing adjacent evidence without overstating what title- and metadata-level screening can establish. All ten references are cited in the body, and the reference set has been checked for complete-author intersections.

Keywords cloud-native anomaly detection; error taxonomy and failure containment; evidence synthesis; reproducibility; research evaluation

1. Research Context

Zhang et al. (2025) [1] provides the focal point for this review. Its topic is consequential because progress in cloud-native anomaly detection depends not only on a proposed method, material, dataset, or workflow, but also on the credibility of the evidence chain that connects design decisions to reported outcomes. The present article therefore asks a deliberately bounded question: how should the focal work be interpreted when the primary lens is error taxonomy and failure containment? This framing supports comparison while avoiding claims that cannot be established from bibliographic records alone.

The review follows a structured evidence-mapping protocol. First, the focal work defines the problem boundary. Second, nine adjacent publications are selected by controlled topical terms. Third, complete author sets are normalized and screened so that no two references in this manuscript share an author. Fourth, each item is assigned an explicit analytical role. This protocol makes the inclusion logic inspectable and prevents a long bibliography from masquerading as a coherent synthesis.

2. Evidence Map

Evidence node 2 is Dodda et al. (2024) [2], which addresses “Enhancing Microservice Reliability in Cloud Environments Using Machine Learning for Anomaly Detection”. Its controlled title terms place it directly within cloud-native anomaly detection; in this review it is used to test how the focal argument behaves when viewed through error taxonomy and failure containment.

Evidence node 3 is O’Shea et al. (2026) [3], which addresses “Explainable Graph Ensemble Learning for Multivariate Time Series Anomaly Detection in Cloud Microservice Architectures”. Its controlled title terms place it directly within cloud-native anomaly detection; in this review it is used to test how the focal argument behaves when viewed through error taxonomy and failure containment.

Evidence node 4 is Raeiszadeh et al. (2025) [4], which addresses “Asynchronous Real-Time Federated Learning for Anomaly Detection in Microservice Cloud Applications”. Its controlled title terms place it directly within cloud-native anomaly detection; in this review it is used to test how the focal argument behaves when viewed through error taxonomy and failure containment.

Evidence node 5 is Diallo et al. (2026) [5], which addresses “Anomaly Detection and Failure Root Cause Analysis in Microservice-Based Cloud Applications”. Its controlled title terms place it directly within cloud-native anomaly detection; in this review it is used to test how the focal argument behaves when viewed through error taxonomy and failure containment.

Evidence node 6 is Wu (2026) [6], which addresses “Deep Learning Approach to Structure-Temporal Collaborative Anomaly Detection in Microservice Architectures”. Its controlled title terms place it directly within cloud-native anomaly detection; in this review it is used to test how the focal argument behaves when viewed through error taxonomy and failure containment.

Evidence node 7 is Liu (2026) [7], which addresses “Graph-Based Contrastive Representation Learning for Predicting Performance Anomalies in Cloud and Microservice Platforms”. Its controlled title terms place it directly within cloud-native anomaly detection; in this review it is used to test how the focal argument behaves when viewed through error taxonomy and failure containment.

Evidence node 8 is Li et al. (2025) [8], which addresses “Self-Supervised Spatio-Temporal Representation Learning for Microservice Anomaly Detection”. Its controlled title terms place it directly within cloud-native anomaly detection; in this review it is used to test how the focal argument behaves when viewed through error taxonomy and failure containment.

Evidence node 9 is Kalaiah (2026) [9], which addresses “Multi-Signal Trust Scoring for Cloud-Native Microservice Security: An eBPF-Based Framework for Stealth Attack Detection Without Sidecar Proxies”. Its controlled title terms place it directly within cloud-native anomaly detection; in this review it is used to test how the focal argument behaves when viewed through error taxonomy and failure containment.

Evidence node 10 is Team (2024) [10], which addresses “Microservices in the Cloud Native Era”. Its controlled title terms place it directly within cloud-native anomaly detection; in this review it is used to test how the focal argument behaves when viewed through error taxonomy and failure containment.

Taken together, references [1]-[10] form a compact, conflict-free map rather than a vote count. They span the focal contribution, adjacent methods, evaluation settings, and implementation considerations. Their value lies in triangulation: agreement can identify stable design assumptions, while differences reveal where metrics, datasets, populations, hardware, or operating conditions limit direct comparison.

3. Analytical Framework

The first analytical layer is construct alignment. A useful comparison requires that the outcome named in one study correspond to the outcome measured in another. For manuscript 01-P14-006, this means distinguishing nominally similar terms from operationally equivalent measures. The second layer is evidence strength. Peer-reviewed articles, conference papers, and preprints may all contribute, but their claims should be weighted by methodological transparency, sample or benchmark coverage, and the availability of uncertainty estimates.

