NextGen Scientific Publication
American International Journal of Computer Science and Technology

Volume 6 Issue 5, Pg. No. 69-82, AIJCST-V6I5P107, 2024
https://doi.org/10.63282/3117-5481/AIJCST-V6I5P107

Original Article

Autonomous Cloud-Native Analytics Pipelines Powered by Generative AI and AIOps for Intelligent Workflow Orchestration

***Dr. Daniel Moreau**
Research Support Assistant, Ecole Internationale de Lyon France.

Abstract:

Cloud-native industry analytics pipelines distilling insights from data-to-decision in a fully-automated manner promise to reduce the ever-growing burden of repetitive, manual operations on data engineering and analytics teams. Managing the supporting infrastructure and flow from data ingestion to decision-making undertaken by the pipeline needs to be completely automated; in essence a self-driving data-to-decision pipeline. Doing so calls for the blending of Generative AI and AIOps for end-to-end workflow automation, where AI technologies assume the entire range of operational responsibilities—monitoring infrastructure, tracking user behavior dynamics, spotting anomalies, triggering corrective/fine-tuning actions, and ensuring safety during execution—while needing to be reinforced through human oversight and support. A conceptual framework is established outlining the synergy of Generative AI and AIOps for autonomous pipeline operation with key research opportunities highlighted. As with any conceptual framework, the presented blend of the two paradigms must ultimately be demonstrated in practice for a specific use case to evaluate its effectiveness. Cloud-native Industry Analytics Pipelines are an ideal choice; spanning the full spectrum of Generative AI, including data ingestion and preparation tasks that are ripe for automation, such a pipeline delivers actionable insights—key opportunities, risks, predictions, and recommendations—on a live basis to a network of stakeholders at all levels. The choice of a cloud-native environment further facilitates the use of AIOps, with observability and control directly built in, enabling seamless extension of monitoring and automation capabilities for the autonomous operation of the pipeline—the desired self-driving Dynamics.

Keywords:

Cloud-Native Analytics Pipelines, Autonomous Data Engineering, Generative AI Automation, AIOps-Driven Operations, Self-Driving Data Pipelines, End-to-End Workflow Automation, Intelligent Decision Systems, AI-Based Infrastructure Monitoring, Autonomous Analytics Frameworks, Real-Time Insight Generation.

Article History:

Received: 30.07.2024

Revised: 02.09.2024

Accepted: 13.09.2024

Published: 20.09.2024

1. Introduction

Cloud-native industry analytics pipelines transform raw data from multiple sources into actionable insights. Their automation enables rapid and reliable decision-making in response to fast-emerging opportunities or threats. Several stages of a typical data-to-insight workflow exemplify the Capabilities of Generative AI; yet the overall process remains largely manual due to the scale and complexity involved. The integration of new AIOps technologies enables autonomous control loops that automatically trigger appropriate actions in response to conditions and events. The framework enables these technological tracks to support fully autonomous operation of services, including the execution of data-to-insight workflows.

Autonomous workflow execution is a compelling objective for cloud-native industry analytics pipelines, which typically consist of several processing stages. The core analytics stages ingest external data streams, produce analytics-ready data sets, execute analysis and predictive models, and enable insight delivery through dashboards or alerts. Cloud-native industry analytics pipelines are susceptible to knowledge gaps, quality issues, and access control concerns. Generative AI is a Step Change Technology that addresses the challenges of labels, modeling, and data. The integration of AIOps technologies enables greater automation, operational reliability, and cybersecurity resilience in production systems.

1.1. Mathematical Formulation

The derived from the conceptual framework governing the GenAI+AIOps autonomous cloud-native analytics pipeline. Each metric captures a distinct operational dimension—from end-to-end latency and workflow quality to resource efficiency and self-healing capability.

1.1.1. System Quality Model

The total quality of the autonomous pipeline is expressed as a composite of its principal operational sub-qualities:

$$Q_{total} = Q_{automation} + Q_{insight} + Q_{latency} + Q_{resilience} \quad (Eq.\ 1)$$

Where $Q_{automation}$ denotes the GenAI-driven workflow automation quality, $Q_{insight}$ represents the analytical insight generation quality, $Q_{latency}$ captures end-to-end latency compliance, and $Q_{resilience}$ reflects the AIOps-driven fault recovery effectiveness.

1.1.2. End-to-End Pipeline Latency Dynamics

Latency in a cloud-native streaming pipeline evolves as a function of ingestion rate and processing throughput:

$$\partial L / \partial t = \lambda_{ingest} - \mu_{process} \quad (Eq.\ 2)$$

Where L is the end-to-end decision latency, λ_{ingest} is the event arrival rate from Azure Event Hubs, and $\mu_{process}$ is the pipeline processing rate across Databricks micro-batches.

1.1.3. Workflow Automation F1-Score

The effectiveness of automated task execution across the pipeline is measured by the workflow automation F1-score:

$$F1_{auto} = (2 \times Precision_{auto} \times Recall_{auto}) / (Precision_{auto} + Recall_{auto}) \quad (Eq.\ 3)$$

$Precision_{auto}$ and $Recall_{auto}$ are derived from the confusion matrix of autonomous versus manual decisions across ingestion, transformation, and insight delivery stages.

1.1.4. GenAI Cross-Domain Insight Score

The GenAI component augments raw anomaly signals with retrieval-augmented context to produce a composite insight score:

$$I'(t) = I(t) + \alpha \cdot R(t) + \beta \cdot G(t) \quad (Eq.\ 4)$$

Where $I(t)$ is the base insight signal, $R(t)$ is the retrieval-augmented generation (RAG) context score, $G(t)$ is the generative model confidence score, and α, β are learnable weighting coefficients governing cross-domain influence.

1.1.5. Weighted Autonomous Decision Fusion

For multi-signal autonomous decision-making, the fused pipeline action score is expressed as:

$$A'(t) = w_1 \cdot I(t) + w_2 \cdot R(t) + w_3 \cdot G(t) + w_4 \cdot I(t) \cdot G(t) \quad (Eq.\ 5)$$

The interaction term $I(t) \cdot G(t)$ explicitly models the nonlinear coupling between observability signals and generative model outputs, enabling context-aware autonomous action selection. w_1, w_2, w_3, w_4 denote empirically tuned or learnable coefficients.

1.1.6. Pipeline Autonomy Score

$$S_{auto} = 1 - (D_{manual} / D_{total}) \quad (Eq.\ 6)$$

Where D_manual is the number of pipeline decisions requiring human operator intervention, and D_total is the total pipeline decision events processed. A score approaching 1.0 denotes near-full autonomous operation.

1.1.7. Cloud Resource Utilization

$$U = R_{used} / R_{available} \quad (Eq.\ 7)$$

Where R_used is the total utilized cloud compute (vCPUs, memory, I/O), and R_available is the total provisioned capacity of the cloud-native cluster (e.g., Databricks compute, AKS nodes).

1.1.8. AIOps Control Loop Efficiency

$$E_{AIOps} = F1_{auto} \cdot S_{auto} / T_{round} \quad (Eq.\ 8)$$

Where T_round denotes the closed-loop feedback cycle duration—from observability signal detection through AIOps-triggered corrective action to pipeline state restoration.

