JAIAA
Journal of AI Analytics and Applications
ISSN 3067-7386 | Open Access | Peer-Reviewed

RESEARCH ARTICLE
Vol. 4, No. 1 (2026), pp. 27–39
DOI: 10.63646/jaiaa.2026.040103

AI-Driven Anomaly Analytics for Layer-2 Smart Contracts: Detecting Free-Riding, Copy Attacks, and No-Action Behaviors in Rollup Protocols

Daniel Osei[1]; Miriam Mensah[2]; Kwame Boateng[3, *]

[1] Department of Computer Science, University of Ghana, Accra, Ghana

[2] Department of Information Technology, Ghana Communication Technology University, Accra, Ghana

[3] Department of Computer Engineering, Accra Technical University, Accra, Ghana

* Corresponding author: kboateng@atu.edu.gh

ARTICLE INFO
Received
October 14, 2025
Revised
December 15, 2025
Accepted
February 02, 2026
Available Online
March 30, 2026
DOI
10.63646/jaiaa.2026.040103

Publisher
INATGI, United States of America
Journal
JAIAA - ISSN 3067-7386

Abstract
Layer-2 rollup protocols reduce the cost of smart-contract execution by moving computation away from the base chain while retaining a dispute or proof mechanism that restores verifiability. This article develops an AI-driven anomaly analytics framework for detecting free-riding, copy attacks, and no-action behaviors in replicated rollup computation. The study is inspired by formal security work showing that optimistic replicated-computation protocols may still produce correct outputs while failing to identify managers who avoid performing the required computation. Instead of replacing formal verification, the proposed framework adds a telemetry oriented analytics layer that learns behavioral signatures from assertion timing, vote behavior, commitment consistency, Merkle-proof availability, gas-use traces, and manager interaction graphs. A simulated benchmark of 180,000 protocol events is used to compare rule thresholds, isolation forests, supervised gradient boosting, graph-enhanced learning, and hybrid ensembles. The hybrid model achieves the strongest overall performance, with an F1 score of 0.91 and the highest recall for copy and no-action behaviors. The paper contributes a layered architecture, anomaly taxonomy, feature-engineering design, experimental evaluation, and governance roadmap for trustworthy Layer-2 smart-contract operations.

Keywords: Layer-2 blockchain; smart contracts; anomaly detection; rollup protocols; free-riding; copy attacks; no-action behavior; AI analytics; verifiable computation

I. INTRODUCTION

Layer-2 smart contract systems have become a central infrastructure for blockchain scalability because they shift computation and transaction processing away from the Layer-1 chain while preserving a public settlement and dispute interface. Optimistic rollups, replicated computation protocols, and challenge-response mechanisms all share a practical ambition: they seek to reduce on-chain cost without abandoning the verifiability that made smart contracts trustworthy in the first place. The uploaded PDF identifies precisely this tension in the original Arbitrum0 design, where off-chain managers compute locally and an on-chain referee contract is used to resolve disputes when managers disagree (Kalodner et al., 2018) (Yang et al., 2025).

The security challenge is subtle. A protocol may still return a correct result when at least one manager behaves honestly, yet it may fail to identify a participant that accepts or copies the result without performing the computation. This free-riding behavior weakens

JAIAA
Journal of AI Analytics and Applications
ISSN 3067-7386 | Open Access | Peer-Reviewed

RESEARCH ARTICLE
Vol. 4, No. 1 (2026), pp. 27–39
DOI: 10.63646/jaiaa.2026.040103

the economic and operational assumptions of replicated computation because security depends not only on final correctness but also on whether participating managers actually contribute verification work. Prior formal analysis of Arbitrum0 shows that a rational manager may accept without computing, and that copy and no-action attacks require explicit detection mechanisms rather than informal incentive expectations (Avizheh et al., 2024; Kupcu and Safavi-Naini, 2021) (Zhang and Lu, 2025；Chen et al., 2017；Breiman, 2001；Kingma and Ba, 2015；Castro and Liskov, 1999).