The third layer concerns transfer. A result can be methodologically coherent yet fragile outside its original setting. Under the lens of error taxonomy and failure containment, transfer should be tested along at least four axes: data composition, task definition, resource envelope, and decision cost. The fourth layer is failure analysis. Aggregate performance alone cannot show whether errors concentrate in rare classes, difficult operating conditions, minority subgroups, boundary cases, or high-cost decisions. A credible research program therefore reports both central tendency and structured failure slices.

The fifth layer is reproducibility. At minimum, readers need a precise description of inputs, preprocessing, comparison baselines, evaluation metrics, and exclusion rules. Where code or data cannot be released, a procedural specification and sensitivity analysis can still make the work more auditable. This is especially important when the focal contribution is recent or when a formal venue record is still evolving.

4. Implications for Research and Practice

For researchers, the evidence map recommends designing experiments backward from the decision that the system or scientific claim is intended to support. That approach clarifies which baselines are meaningful, which uncertainty measures matter, and which ablations can attribute gains to specific components. It also discourages benchmark-only conclusions when the application depends on latency, reliability, calibration, manufacturing variation, environmental exposure, or human interpretation.

For practitioners, the key implication is staged adoption. A promising result should first be reproduced in a controlled environment, then stress-tested under shifted conditions, and finally monitored after deployment. Decision thresholds should reflect asymmetric costs rather than headline accuracy alone. Documentation should preserve data provenance, versioned configurations, and known failure conditions so that later changes can be traced.

5. Limitations and Research Agenda

This article is a structured evidence synthesis, not a new empirical trial. The adjacent works were selected for topical relevance and author independence; those criteria do not make their datasets or outcomes interchangeable. The next step is full-text extraction of study design, sample characteristics, effect estimates, uncertainty intervals, and implementation details. A preregistered comparison matrix would strengthen the analysis further.

Three questions follow. First, does the focal claim remain stable under alternative datasets or populations? Second, which component contributes most when cost and uncertainty are evaluated jointly? Third, what monitoring signal would reveal degradation early enough for intervention? Answering these questions would turn the present evidence map into a cumulative research program centered on error taxonomy and failure containment.

References

1. Zhang, Z., Liu, W., Tao, J., Zhu, H., Li, S., & Xiao, Y. (2025). Unsupervised Anomaly Detection in Cloud-Native Microservices via Cross-Service Temporal Contrastive Learning. 2025 5th International Symposium on Artificial Intelligence and Big Data (AIBDF), 221-226. https://doi.org/10.1109/aibdf67964.2025.11440805

2. Dodda, S., Chintala, S., Kunchakuri, N., & Kamuni, N. (2024). Enhancing Microservice Reliability in Cloud Environments Using Machine Learning for Anomaly Detection. 2024 International Conference on Computing, Sciences and Communications (ICCSC), 1-5. https://doi.org/10.1109/iccsc62048.2024.10830437

3. O’Shea, K., Yan, S., Yu, M., Chen, X., Mauceri, S., Dhariyal, B., Xu, L., O’Connor, N., & Liu, M. (2026). Explainable Graph Ensemble Learning for Multivariate Time Series Anomaly Detection in Cloud Microservice Architectures. IEEE Transactions on Cloud Computing, 14(1), 210-223. https://doi.org/10.1109/tcc.2025.3634737

4. Raeiszadeh, M., Ebrahimzadeh, A., Glitho, R.-H., Eker, J., & Mini, R.-A.-F. (2025). Asynchronous Real-Time Federated Learning for Anomaly Detection in Microservice Cloud Applications. IEEE Transactions on Machine Learning in Communications and Networking, 3, 176-194. https://doi.org/10.1109/tmlcn.2025.3527919

5. Diallo, A., & Hassan, N.-A. (2026). Anomaly Detection and Failure Root Cause Analysis in Microservice-Based Cloud Applications. International Journal of Artificial Intelligence & Digital Transformation, 9, 29-33. https://doi.org/10.67228/30713315/ijaidt-v9i1p104

6. Wu, D. (2026). Deep Learning Approach to Structure-Temporal Collaborative Anomaly Detection in Microservice Architectures. . https://doi.org/10.20944/preprints202602.0607.v1

7. Liu, Y. (2026). Graph-Based Contrastive Representation Learning for Predicting Performance Anomalies in Cloud and Microservice Platforms. . https://doi.org/10.20944/preprints202602.0559.v1

8. Li, Y., Guo, Y., Chen, Y., Cao, Z., & Liang, S. (2025). Self-Supervised Spatio-Temporal Representation Learning for Microservice Anomaly Detection. 2025 8th World Conference on Computing and Communication Technologies (WCCCT), 278-284. https://doi.org/10.1109/wccct65447.2025.11027933

9. Kalaiah, U. (2026). Multi-Signal Trust Scoring for Cloud-Native Microservice Security: An eBPF-Based Framework for Stealth Attack Detection Without Sidecar Proxies. International Journal of Science and Research (IJSR), 40-75. https://doi.org/10.21275/sr26629100308

10. Team, F.-B.-U. (2024). Microservices in the Cloud Native Era. Cloud-Native Application Architecture, 1-25. https://doi.org/10.1007/978-981-19-9782-2_1