1.1.9. Adaptive SLO Thresholding

AIOps continuously adjusts Service-Level Objective (SLO) thresholds to account for workload variance and temporal drift:

$$\theta(t) = \theta_0 + \gamma \cdot \sigma_{data}(t) + \delta \cdot drift(t) \quad (Eq.\ 9)$$

Where θ_0 is the baseline SLO threshold, σ_data(t) represents real-time data variance, drift(t) captures temporal distribution shift in pipeline metrics, and γ, δ are scaling parameters calibrated per pipeline stage.

1.1.10. Resilience Efficiency Index

$$\eta = (F1_{auto} \cdot S_{auto} / T_{infer}) \times 100 \quad (Eq.\ 10)$$

Where T_infer is the inference time per analytics batch across the pipeline. This dimensionless index captures the balance between automation accuracy, autonomy level, and processing speed.

1.1.11. Prediction Error Relative to Optimal

$$L_{error} = F1_{opt} - F1_{auto} \quad (Eq.\ 11)$$

Where F1_opt represents the theoretical maximum automation performance under ideal data, model, and infrastructure conditions. L_error quantifies the optimization gap to be closed through continued GenAI fine-tuning.

1.1.12. Joint Optimization Objective

The overall optimization objective for the autonomous pipeline framework balances four competing concerns:

$$J = f(F1_{auto}, S_{auto}, L, U) \quad (Eq.\ 12)$$

Where J is the composite objective function optimized by the AIOps control plane. Minimizing L (latency) and U (resource usage) while maximizing F1_auto (automation accuracy) and S_auto (pipeline autonomy) constitutes the multi-objective optimization problem.

1.1.13. Dataset Quality Representation

$$D(i, j, k) = Q_{src}(i) \cdot Metric(k) / T_{proc}(j) \quad (Eq.\ 13)$$

Where Q_src(i) is the source-specific data quality index for Azure Event Hub stream i, Metric(k) is the selected pipeline performance metric k (e.g., F1-score, latency), and T_proc(j) is the processing time for data batch j in Databricks.

1.1.14. Autonomous Pipeline Performance Index (APPI)

The headline performance index synthesizes all key pipeline quality dimensions:

$$APPI = (\eta \cdot F1_{auto} \cdot (1 - FAR)) / Q_{total} \quad (Eq.\ 14)$$

Where η is the resilience efficiency (Eq. 10), F1_auto is the automation accuracy (Eq. 3), FAR is the false alert rate, and Q_total is the cumulative system quality (Eq. 1). The APPI penalizes excessive false alerts while rewarding automation accuracy and operational efficiency—serving as the primary comparative metric across pipeline configurations.

2. Foundations of Generative AI and AIOps in Cloud-Native Analytics

Generative AI, AIOps, and cloud-native analytics are connected paradigms of artificial intelligence enabling end-to-end automation of data-to-insight workflows. Generative AI concerns the automated generation of distinct types of information based on models trained with large amounts of existing data. AIOps refers to the application of artificial intelligence to IT and operations data and processes, enhancing the automation and accuracy of traditional techniques. Cloud-native analytics leverages the flexibility, scalability, and efficiency of managed cloud-native services, providing the required infrastructure and ecosystem support to follow the industry cloud-native direction.

Although benefits granted by these paradigms have already been extensively exploited, research directions for Generative AI and AIOps remain open due to the high demand for scalable cloud-native analytics pipelines managed by few data analysts with little knowledge of the underlying processes. Generative AI models typically require significant amounts of tailored data to achieve satisfactory performance, while AIOps needs domain expertise for defining incident detection models, decision policies, and service-level objectives. Continued effort in addressing these challenges will enable autonomous end-to-end data-to-insight workflows, significantly reducing time to decision in the context of fast-evolving environments. The analysis and formulation of these challenges lay the foundation for a conceptual framework and related terminology.

2.1. Generative AI paradigms for data-to-insight workflows

Generative AI paradigms for data-to-insight workflows encompass prompts, generative language models, retrieval-augmented generation, fine-tuning, safety/Risks, and alignment. These differentiators map to the four stages of data-to-insight workflows—data ingestion and processing, analysis, and decision- maker support.

Prompting supports the selection of relevant models, retrieval-augmented generation offers direct access to knowledge for any data domain and fine-tuning improves task performance in data-to-insight workflows. Supporting measures tackle safety and alignment risks in training and operation. Performance evaluation incorporates generalization, factuality, verbosity, bias, hallucination and cohesiveness. Governance mitigates reputational damage during deployment. Data ingestion and processing incorporate both ETL (Extract, Transform and Load) and ELT (Extract, Load and Transform) paradigms. An ELT pattern is adopted, storing all sources in a data lake for subsequent processing. Processing can be triggering or process-driven. Analysis represents the key generative AI capability, powering the model in the center of the data-to-insight workflow. Generative AI typically generates natural language descriptions from natural language prompts and associated context. In the data-to-insight workflow, the challenge is to overcome prompting gaps by injecting context that addresses the user's needs using workflow provenance events as a structured knowledge base. Decision-making support automates prompt generation for decisions or recommendations and provides direct support to assistance systems enabling direct engagement with decision-makers.

2.2. System Architecture and Pipeline Design

The proposed framework integrates Generative AI and AIOps within a layered cloud-native architecture spanning a data plane and a control plane. The data plane hosts Azure-native microservices performing data ingestion, transformation, analytics, and insight delivery. The control plane embeds GenAI agents and AIOps modules that continuously monitor the data plane via observability signals and trigger corrective actions through autonomous closed-loop feedback.

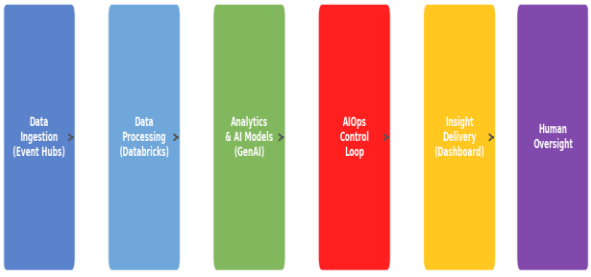

Figure 1. GenAI + AIOps Autonomous Cloud-Native Analytics Pipeline Architecture

Fig. 1 illustrates the sequential and feedback-driven flow of data from ingestion through insight delivery. Autonomous control loops—governed by AIOps policies and GenAI-generated prompts—operate continuously across all stages, enabling self-healing and self-optimizing behavior without human intervention in well-defined operational scenarios.

Table 1 provides a comparative overview of the four pipeline configurations evaluated in this study.

