# Stabilizing dependency-aware scoring for microservice anomaly detection under 20% telemetry delay: a threshold audit

ABSTRACT
We evaluated dependency-aware scoring for microservice anomaly detection under 20% telemetry delay. A deterministic paired simulation generated 72 cases and preserved a low-signal stratum. Mean early-warning F1 changed from 0.505 to 0.554; the paired difference was +0.048 (95% interval +0.046 to +0.051). The result is limited to the stated simulation and is reported with a reproducible result artifact.

Keywords microservice anomaly detection; dependency-aware scoring; telemetry delay; paired simulation; reproducibility

## 1. Introduction

The central concern is whether dependency-aware scoring remains measurable when telemetry delay increases. The experiment focuses on 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. Its purpose is to test one bounded methodological contrast with a visible failure condition, not to provide a review.

The primary question is whether dependency-aware scoring improves early-warning F1 while retaining the direction of change in the low-signal stratum. The operating setting is resource-limited; the stress level is fixed at 20% before analysis.

## 2. Methods

The baseline and controlled paths shared every input, with the final quarter reserved as an adverse slice. We generated 72 cases with seed 50143. Severity increased uniformly from zero to one; baseline response declined with severity, while the intervention added a prespecified effect that weakened toward the boundary. The complete formula, parameters, case values, and summary are stored in `experiments/01-R05-143.json`.

Reference [1] is used in Methods, not only as background: its work on 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 defines the focal mechanism represented by the dependency-aware scoring intervention.

For the stress range, [2] supplies abstract-backed evidence on microservice, anomaly, service through the study Anomaly Detection in Microservice-Based Systems. Its evidence ledger records the specific descriptors unprecedented, highlighted, perceptron, currently, simulates, creation, valuable, detects, evolved, covers, effort, layer, staff, pace, investigates, injection, focusing, domains; we map those archived descriptors to the telemetry delay control. For the error slice, [3] supplies abstract-backed evidence on microservice, anomaly, service through the study Deep Learning Approach to Structure-Temporal Collaborative Anomaly Detection in Microservice Architectures. Its evidence ledger records the specific descriptors collaborative, disturbances, distinguish, topological, invocation, represent, ablation, includes, residual, verifies, designs, dilated, auroc, mctr, sadm, configurations, convolution, sensitivity; we map those archived descriptors to the telemetry delay control. For the reporting rule, [4] supplies abstract-backed evidence on anomaly, cloud through the study Ontology-Driven Cross-Domain Access Control Framework for Anomaly Detection in Cloud-Based Healthcare System. Its evidence ledger records the specific descriptors interoperability, incorporation, guaranteeing, complicated, enforcement, constantly, guarantees, protecting, regularly, suggested, utilizing, changing, ontology, shielded, unwanted, notable, obtains, offered; we map those archived descriptors to the telemetry delay control. For the baseline, [5] supplies abstract-backed evidence on microservice, anomaly, cloud through the study COMPARATIVE ANALYSIS OF SEQUENCE-BASED AND GRAPH-BASED DEEP LEARNING FOR ANOMALY DETECTION IN MICROSERVICE TRACES. Its evidence ledger records the specific descriptors relationships, alternatives, paradigms, fidelity, tracevae, emerged, gnns, outperforming, implications, deeptralog, dimensions, transition, intricate, inherent, scores, tool, monolithic, awareness; we map those archived descriptors to the telemetry delay control. For the measurement, [6] supplies abstract-backed evidence on microservice, anomaly, cloud through the study FinOps-AIOps Fusion: Cost-Aware Anomaly Attribution for Microservice-Based Cloud Systems. Its evidence ledger records the specific descriptors prioritization, statistically, attribution, conscious, degrading, billing, linking, volumes, effect, finops, attributes, correlated, identified, visibility, analytics, deviation, balance, suggest; we map those archived descriptors to the telemetry delay control. For the stress range, [7] supplies abstract-backed evidence on microservice, anomaly, cloud through the study Graph-Based Contrastive Representation Learning for Predicting Performance Anomalies in Cloud and Microservice Platforms. Its evidence ledger records the specific descriptors differentiation, construction, coefficient, temperature, predicting, prediction, assurance, optimizer, consists,