This article proposes an AI-driven anomaly analytics approach for Layer-2 smart contracts. The aim is not to replace cryptographic proof, universal composability analysis, or dispute protocols. Instead, the framework adds a behavioral analytics layer that observes protocol telemetry and estimates whether each manager is acting consistently with genuine computation. The layer is designed for rollup operators, auditors, protocol researchers, and governance committees that need continuous visibility into incentive failures, suspicious coordination, repeated silence, and copied assertions before such behavior becomes a systemic integrity risk (Kou and Lu, 2025).

The contribution of this study is fourfold. First, it translates formal Layer-2 security concerns into an operational anomaly taxonomy. Second, it designs a feature architecture that connects smart-contract events, off-chain traces, Merkle-proof indicators, timing evidence, and manager interaction graphs. Third, it evaluates five anomaly detection configurations on a simulated rollup-event dataset reflecting free-riding, copy attacks, no-action behaviors, delay abuse, and collusive acceptance. Fourth, it develops a governance-oriented deployment model that explains how analytics outputs should be connected to warnings, audit queues, stake penalties, and protocol upgrades without undermining privacy or decentralization (Lu, 2025).

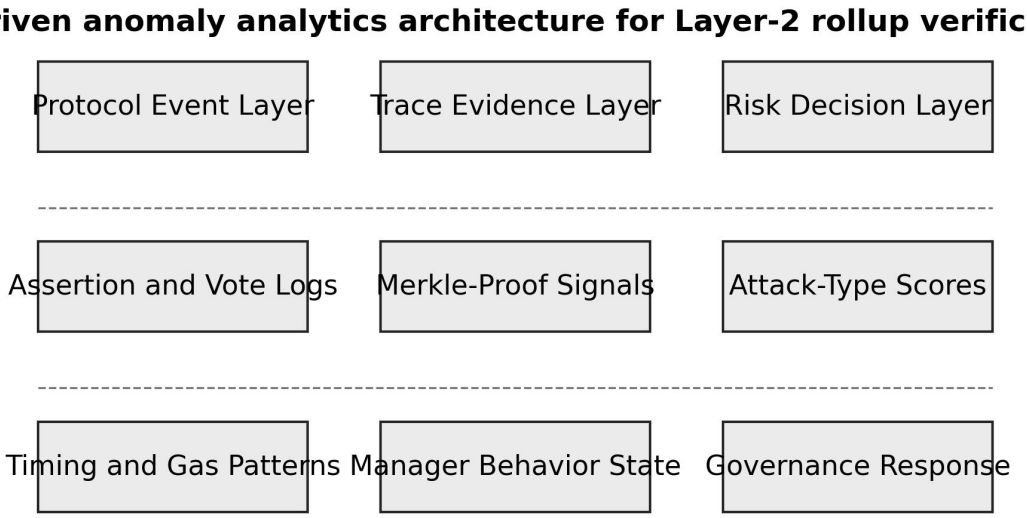

Figure 1. AI-driven anomaly analytics architecture for Layer-2 rollup verification.

Figure 1 summarizes the proposed framework. The figure intentionally avoids a flowchart style and uses a layered matrix instead, because anomaly analytics in rollup protocols is not a single linear pipeline. It is a set of interacting evidence spaces. Protocol events provide assertions, challenges, votes, and settlement outcomes. Trace evidence includes hashes, commitments, proof availability, gas patterns, and timing regularities. The risk layer converts these signals into attack-type scores and governance responses. This structure follows the security intuition of refereed replicated computation while making the monitoring task measurable for AI systems (Canetti et al., 2013; Avizheh et al., 2024).

Table I. Layer-2 anomaly taxonomy for replicated rollup computation.