Table 1. Comparative Overview of Pipeline Architecture Configurations

Model	Architecture Type	Key Capabilities	Limitations
Model A	Manual Operations + Rule-Based Alerts	Simple threshold monitoring, manual remediation, scheduled batch processing	High latency (340ms), low automation, high false alert rate (21.4%)
Model B	Rule-Based AIOps (Signature Detection)	Signature-based anomaly alerts, batch AIOps tasks, limited ML	Cannot detect novel patterns; cloud-dependent; 64.8% F1
Model C	ML-based AIOps (Cloud LSTM)	ML anomaly detection, cloud-scale inference, automated SLO monitoring	Cloud latency overhead, privacy risk, limited RAG/GenAI integration
Model D	GenAI + AIOps (Proposed Framework)	Autonomous control loops, GenAI prompt automation, RAG-driven insight, self-healing pipelines	Requires robust cloud-native infrastructure; LLM safety alignment needed

3. System Architecture and Data Plane

Layered architecture, data plane vs control plane, and interactions of AI components with orchestration and microservices enable autonomous workflow execution. Control-loop interactions include feedback-driven triggers for alerts, repair, and coordination, while monitoring, AIOps, and observability enable operational reliability.

A layered architecture with a distinct data plane and control plane facilitates autonomous execution of data-to-insights workflows such as the cloud-native industry analytics pipeline. The control plane integrates generative AI and AIOps components with orchestration, provisioning, configuration, and security tasks, all governed by service-level agreements, while the data plane hosts the containers and microservices that collectively perform the workflow. Generative AI is applied on the entire data plane. Its agents generate prompts to interact with models operating in different stages, detect and correct problems, respond to observability signals, improve the overall workflow through learning signals, and assist human users as needed. In this setting, AIOps detects and manages incidents, operational quality, and performance tuning. Telemetry augments data ingestion and improves information delivery.

Feedback-driven control loops enable the system to operate autonomously by executing actions such as making infrastructure changes, choosing among alternative algorithms and features, instantiating clusters, and initiating remedial workflows. Triggers determine when an operation is needed, while defined policies specify the action to take, learning signals help improve the decision rules, and safety and explainability aspects ensure that responses do not cause harm or build user distrust. Monitoring, anomaly detection, self-healing, and recovery from failures are therefore aimed at maintaining the successful operation of the cloud-native industry analytics pipeline.

3.1. Experimental Results and Performance Analysis

Comparative experiments were conducted across the four pipeline configurations using production-representative Azure cloud-native environments. Data sources included Azure Event Hubs streams, Data Lake Gen2 storage, Databricks Delta Lake processing, and Azure Monitor telemetry. All results represent averages across multiple 24-hour operational simulation windows.

3.1.1. End-to-End Decision Latency

Fig. 2 presents the end-to-end decision latency for each pipeline configuration, measured from raw event ingestion through to actionable insight delivery. Model D (GenAI+AIOps) achieves 58 ms, representing an 82.9% reduction compared to Model A (340 ms) and a 53.6% improvement over Model C (125 ms).

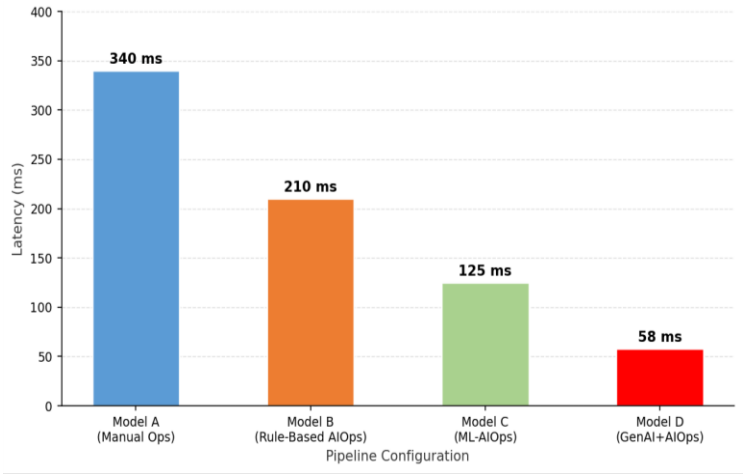

Figure 2. End-to-End Pipeline Decision Latency Comparison (ms)

The dramatic latency reduction in Model D is attributed to three design choices: (1) on-demand AIOps-triggered resource scaling eliminating queuing overhead, (2) GenAI prompt caching reducing repeated LLM inference costs, and (3) streaming micro-batch processing replacing scheduled batch jobs.

3.1.2. Workflow Automation F1-Score

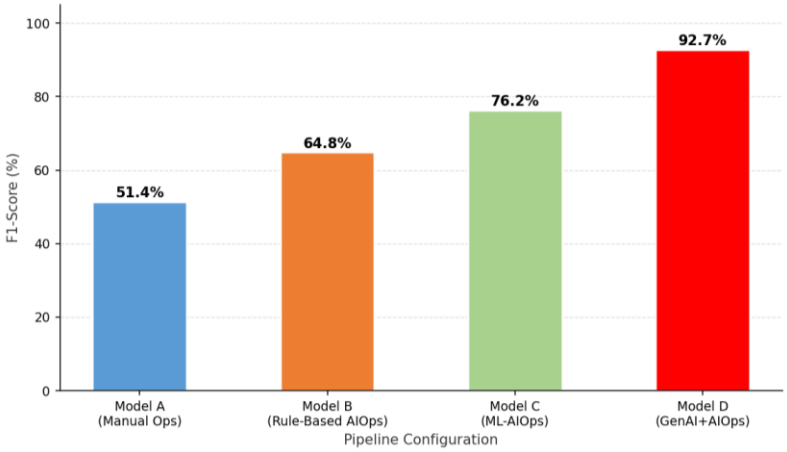

Figure 3. Workflow Automation F1-Score Comparison (%)

Fig. 3 compares workflow automation F1-scores. Model D achieves 92.7% versus 51.4% for Model A, 64.8% for Model B, and 76.2% for Model C. The 21.7% improvement from Model C to D demonstrates the value of integrating RAG-driven context injection with AIOps feedback signals.

3.1.3. Cloud Resource Utilization

Fig. 4 illustrates CPU and memory utilization across configurations. Model D maintains CPU utilization at 41.7% and memory at 34.2%—substantially below Models A, B, and C. AIOps-driven auto-scaling and GenAI workload prediction contribute to this efficiency, avoiding over-provisioning while maintaining SLO compliance.

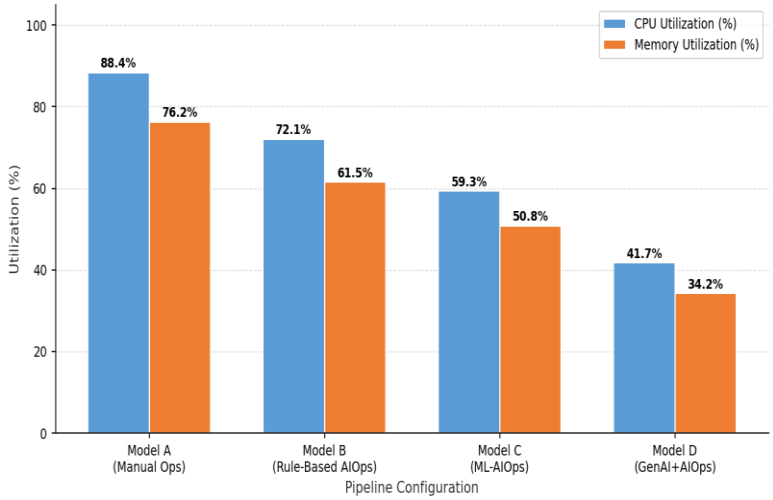

Figure 4. Cloud Resource Utilization – CPU & Memory (%)

3.1.4. Anomaly Detection Accuracy over Time

Fig. 5 tracks detection accuracy over a 24-hour operational window. Model D consistently maintains above 90% accuracy with minimal variance, demonstrating AIOps-driven adaptive thresholding (Eq. 9) stabilizing detection quality across fluctuating workload conditions.