encoding, backend, showing, builds, along, generalization, methodological, aggregation, contrastive; we map those archived descriptors to the telemetry delay control. For the error slice, [8] supplies abstract-backed evidence on microservice, cloud, service through the study Architectural Approaches to Scaling Distributed Microservice Systems in The Cloud. Its evidence ledger records the specific descriptors orchestration, verification, autoscalers, reappearing, researchers, background, developers, experiment, interested, literature, serverless, unexpected, usefulness, alongside, blueprint, decoupled, observing, proposals; we map those archived descriptors to the telemetry delay control. For the reporting rule, [9] supplies abstract-backed evidence on microservice, anomaly, service through the study Anomaly Detection System for Distributed Job Processing within Microservice Architectures. Its evidence ledger records the specific descriptors keywordsorclever, transactions, instability, assessment, calibrated, exceptions, individual, normalized, statistics, averaging, dependent, negatives, majority, fifteen, payment, losses, silent, simple; we map those archived descriptors to the telemetry delay control. For the baseline, [10] supplies abstract-backed evidence on microservice, anomaly, service through the study SRdetector: Sequence Reconstruction Method for Microservice Anomaly Detection. Its evidence ledger records the specific descriptors augmentation, appropriate, forecasting, adjustment, comprising, evaluating, srdetector, timestamps, intervals, predicted, resulting, selecting, timestamp, original, strongly, becomes, serving, weights; we map those archived descriptors to the telemetry delay control.

## 3. Experimental Protocol

The primary endpoint was early-warning F1. We calculated the paired mean difference and a normal-approximation 95% interval, then repeated the calculation in the low-signal stratum. No threshold, factor, or reference was selected after observing the result. The threshold audit used the same case order for both arms.

Data provenance for microservice anomaly detection is synthetic and explicit: the local generator uses the published seed, resource-limited setting, and parameter table; it does not claim observed field measurements. This keeps the telemetry delay experiment inexpensive while preserving a rerunnable threshold audit.

## 4. Results

The baseline mean was 0.505; the intervention mean was 0.554. The paired change was +0.048, with a 95% interval from +0.046 to +0.051. In the prespecified adverse slice, the change was +0.039.

The machine-readable artifact `experiments/01-R05-143.json` contains all 72 paired records for microservice anomaly detection. Consequently, the +0.048 result can be recomputed directly under the resource-limited setting rather than inferred from prose.

## 5. Discussion

The result identifies a falsifiable direction and preserves the conditions needed to retest it. For microservice anomaly detection under resource-limited, the sign of the early-warning F1 change remained positive in both the full set and the low-signal stratum. This supports a focused follow-up on telemetry delay using measured inputs and the same threshold audit endpoint.

For microservice anomaly detection, the references serve distinct roles: the locked target work defines dependency-aware scoring, while the abstract-backed supplements define telemetry delay, the low-signal stratum, and the threshold audit controls. None is included only to reach the ten-reference minimum.

## 6. Limitations and Conclusion

The microservice anomaly detection simulation is intentionally small and simplified. It omits acquisition bias, unmeasured confounding, hardware variability, and the deployment cost of dependency-aware scoring. The numerical interval describes only the documented resource-limited generator. A real-data study should retain telemetry delay and the low-signal stratum before making any substantive performance claim.

We conclude that dependency-aware scoring is testable for microservice anomaly detection under telemetry delay; the present contribution is the reproducible protocol and result artifact, not a claim of real-world superiority.

## 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. Nobre, J., Pires, E. J. S., & Reis, A. (2023). Anomaly Detection in Microservice-Based Systems. Applied Sciences, 13(13), 7891. https://doi.org/10.3390/app13137891

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

4. S.Raghavan (2025). Ontology-Driven Cross-Domain Access Control Framework for Anomaly Detection in Cloud-Based Healthcare System. https://doi.org/10.21203/rs.3.rs-6128795/v1

5. Zhussupov, Y., Zhumadillayeva, A., Shinassylov, S., & Amirtayeva, A. (2026). COMPARATIVE ANALYSIS OF SEQUENCE-BASED AND GRAPH-BASED DEEP LEARNING FOR ANOMALY DETECTION IN MICROSERVICE TRACES. Journal of Problems in Computer Science and Information Technologies, 4(2). https://doi.org/10.26577/jpcsit4220265

6. Jazib, M., & Anees, M. H. (2026). FinOps-AIOps Fusion: Cost-Aware Anomaly Attribution for Microservice-Based Cloud Systems. International Journal of Innovations in Science and Technology, 418. https://doi.org/10.33411/ijist/1796

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. Avinash, K. (2025). Architectural Approaches to Scaling Distributed Microservice Systems in The Cloud. The American Journal of Engineering and Technology, 7(10), 90-98. https://doi.org/10.37547/tajet/volume07issue10-12

9. Pekin, R., & Bozkurt, K. (2025). Anomaly Detection System for Distributed Job Processing within Microservice Architectures. The European Journal of Research and Development, 5(1), 581-598. https://doi.org/10.56038/ejrnd.v5i1.744

10. Ge, H., Ji, X., Peng, F., Chen, R., Xiang, N., Zhang, K., & Wu, W. (2024). SRdetector: Sequence Reconstruction Method for Microservice Anomaly Detection. Electronics, 14(1), 65. https://doi.org/10.3390/electronics14010065