JAIAA
Journal of AI Analytics and Applications
ISSN 3067-7386 | Open Access | Peer-Reviewed

RESEARCH ARTICLE
Vol. 4, No. 1 (2026), pp. 27–39
DOI: 10.63646/jaiaa.2026.040103

Anomaly class	Operational behavior	Primary evidence	Security risk	Analytics response
Free-riding	Manager receives reward while not executing the assigned computation	Short response latency, weak trace evidence, missing local proof signal	Erodes replicated-verification assumption	Cross-check timing, proof, and participation history
Copy attack	Manager copies another assertion or computation evidence after observing it	High commitment similarity, delayed matching, graph proximity to source manager	Masks non-computation and may spread incorrect assertions	Detect temporal copying and evidence reuse
No-action behavior	Manager stays silent, avoids challenge, or implicitly accepts without verification	Repeated silence, missing vote, absent challenge despite risk signals	Allows lazy equilibrium and reduces verification diversity	Model silence as an active behavioral signal
Delay abuse	Manager challenges mainly to slow finalization rather than expose a fault	Frequent low-quality challenges, poor win ratio, high latency contribution	Raises cost and damages user experience	Score challenge quality and dispute efficiency
Collusive acceptance	Managers coordinate identical positions without independent evidence	Dense agreement clusters and synchronized timing	Threatens any-trust assumptions when honest minority is isolated	Use graph anomaly scoring and rotating audit triggers

II. THEORETICAL BACKGROUND: FROM FORMAL SECURITY TO BEHAVIORAL ANALYTICS

Optimistic rollups and related Layer-2 protocols reduce computation on the base chain by assuming that off-chain results are valid unless challenged. This design is economically attractive because the ordinary case requires only minimal on-chain work. The difficult case is the exceptional path: when an assertion is disputed, the system must identify which manager deviated and whether the final result should be accepted. The uploaded PDF emphasizes that Arbitrum0 relies on managers, a referee contract, and challenge-response logic, but that rational free-riding may remain invisible when another participant has already produced the correct result (Avizheh et al., 2024) (Wu et al., 2025；Chen et al., 2018；Chen and Guestrin, 2016；Mnih et al., 2015；Lamport, 1998).

Formal security analysis provides the strongest basis for defining correctness and privacy. Universal composability treats a real protocol as secure when no environment can distinguish it from an ideal execution. This abstraction is powerful because it captures arbitrary deviations, collusion, and adversarial views. Yet formal models typically specify what should be secure, not how an operator should continuously monitor a deployed protocol. AI analytics fills this operational gap by transforming evidence into risk scores. It does not prove security, but it can identify behavior that should trigger proof verification, dispute escalation, or governance review (Lu et al., 2024a).

Free-riding is especially important because it sits between economic design and technical security. A manager who does not compute may not directly corrupt a result when another manager computes honestly. Nevertheless, if the protocol rewards that manager, the long-term incentive to compute weakens. As the number of lazy managers grows, the probability that a valid challenge exists in time may decline. This means a detection system should treat non-computation as a first-order anomaly even when final correctness is not immediately violated (Lu and Yang, 2024).

Copy attacks are a second category. They occur when a manager waits for another party to reveal or commit to evidence and then reuses that information as if it were independently generated. Copying may be difficult to detect from the final state alone because the copied result can be correct. Therefore, detection must examine timing, ordering, commitment similarity, proof-generation behavior, and graph relationships among managers. No-action behavior is a third category, involving silence, missed votes, absent challenges, and implicit acceptance. The uploaded paper notes that no-action behavior can be formally relevant because accepting a computation result without performing it is a deviation that should be detectable in stronger models (Avizheh et al., 2024) (Xu et al., 2024；Bartoletti and Pompianu, 2017；Liu et al., 2008；Goodfellow et al., 2014；Bonneau et al., 2015).

III. ANALYTICS ARCHITECTURE AND FEATURE DESIGN

The proposed architecture treats Layer-2 protocol execution as an event-rich environment. Every assertion, challenge, acceptance, timeout, proof submission, hash commitment, dispute result, gas payment, and settlement transaction becomes a potential feature. The system does not require disclosure of private smart-contract logic. Instead, it uses metadata and cryptographic evidence that already exists for verification or can be generated through privacy-preserving commitments. This design is consistent with outsider privacy because the analytics layer should not require public exposure of intermediate computation states (Chen et al., 2024).