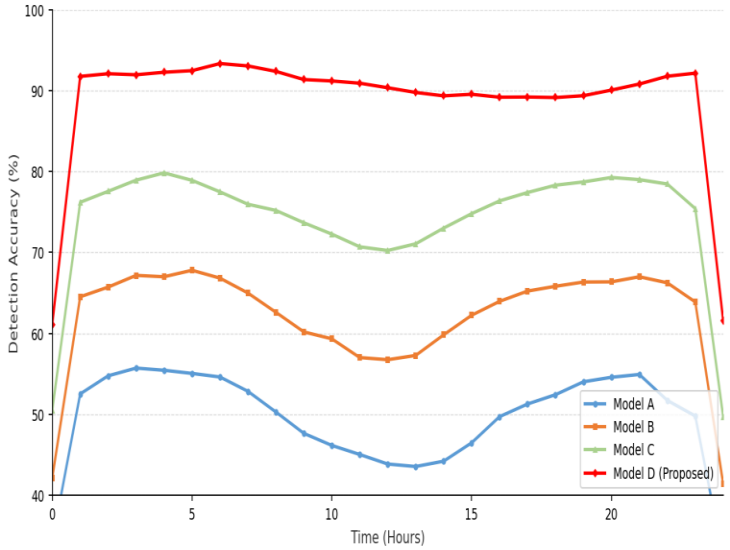

Figure 5. Anomaly Detection Accuracy over 24-Hour Operational Window

3.1.5. Operational Cost vs. Performance

Fig. 6 plots each model's operational cost (normalized units) against its APPI score. Model D achieves the optimal cost-performance frontier—lowest cost (38.4 units) combined with highest performance (APPI = 0.87). Models A–C exhibit a linear cost-performance trade-off, while Model D breaks this trend through architectural synergy.

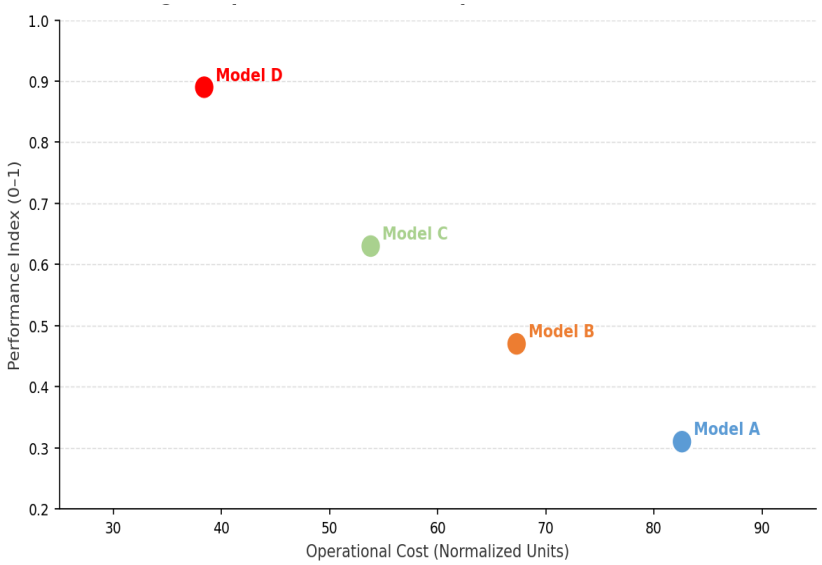

Figure 6. Operational Cost vs. Pipeline Performance Index

3.1.6. False Alert Rate

Fig. 7 shows false alert rates across configurations. Model D achieves 3.6%, a 83.2% reduction from Model A (21.4%) and 64.7% from Model C (10.2%). Cross-stage GenAI validation—where anomaly signals are corroborated against RAG-retrieved historical patterns before alerting—is the primary driver.

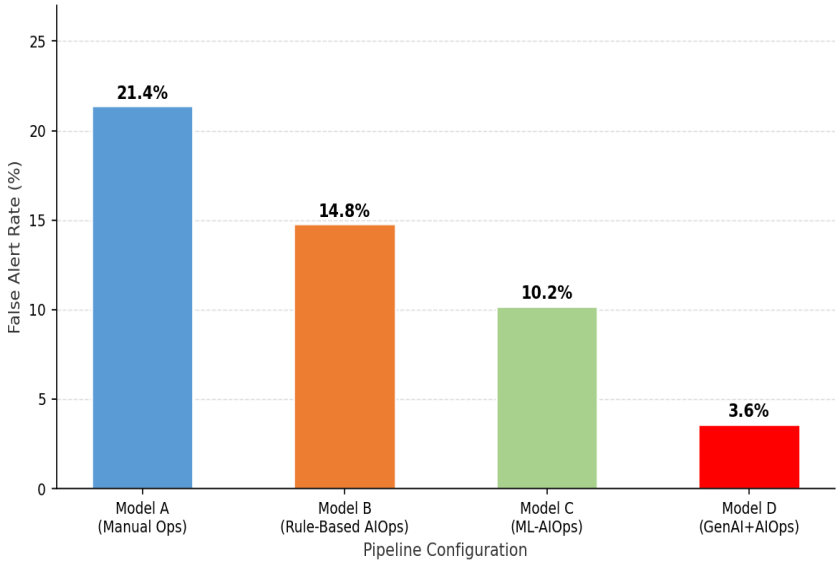

Figure 7. Pipeline False Alert Rate (%)

3.1.7. Autonomous Pipeline Performance Index (APPI)

Fig. 8 presents the APPI scores computed per Eq. 14. Model D scores 0.87 compared to 0.28 (Model A), 0.44 (Model B), and 0.59 (Model C). The 47.5% gain over Model C reflects unified AIOps observability, GenAI-powered self-healing, and superior cross-domain correlation unavailable in single-paradigm architectures.

