Stress-testing dependency-aware scoring for microservice anomaly detection under 40% telemetry delay: a blocked comparison

ABSTRACT
We evaluated dependency-aware scoring for microservice anomaly detection under 40% telemetry delay. A deterministic paired simulation generated 64 cases and preserved a upper-severity quartile. Mean early-warning F1 changed from 0.519 to 0.563; 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

This study isolates one operational question in microservice anomaly detection: whether a transparent control survives noisy-input inputs. 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 upper-severity quartile. The operating setting is noisy-input; the stress level is fixed at 40% before analysis.

2. Methods

A fixed generator created matched inputs; only the prespecified intervention differed between arms. We generated 64 cases with seed 50066. 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-R05-066.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, cloud, service through the study Counterfactual Telemetry Simulation for Release-Aware Capacity Guardrails and Performance Risk Mitigation in Microservice Architectures. Its evidence ledger records the specific descriptors counterfactual, amplification, retrospective, alternative, prospective, conceptual, discussion, downstream, guardrails, protection, saturation, decisions, guardrail, plausible, applying, develops, metadata, versions; we map those archived descriptors to the telemetry delay control. For the error slice, [3] supplies abstract-backed evidence on anomaly, cloud through the study Private anomaly detection of student health conditions based on wearable sensors in mobile cloud computing. Its evidence ledger records the specific descriptors transmission, fortunately, recognition, infeasible, education, guarantee, monitored, recognize, becoming, managers, physique, probably, relative, students, wearable, impedes, massive, renders; 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 Unified Graph Modeling of Multi-Source Monitoring Data with Spatiotemporal Attention for Microservice Anomaly Detection. Its evidence ledger records the specific descriptors adaptively, modalities, formulate, encoded, thereby, edges, tight, msds, discriminative, spatiotemporal, outperforming, correlations, implications, attributes, baselines, dimension, instances, intricate; 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 Generative AI-Driven Anomaly Detection for Autonomous Cloud Infrastructure Management. Its evidence ledger records the specific descriptors opentelemetry, reinforcement, impractical, autonomous, openstack, processes, clusters, emerging, injected, predicts, severity, testbeds, request, alarms, region, flag, configuration, remediation; we map those archived descriptors to the telemetry delay control. For the measurement, [6] 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 stress range, [7] 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 error slice, [8] supplies abstract-backed evidence on microservice, cloud, service through the study Designing Resilient Microservice Architectures for High-Through Put PEO Systems in the Cloud. Its evidence ledger records the specific descriptors responsiveness, organization, professional, bottlenecks, limitations, balancing, designing, resilient, tolerance, breakers, detailed, employer, failover, circuit, clients, modular, payroll, retries; 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 Unified ensemble federated learning with cloud computing for online anomaly detection in energy-efficient wireless sensor networks. Its evidence ledger records the specific descriptors consumption, contributes, optimizing, conserve, wireless, avenues, created, fastest, lowest, online, opens, wsns, aggregation, beneficial, industrial, promising, advanced, reducing; 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 upper-severity quartile. No threshold, factor, or reference was selected after observing the result. The blocked comparison 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, noisy-input setting, and parameter table; it does not claim observed field measurements. This keeps the telemetry delay experiment inexpensive while preserving a rerunnable blocked comparison.

4. Results

The baseline mean was 0.519; the intervention mean was 0.563. The paired change was +0.043, with a 95% interval from +0.041 to +0.045. In the prespecified adverse slice, the change was +0.035.

The machine-readable artifact `experiments/01-R05-066.json` contains all 64 paired records for microservice anomaly detection. Consequently, the +0.043 result can be recomputed directly under the noisy-input setting rather than inferred from prose.

5. Discussion

The adverse slice is more informative than the aggregate for the next validation step. For microservice anomaly detection under noisy-input, the sign of the early-warning F1 change remained positive in both the full set and the upper-severity quartile. This supports a focused follow-up on telemetry delay using measured inputs and the same blocked comparison 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 upper-severity quartile, and the blocked comparison 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 noisy-input generator. A real-data study should retain telemetry delay and the upper-severity quartile 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. Mutyam, N. (2024). Counterfactual Telemetry Simulation for Release-Aware Capacity Guardrails and Performance Risk Mitigation in Microservice Architectures. American International Journal of Computer Science and Technology, 6, 109-119. https://doi.org/10.63282/3117-5481/aijcst-v6i6p111

3. Xie, Y., Zhang, K., Kou, H., & Mokarram, M. J. (2022). Private anomaly detection of student health conditions based on wearable sensors in mobile cloud computing. Journal of Cloud Computing, 11(1), 38. https://doi.org/10.1186/s13677-022-00300-x

4. Chen, X., Shi, G., & Cao, W. (2026). Unified Graph Modeling of Multi-Source Monitoring Data with Spatiotemporal Attention for Microservice Anomaly Detection. https://doi.org/10.2139/ssrn.6504540

5. Dr. Aaron Mitsuo Takeda (2025). Generative AI-Driven Anomaly Detection for Autonomous Cloud Infrastructure Management. Darpan International Research Analysis, 13(4), 78-85. https://doi.org/10.36676/dira.v13.i4.187

6. 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

7. 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

8. Chaudhari, S. (2025). Designing Resilient Microservice Architectures for High-Through Put PEO Systems in the Cloud. https://doi.org/10.2139/ssrn.5277296

9. Gayathri, S., & Surendran, D. (2024). Unified ensemble federated learning with cloud computing for online anomaly detection in energy-efficient wireless sensor networks. Journal of Cloud Computing, 13(1), 49. https://doi.org/10.1186/s13677-024-00595-y

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