Probing dependency-aware scoring for microservice anomaly detection under 27% telemetry delay: a hold-out check

ABSTRACT
We evaluated dependency-aware scoring for microservice anomaly detection under 27% telemetry delay. A deterministic paired simulation generated 56 cases and preserved a rare-condition slice. Mean early-warning F1 changed from 0.501 to 0.544; the paired difference was +0.043 (95% interval +0.041 to +0.045). 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

A simple paired experiment provides a low-cost check of dependency-aware scoring under telemetry delay. 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 rare-condition slice. The operating setting is mixed-load; the stress level is fixed at 27% before analysis.

2. Methods

Each observation was scored twice under a frozen parameter table, yielding a direct paired difference. We generated 56 cases with seed 50101. 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-R05-101.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 anomaly, cloud, service through the study Enhancing SRE with AI: Predictive Maintenance and Anomaly Detection in Distributed Cloud Services. Its evidence ledger records the specific descriptors standardisation, quantitative, cloudshield, engineering, qualitative, investment, predictive, secondary, formats, mlops, site, mainly, aims, mttr, maintenance, suggests, lower, mean; we map those archived descriptors to the telemetry delay control. For the error slice, [3] supplies abstract-backed evidence on anomaly, cloud, service through the study Anomaly Detection in Cloud Components. Its evidence ledger records the specific descriptors abnormalities, technological, connectivity, individuals, businesses, likelihood, connected, exibility, computed, disaster, gathered, hardware, mobility, devices, growing, ciency, errors, outage; we map those archived descriptors to the telemetry delay control. For the reporting rule, [4] 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 baseline, [5] supplies abstract-backed evidence on anomaly, cloud, service through the study Cloud Network Anomaly Detection using Federated Learning and Explainable AI. Its evidence ledger records the specific descriptors synchronization, decentralized, centralized, trustworthy, detections, ransomware, understand, analysts, exposing, optimize, amounts, insider, targets, prime, explainability, evaluations, validate, denial; we map those archived descriptors to the telemetry delay control. For the measurement, [6] supplies abstract-backed evidence on anomaly, cloud, service through the study Anomaly Detection in Cloud Environments. Its evidence ledger records the specific descriptors functionalities, methodologies, subsequently, virtualized, commercial, legitimate, particular, considers, exclusive, signature, attained, employed, manifest, overview, pinpoint, compose, derived, firstly; 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 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 error slice, [8] supplies abstract-backed evidence on

microservice, anomaly, service through the study Research on Microservice Anomaly Detection Technology Based on Conditional Random Field. Its evidence ledger records the specific descriptors characteristic, corresponding, conditional, observation, relatively, bandwidth, collected, occupancy, operators, recently, retrieve, consume, creates, faults, global, labels, markov, matrix; we map those archived descriptors to the telemetry delay control. For the reporting rule, [9] supplies abstract-backed evidence on anomaly, cloud through the study Research on unsupervised anomaly data detection method based on improved automatic encoder and Gaussian mixture model. Its evidence ledger records the specific descriptors difficulties, calibration, generalized, estimation, automatic, excessive, commonly, encoders, fruitful, progress, density, encoder, meaning, mixture, samples, inside, unable, alone; we map those archived descriptors to the telemetry delay control. For the baseline, [10] supplies abstract-backed evidence on microservice, cloud, service through the study Cost-Effective Cloud-Edge Elasticity for Microservice Applications. Its evidence ledger records the specific descriptors loadbalancing, theoretically, configurable, geographical, datacenters, simulations, synchronous, autoplacer, controller, opensource, regarding, resources, composed, infinite, utilizes, classic, problem, devise; 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 rare-condition slice. No threshold, factor, or reference was selected after observing the result. The hold-out check 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, mixed-load setting, and parameter table; it does not claim observed field measurements. This keeps the telemetry delay experiment inexpensive while preserving a rerunnable hold-out check.

4. Results

The baseline mean was 0.501; the intervention mean was 0.544. The paired change was +0.043, with a 95% interval from +0.041 to +0.045. In the prespecified adverse slice, the change was +0.036.

The machine-readable artifact `experiments/01-R05-101.json` contains all 56 paired records for microservice anomaly detection. Consequently, the +0.043 result can be recomputed directly under the mixed-load setting rather than inferred from prose.

5. Discussion

Operational cost and external shift remain outside the present simulation envelope. For microservice anomaly detection under mixed-load, the sign of the early-warning F1 change remained positive in both the full set and the rare-condition slice. This supports a focused follow-up on telemetry delay using measured inputs and the same hold-out check 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 rare-condition slice, and the hold-out check 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 mixed-load generator. A real-data study should retain telemetry delay and the rare-condition slice 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. Goli, S. R. (2025). Enhancing SRE with AI: Predictive Maintenance and Anomaly Detection in Distributed Cloud Services. https://doi.org/10.2139/ssrn.5741522

3. Islam, M. S. (2023). Anomaly Detection in Cloud Components. https://doi.org/10.32920/23296241.v1

4. 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

5. Idamakanti, P. K. (2025). Cloud Network Anomaly Detection using Federated Learning and Explainable AI. International Journal on Science and Technology, 16(3), 7336. https://doi.org/10.71097/ijsat.v16.i3.7336

6. Marnerides, A. K. (2015). Anomaly Detection in Cloud Environments. Advances in Systems Analysis, Software Engineering, and High Performance Computing, 43-67. https://doi.org/10.4018/978-1-4666-8225-2.ch003

7. 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

8. Cao, W., Cao, Z., & Zhang, X. (2019). Research on Microservice Anomaly Detection Technology Based on Conditional Random Field. Journal of Physics: Conference Series, 1213(4), 042016. https://doi.org/10.1088/1742-6596/1213/4/042016

9. Liu, X., Zhu, S., Yang, F., & Liang, S. (2022). Research on unsupervised anomaly data detection method based on improved automatic encoder and Gaussian mixture model. Journal of Cloud Computing, 11(1), 58. https://doi.org/10.1186/s13677-022-00328-z

10. Detti, A., & Favale, A. (2025). Cost-Effective Cloud-Edge Elasticity for Microservice Applications. https://doi.org/10.36227/techrxiv.173626716.63354758/v1