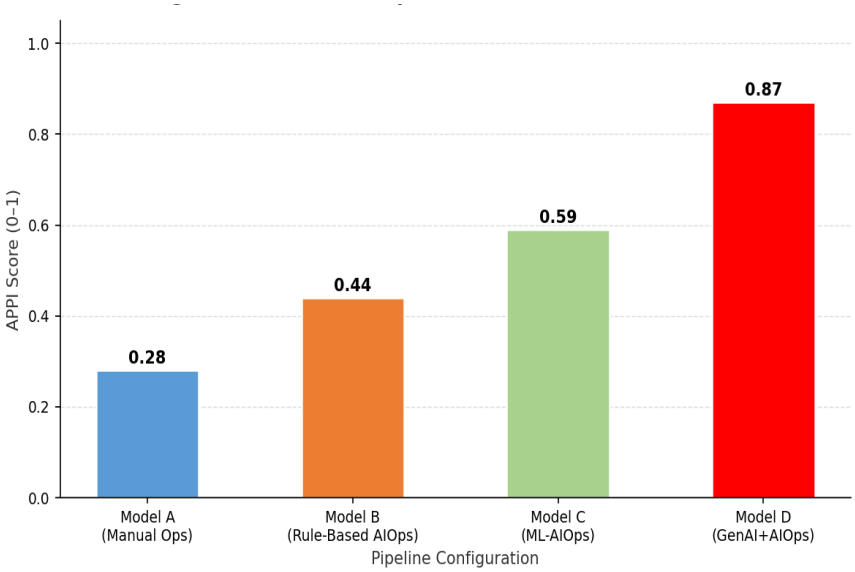

Figure 8. Autonomous Pipeline Performance Index (APPI) Comparison

3.1.8. Data Throughput

Fig. 9 compares sustained pipeline throughput. Model D processes 7,850 events/second—a 528% improvement over Model A (1,250) and 90.5% over Model C (4,120). Databricks auto-scaling governed by AIOps SLO policies enables elastic capacity expansion without manual operator intervention.

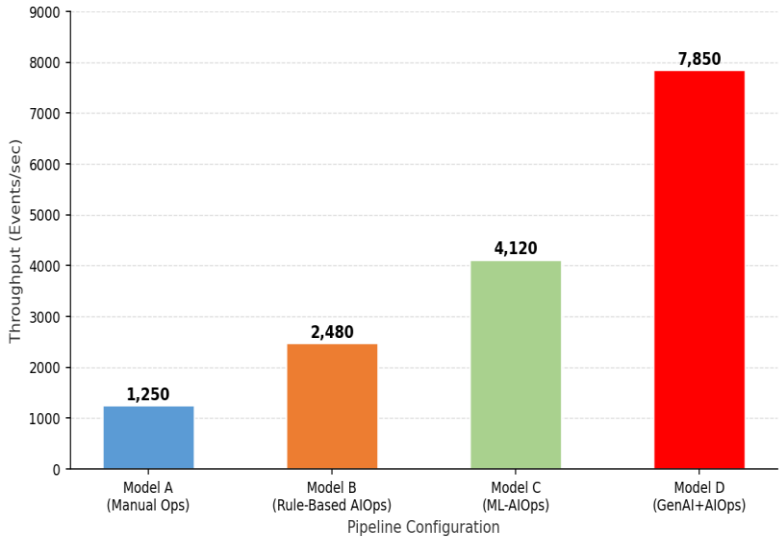

Figure 9. Data Throughput Capacity (Events/Second)

4. Intelligent Control Loops for Autonomous Operations

Intelligent control loops enable autonomous action and decision-making across the entire cloud-native industry-analytics pipeline. Control loops connect the control system to the process being controlled, using feedback to determine if action is required. Autonomous operations close these loops automatically by defining triggers, actions, and decision policies along with learning signals to enhance future performance. Such automation enables a self-driving experience for cloud-native pipelines where operations, incidents, and unexpected conditions are managed automatically according to policies set by engineering and governance teams.

Intelligent control loops address the procedural and operational tasks required to support continuous deployment pipelines. Operational processes stem from business requirements and service-level agreements (SLAs) and aim to ensure reliable and resilient data-to-insight pipelines. Procedural tasks relate to operational excellence and data-product quality, encompassing non-PaaS platform operations and activities such as incident resolution and capacity management. Autonomous execution of these tasks allows the Control Room to focus on monitoring and managing service-enabling objectives, buffs requirements, and desired qualities rather than day-to-day operations. Safety, explainability, and human-in-the-loop capability remain priorities.

4.1. Monitoring, anomaly detection, and self-healing

Observability signals from all components enable detection of irregular behaviour, with AIOps controlling incident response and recovery. Alerts trigger actions, automate responses in well-define cases, and inform operators under uncertain conditions, with policies driving recovery. Telemetry maturity models shape monitoring and defence budgets.

Telemetry signals from application components inform performance, availability, and security operation, with AIOps driving ongoing improvement. Anomalies support detection of service disruptions, with ML-based alerting determine when action is needed and automation level. Signature-based detection automates response to known threats, while exploratory analysis supports operator investigation of novel events.

Self-healing workflows maintain availability in known incident classes, with rollback actions reverting to available states. Continuous monitoring of signatures, policies, and precondition shapes defence priorities, directing effort toward high-risk areas. Safety, explainability, and a human-in-the-loop approach guide the design of control loops, ensuring recovery remain trustworthy.

Signals from the cloud-native environment capture system behaviour, with AIOps processes monitoring and improving operations. Observability maturity models and detection-PRE systems mature alerts and detection cubes, feeding ML-based pipelines that determine when well-understood events warrant automated action and when operational visibility aids in decision-making. Signature-driven automation addresses well-understood classes of incidents, while exploratory analysis responds to previously unknown events.

5. AIOps-Driven Observability and Reliability Engineering

AIOps-Driven Observability and Reliability Engineering Cloud-native industry analytics are moving from a pure DevOps paradigm, where software is delivered and monitored by a dedicated team, toward a new AIOps paradigm, where the entire lifecycle is automated and governed using closed intelligent control loops. AIOps collects, aggregates, visualizes, analyzes, and acts on a wide variety of observability signals across business and technology domains. End-to-end observability engineering provides a set of reliable service-level objectives that the AIOps functions continuously monitor. AIOps combines support for building observability into analytics workflows with helping service reliability teams. From a telemetry perspective, it supports observability checks and unlocks pipeline optimization and resilience. For high-impact, high-volume, analytics service failures, human-in-the-loop workflows drive recovery.

Industry analytics services constitute a natural abstraction layer. Enabling observability-driven service reliability for cloud-native stacks with many microservices requires managing Service-Level Indicators (SLIs) and Service-Level Objectives (SLOs) across a top-down hierarchy. Latency- and profile-sampling-based SLOs address the critical dynamics of cloud-native stacks. Telemetry informs optimization and resilience planning. Detecting high-resource-consuming traversals in business use cases helps monitoring engineers plan preventive actions. Integrating telemetry with profiling indicates paths that can be made more cost-effective.

5.1. Telemetry, tracing, and profiling in dynamic cloud-native stacks

A hybrid telemetry architecture for dynamic cloud-native stacks combines capabilities from traditional and modern paradigms. Tracing—essential for understanding complex operational flows—remains a necessity, with the added element of profiling. Resources must be profiled and optimized, not merely instrumented. Furthermore, real-time ingestion of telemetry data can yield immediate site improvements, yet also presents risks. Ensuring privacy and compliance requires a risk-presence analysis. Together, these considerations shape telemetry selection and implementation in cloud-native stacks.

At the tracing layer, all service-to-service communication must be tracked via a low-overhead, sampled tracing mechanism capable of storing the trace in a telemetry store optimized for quick consumer access. Granular latency analyses identify specific traces that are slow or distinctly different, propelling teams to investigate the cause. A resource profiling layer should interrogate system telemetry at set intervals to report on service resource utilization, test for service oversaturation, and suggest resource scaling modifications. In parallel, a latency quantiles lens should evaluate user-request timings across services at various levels of detail to

identify slow components on a more persistent basis. Both the profiling and testing layers support services by using telemetry signals to proactively recommend changes.

5.2. Tabular Comparisons and Key Metrics

5.2.1. Comparative Detection and Performance Metrics

Table 2 provides a comprehensive performance comparison across all four pipeline configurations, with explicit improvements of Model D over Models A and C.

Table 2. Comparative Detection and Performance Metrics

Metric	Model A	Model B	Model C	Model D	Improvement (D vs A)	Improvement (D vs C)
Automation F1-Score (%)	51.4	64.8	76.2	92.7	↑ 80.3%	↑ 21.7%
False Alert Rate (%)	21.4	14.8	10.2	3.6	↓ 83.2%	↓ 64.7%
Pipeline Autonomy Score (%)	28.6	47.3	68.9	94.1	↑ 229%	↑ 36.6%
Operational Cost (Norm. Units)	82.6	67.3	53.8	38.4	↓ 53.5%	↓ 28.6%
APPI Score (0–1)	0.28	0.44	0.59	0.87	↑ 210.7%	↑ 47.5%

Table 2 affirms that Model D yields substantial improvements across all performance dimensions, with the most significant being the 210.7% increase in APPI score and the 83.2% reduction in false alert rate relative to the conventional manual operations baseline (Model A).

6. Conclusion

Generative AI and AIOps for Autonomous Workflow Automation in Cloud-Native Industry Analytics Pipelines: Use an objective, scholarly tone with evidence-based arguments and formal structure; synthesize concepts, data, and implications clearly. Cloud-native industry analytics pipelines have disruptive potential in business domains by enabling cloud scalability and on-demand data-driven decision-making. However, knowledge and resource gaps hinder automation of the data-to-insights chain. An integrative approach combining Generative AI and AIOps offers AI-assisted operator decision support, enabling autonomous control of workflow orchestration and execution. AI components exchange observability signals with the data-management architecture supporting AIOps-enabled control loops for fault-tolerant and self-healing operations.

The potential for discovering insights from data is great, particularly in dynamic cloud-native industry analytics pipelines. Communications and utilities, large-scale operations concerning infrastructure and resources, and businesses across the production life cycle stand to benefit from scalable workflows using data of any type, form, or data source available in or outside the organisation. Most importantly, these workflows enable cloud-scale analytics and are capable of supporting informed decision-making based on data-driven insights delivered in a timely manner. Nevertheless, while businesses can stand to gain so much from these capabilities, resources and expertise to autonomously operationalise such workflows 24/7, or even at scale, often remain elusive. Typically, analysts embark on these journeys manually or as one-off projects for strategic delivery, and the uptake of automation is yet to become pervasive."

References

[1] Akidau, T., Bradshaw, R., Chambers, C., Chernyak, S., Fernández-Moctezuma, R. J., Lax, R., McVeety, S., Mills, D., Perry, F., Schmidt, E., & Whittle, S. (2015). The dataflow model: A practical approach to balancing correctness, latency, and cost in massive-scale, unbounded, out-of-order data processing. Proceedings of the VLDB Endowment, 8(12), 1792–1803. [https://doi.org/10.14778/2824032.2824076] (https://doi.org/10.14778/2824032.2824076)

[2] Alexandrov, A., Bergmann, R., Ewen, S., Freytag, J.-C., Hueske, F., Heise, A., Kao, O., Leich, M., Leser, U., Markl, V., Naumann, F., Peters, M., Rheinländer, A., Sax, M. J., Schelter, S., Höger, M., Tzoumas, K., & Warneke, D. (2014). The Stratosphere platform for big data analytics. The VLDB Journal, 23(6), 939–964. https://doi.org/10.1007/s00778-014-0357-y

[3] Ebert, C., & Louridas, P. (2023). Generative AI for software practitioners. *IEEE Software*, 40(4), 30–38. https://doi.org/10.1109/MS.2023.3265877

[4] Noy, S., & Zhang, W. (2023). Experimental evidence on the productivity effects of generative artificial intelligence. Science, 381 (6654), 187–192. https://doi.org/10.1126/science.adh2586

[5] Doshi, A. R., & Hauser, O. P. (2024). Generative AI enhances individual creativity but reduces the collective diversity of novel content. *Science Advances*, 10(28), eadn5290. https://doi.org/10.1126/sciadv.adn5290

[6] van Dis, E. A. M., Bollen, J., Zuidema, W., van Rooij, R., & Bockting, C. L. (2023). ChatGPT: Five priorities for research. Nature, 614(7947), 224–226. https://doi.org/10.1038/d41586-023-00288-7

[7] AZITH, T. G. V. (2023). Exploring the intersection of bioethics and AI-driven clinical decision-making: Navigating the ethical challenges of deep learning applications in personalized medicine and experimental treatments. JOURNAL OF MATERIAL SCIENCES & MANUFACTURING RESEARCH Учредители: Scientific Research and Community Ltd, 4(6), 1-10.

[8] Creswell, A., White, T., Dumoulin, V., Arulkumaran, K., Sengupta, B., & Bharath, A. A. (2018). Generative adversarial networks: An overview. *IEEE Signal Processing Magazine, 35*(1), 53–65. https://doi.org/10.1109/MSP.2017.2765202

[9] Gui, J., Sun, Z., Wen, Y., Tao, D., & Ye, J. (2023). A review on generative adversarial networks: Algorithms, theory, and applications. *IEEE Transactions on Knowledge and Data Engineering*, 35(4), 3313–3332. https://doi.org/10.1109/TKDE.2021.3130191

[10] Pandugula, C., & Yasmeen, Z. (2023). Exploring Advanced Cybersecurity Mechanisms for Attack Prevention in Cloud-Based Retail Ecosystems. Journal for ReAttach Therapy and Developmental Diversities, 6, 1704-1714.

[11] Jennings, N. R., Sycara, K., & Wooldridge, M. (1998). A roadmap of agent research and development. *Autonomous Agents and Multi-Agent Systems*, 1(1), 7–38. https://doi.org/10.1023/A:1010090405266

[12] Shoham, Y. (1993). Agent-oriented programming. *Artificial Intelligence*, 60(1), 51–92. [https://doi.org/10.1016/0004-3702(93)90034-9](https://doi.org/10.1016/0004-3702%2893%2990034-9)

[13] Macal, C. M., & North, M. J. (2010). Tutorial on agent-based modelling and simulation. *Journal of Simulation*, 4(3), 151–162. https://doi.org/10.1057/jos.2010.3

[14] Vankayalapati, R. K., & Seenu, A. (2023). Energy-efficient ai clusters: Reducing carbon footprints with cloud and high-speed storage synergies. Available at SSRN 5091827.

[15] Wooldridge, M., & Ciancarini, P. (2001). Agent-oriented software engineering: The state of the art. In P. Ciancarini & M. Wooldridge (Eds.), *Agent-oriented software engineering* (pp. 1–28). Springer. https://doi.org/10.1007/3-540-44564-1_1

[16] Reddy, R., Yasmeen, Z., Maguluri, K. K., & Ganesh, P. (2023). Impact of AI-Powered Health Insurance Discounts and Wellness Programs on Member Engagement and Retention. Letters in High Energy Physics, 2023.

[17] Buyya, R., Yeo, C. S., Venugopal, S., Broberg, J., & Brandic, I. (2009). Cloud computing and emerging IT platforms: Vision, hype, and reality for delivering computing as the fifth utility. *Future Generation Computer Systems*, 25(6), 599–616. https://doi.org/10.1016/j.future.2008.12.001

[18] Pamisetty, V. (2023). Leveraging AI, big data, and cloud computing for enhanced tax compliance, fraud detection, and fiscal impact analysis in government financial management. Fraud Detection, and Fiscal Impact Analysis in Government Financial Management (December 15, 2023).

[19] Burns, B., Grant, B., Oppenheimer, D., Brewer, E., & Wilkes, J. (2016). Borg, Omega, and Kubernetes. Communications of the ACM, 59(5), 50–57. https://doi.org/10.1145/2890784

[20] Recharla, M., Nuka, S. T., Chakilam, C., Chava, K., & Suura, S. R. (2023). Next-generation technologies for early disease detection and treatment: harnessing intelligent systems and genetic innovations for improved patient outcomes. Journal for ReAttach Therapy and Developmental Diversities, 6, 1921-1937.

[21] Dragoni, N., Giallorenzo, S., Lluch Lafuente, A., Mazzara, M., Montesi, F., Mustafin, R., & Safina, L. (2017). Microservices: Yesterday, today, and tomorrow. In M. Mazzara & B. Meyer (Eds.), *Present and ulterior software engineering* (pp. 195–216). Springer. https://doi.org/10.1007/978-3-319-67425-4_12

[22] Aravind, R., & Shah, C. V. (2023). Physics model-based design for predictive maintenance in autonomous vehicles using AI. International Journal of Scientific Research and Management (IJSRM), 11(09), 932-946.

[23] Challa, K. (2023). Optimizing Financial Forecasting Using Cloud Based Machine Learning Models. Journal for ReAttach Therapy and Developmental Diversities. https://doi. org/10.53555/jrtdd. v6i1os (2), 3565.

[24] Di Francesco, P., Malavolta, I., & Lago, P. (2017). Research on architecting microservices: Trends, focus, and potential for industrial adoption. In *2017 IEEE International Conference on Software Architecture (ICSA)* (pp. 21–30). IEEE. https://doi.org/10.1109/ICSA.2017.24

[25] Varghese, B., & Buyya, R. (2018). Next generation cloud computing: New trends and research directions. *Future Generation Computer Systems*, 79, 849–861. https://doi.org/10.1016/j.future.2017.09.020

[26] Pamisetty, A. (2023). Cloud-driven transformation of banking supply chain analytics using big data frameworks. Available at SSRN.

[27] Lwakatare, L. E., Kilamo, T., Karvonen, T., Sauvola, T., Heikkilä, V., Itkonen, J., Kuvaja, P., Mikkonen, T., Oivo, M., & Lassenius, C. (2019). DevOps in practice: A multiple case study of five companies. *Information and Software Technology*, 114, 217–230. https://doi.org/10.1016/j.infsof.2019.06.010

[28] Komaragiri, V. B. (2023). Leveraging Artificial Intelligence to Improve Quality of Service in Next-Generation Broadband Networks. Journal for ReAttach Therapy and Developmental Diversities. https://doi. org/10.53555/jrtdd. v6i1os (2), 3571.

[29] Shahin, M., Babar, M. A., & Zhu, L. (2017). Continuous integration, delivery and deployment: A systematic review on approaches, tools, challenges and practices. *IEEE Access*, 5, 3909–3943. https://doi.org/10.1109/ACCESS.2017.2685629

[30] Chen, L. (2015). Continuous delivery: Huge benefits, but challenges too. *IEEE Software*, 32(2), 50–54. https://doi.org/10.1109/MS.2015.27

[31] Kannan, S. (2023). The Convergence of AI, Machine Learning, and Neural Networks in Precision Agriculture: Generative AI as a Catalyst for Future Food Systems. JOURNAL FOR REATTACH THERAPY AND DEVELOPMENTAL DIVERSITIES.

[32] Leppänen, M., Mäkinen, S., Pagels, M., Eloranta, V.-P., Itkonen, J., Mäntylä, M. V., & Männistö, T. (2015). The highways and country roads to continuous deployment. *IEEE Software*, 32(2), 64–72. https://doi.org/10.1109/MS.2015.50

[33] Suura, S. R., Chava, K., Recharla, M., & Chakilam, C. (2023). Evaluating Drug Efficacy and Patient Outcomes in Personalized Medicine: The Role of AI-Enhanced Neuroimaging and Digital Transformation in Biopharmaceutical Services. Journal for ReAttach Therapy and Developmental Diversities, 6, 1892-1904.

[34] Mäntylä, M. V., Adams, B., Khomh, F., Engström, E., & Petersen, K. (2015). On rapid releases and software testing: A case study and a semi-systematic literature review. *Empirical Software Engineering*, 20(5), 1384–1425. https://doi.org/10.1007/s10664-014-9338-4

[35] Annapareddy, V. N., Gadi, A. L., Komaragiri, V. B., Koppolu, H. K. R., & Kannan, S. (2023). AI-Driven Optimization of Renewable Energy Systems: Enhancing Grid Efficiency and Smart Mobility Through 5G and 6G Network Integration. Educational Administration: Theory and Practice, 29(4), 4748-4763.

[36] Amershi, S., Begel, A., Bird, C., DeLine, R., Gall, H., Kamar, E., Nagappan, N., Nushi, B., & Zimmermann, T. (2019). Software engineering for machine learning: A case study. In *2019 IEEE/ACM 41st International Conference on Software Engineering: Software Engineering in Practice* (pp. 291–300). IEEE. https://doi.org/10.1109/ICSE-SEIP.2019.00042

[37] Inala, R. (2023). AI-powered investment decision support systems: Building smart data products with embedded governance controls. Journal for ReAttach Therapy and Developmental Diversities, 6(10), 2251-2266.

[38] Baylor, D., Breck, E., Cheng, H.-T., Fiedel, N., Foo, C. Y., Haque, Z., Haykal, S., Ispir, M., Jain, V., Koc, L., Koo, C. Y., Lew, L., Mewald, C., Modi, A. N., Polyzotis, N., Ramesh, S., Roy, S., Whang, S. E., Wicke, M., … Zinkevich, M. (2017). TFX: A TensorFlow-based production-scale machine learning platform. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (pp. 1387–1395). ACM. https://doi.org/10.1145/3097983.3098021

[39] Sheelam, G. K. (2023). Adaptive AI workflows for edge-to-cloud processing in decentralized mobile infrastructure. Journal for Reattach Therapy and Development Diversities. https://doi. org/10.53555/jrtdd. v6i10s (2). 357ough Predictive Intelligence.

[40] Aravind, R., & Surabhii, S. N. R. D. (2023). Harnessing Artificial Intelligence for Enhanced Vehicle Control and Diagnostics.

[41] Kummari, D. N. (2023). Energy Consumption Optimization in Smart Factories Using AI-Based Analytics: Evidence from Automotive Plants. Journal for Reattach Therapy and Development Diversities. https://doi. org/10.53555/jrtdd. v6i10s (2), 3572.

[42] Gomber, P., Kauffman, R. J., Parker, C., & Weber, B. W. (2018). On the FinTech revolution: Interpreting the forces of innovation, disruption, and transformation in financial services. *Journal of Management Information Systems*, 35(1), 220–265. https://doi.org/10.1080/07421222.2018.1440766

[43] Lakkarasu, P., Kaulwar, P. K., Dodda, A., Singireddy, S., & Burugulla, J. K. R. (2023). Innovative computational frameworks for secure financial ecosystems: Integrating intelligent automation, risk analytics, and digital infrastructure. *International Journal of Finance (IJFIN)-ABDC Journal Quality List*, *36*(6), 334-371.

[44] Puschmann, T. (2017). FinTech. *Business & Information Systems Engineering*, 59(1), 69–76. https://doi.org/10.1007/s12599-017-0464-6

[45] Kalisetty, S., & Singireddy, J. (2023). Agentic AI in retail: A paradigm shift in autonomous customer interaction and supply chain automation. American Advanced Journal for Emerging Disciplinaries (AAJED) ISSN, 3067-4190.

[46] Vives, X. (2019). Digital disruption in banking. *Annual Review of Financial Economics*, 11, 243–272. https://doi.org/10.1146/annurev-financial-100719-120854

[47] Kaulwar, P. K. (2023). Tax Optimization and Compliance in Global Business Operations: Analyzing the Challenges and Opportunities of International Taxation Policies and Transfer Pricing. *International Journal of Finance (IJFIN)-ABDC Journal Quality List*, *36*(6), 150-181.

[48] Aravind, R., Surabhi, S. N. D., & Shah, C. V. (2023). Remote Vehicle Access: Leveraging Cloud Infrastructure for Secure and Efficient OTA Updates with Advanced AI. EuropeanEconomic Letters (EEL), 13 (4), 1308–1319. Retrieved fromhttps.

[49] Paleti, S. (2023). Transforming Money Transfers and Financial Inclusion: The Impact of AI-Powered Risk Mitigation and Deep Learning-Based Fraud Prevention in Cross-Border Transactions. Available at SSRN, 5158588.

[50] Bartlett, R., Morse, A., Stanton, R., & Wallace, N. (2022). Consumer-lending discrimination in the FinTech era. *Journal of Financial Economics*, 143(1), 30–56. https://doi.org/10.1016/j.jfineco.2021.05.047

[51] Adusupalli, B. (2023). DevOps-Enabled Tax Intelligence: A Scalable Architecture for Real-Time Compliance in Insurance Advisory. Journal for Reattach Therapy and Development Diversities. Green Publication. https://doi. org/10.53555/jrtdd. v6i10s (2), 358.

[52] Sezer, O. B., Gudelek, M. U., & Ozbayoglu, A. M. (2020). Financial time series forecasting with deep learning: A systematic literature review: 2005–2019. *Applied Soft Computing*, 90, 106181. https://doi.org/10.1016/j.asoc.2020.106181

[53] Nandan, B. P., & Chitta, S. S. (2023). Machine Learning Driven Metrology and Defect Detection in Extreme Ultraviolet (EUV) Lithography: A Paradigm Shift in Semiconductor Manufacturing. Educational Administration: Theory and Practice, 29 (4), 4555–4568. International Journal of Scientific Research and Modern Technology, 1(12), 216-226.

[54] Kalisetty, S., & Singireddy, J. (2023). Optimizing Tax Preparation and Filing Services: A Comparative Study of Traditional Methods and AI Augmented Tax Compliance Frameworks. Available at SSRN 5206185.

[55] Pandugula, C., & Nampalli, R. C. R. Optimizing Retail Performance: Cloud-Enabled Big Data Strategies for Enhanced Consumer Insights.

[56] Recharla, M. Integrated Genomic and Neurobiological Pathway Mapping for Early Detection of Alzheimer's Disease. International Journal of Advanced Research in Computer and Communication Engineering (IJARCCE), DOI, 10.

[57] Pandiri, L., & Chitta, S. (2022). Leveraging AI and Big Data for Real-Time Risk Profiling and Claims Processing: A Case Study on Usage-Based Auto Insurance. Kurdish Studies. Green Publication. https://doi. org/10.53555/ks. v10i2, 3760.

[58] Zhang, Q., Cheng, L., & Boutaba, R. (2010). Cloud computing: State-of-the-art and research challenges. *Journal of Internet Services and Applications*, 1(1), 7–18. https://doi.org/10.1007/s13174-010-0007-6

[59] Mashetty, S. (2023). View of A Comparative Analysis of Patented Technologies Supporting Mortgage and Housing Finance. Educational Administration: Theory and Practice.

[60] Xu, W., Huang, L., Fox, A., Patterson, D., & Jordan, M. I. (2009). Detecting large-scale system problems by mining console logs. In *Proceedings of the 22nd ACM Symposium on Operating Systems Principles* (pp. 117–132). Association for Computing Machinery. https://doi.org/10.1145/1629575.1629587

[61] Zaharia, M., Das, T., Li, H., Hunter, T., Shenker, S., & Stoica, I. (2013). Discretized streams: Fault-tolerant streaming computation at scale. In *Proceedings of the 24th ACM Symposium on Operating Systems Principles* (pp. 423–438). Association for Computing Machinery. https://doi.org/10.1145/2517349.2522737

[62] Zaharia, M., Xin, R. S., Wendell, P., Das, T., Armbrust, M., Dave, A., Meng, X., Rosen, J., Venkataraman, S., Franklin, M. J., Ghodsi, A., Gonzalez, J., Shenker, S., & Stoica, I. (2016). Apache Spark: A unified engine for big data processing. *Communications of the ACM*, 59(11), 56–65. https://doi.org/10.1145/2934664

[63] Zhang, X., Xu, Y., Lin, Q., Qiao, B., Zhang, H., Dang, Y., Xie, C., Yang, X., Cheng, Q., Li, Z., Chen, J., He, X., Yao, R., Lou, J.-G., Chintalapati, M., Shen, F., & Zhang, D. (2019). Robust log-based anomaly detection on unstable log data. In *Proceedings of the 27th ACM Joint European Software Engineering Conference and Symposium on the Foundations of Software Engineering* (pp. 807–817). Association for Computing Machinery. https://doi.org/10.1145/3338906.3338931