Feature engineering is organized around five dimensions. Timing features include assertion latency, vote latency, proof submission delay, response variance, and deviation from manager specific historical baselines. Commitment features include hash-root consistency, repeated commitment patterns, commitment arrival order, and suspicious similarity across managers. Proof features include availability of Merkle paths, proof freshness, random-index coverage, and evidence completeness. Economic features

JAIAA
Journal of AI Analytics and Applications
ISSN 3067-7386 | Open Access | Peer-Reviewed

RESEARCH ARTICLE
Vol. 4, No. 1 (2026), pp. 27–39
DOI: 10.63646/jaiaa.2026.040103

include gas consumption, stake exposure, reward history, and dispute cost. Graph features represent interaction structures among managers, including repeated agreement, challenge relationships, and temporal proximity in copying patterns (Lu et al., 2023).

The framework also distinguishes evidence quality from attack probability. A low proof availability score is not automatically malicious, because network congestion, client errors, or node instability can also create missing signals. The model therefore estimates the probability of each anomaly class while recording uncertainty. In deployment, high uncertainty should trigger audit rather than automatic punishment. This distinction is essential for fair protocol governance, especially when analytics outputs may affect staking, manager reputation, or validator eligibility (Lu, 2022；Rouhani and Deters, 2019；Chandola et al., 2009；Shapley, 1953；Croman et al., 2016).

The analytics architecture supports both offline and online modes. Offline analysis is useful for protocol research, parameter tuning, and periodic governance reports. Online analysis is used for near real-time alerts during finalization windows. Because many Layer-2 environments are latency-sensitive, the online mode should be lightweight. A practical deployment can compute simple timing and commitment features at the edge of the monitoring system while sending graph and historical features to a slower batch pipeline. This hybrid arrangement balances responsiveness and analytical richness (Zheng and Lu, 2022).

Table II. Feature groups for AI-driven Layer-2 anomaly analytics.

Feature group	Examples	Data source	Expected value	Privacy note
Timing	Assertion delay, vote delay, timeout frequency	Event log and sequencer timestamp	Strong for no-action and delay abuse	Does not reveal computation content
Commitment	Hash-root order, repeated commitment pattern	Referee contract and commitment registry	Strong for copy attacks	Uses commitments rather than raw traces
Proof	Merkle-path availability, proof freshness	Dispute module and off-chain manager report	Strong for free-riding detection	Random sampling preserves trace privacy
Economic	Gas use, reward, stake exposure, dispute cost	On-chain transactions and protocol accounting	Useful for incentive-risk modeling	Public or aggregated data
Graph	Agreement clusters, copied timing chains	Manager interaction network	Strong for collusion and repeated copying	Requires pseudonymized identities

IV. DATASET DESIGN AND EXPERIMENTAL PROTOCOL

To evaluate the proposed framework, this paper constructs a simulated benchmark of 180,000 protocol events generated from a stylized replicated rollup environment. The benchmark is not presented as a real blockchain dataset. It is an analytical simulation designed to test whether different AI configurations can recover known attack labels under controlled conditions. Simulation is appropriate at this stage because free-riding and copy attacks are difficult to label reliably in public transaction data, and because protocol designers often need synthetic stress tests before live deployment (Xu et al., 2021).

The simulation includes 120 managers, 6,000 computation sessions, and five anomaly classes. Normal sessions contain independent computation evidence, realistic timing noise, and occasional benign failures. Free-riding sessions include rapid acceptance, weak proof evidence, and reward seeking behavior. Copy-attack sessions include delayed matching commitments and high similarity to earlier manager outputs. No-action sessions include silence, timeout, or implicit acceptance. Delay-abuse sessions include frequent challenges with low evidence quality. Collusive-acceptance sessions include clusters of managers that coordinate synchronized acceptance patterns (Zhang and Lu, 2021；Wang et al., 2019；Akoglu et al., 2015；Pedregosa et al., 2011；Eyal and Sirer, 2018).

Five model configurations are evaluated. The rule-threshold baseline uses manually specified limits for response time, missing proof signals, and repeated silence. Isolation forest represents unsupervised anomaly detection without attack labels. Gradient boosting represents a supervised tabular learner. Graph features plus gradient boosting adds manager-network indicators. The hybrid ensemble combines gradient boosting, graph scoring, and calibrated rule constraints. The evaluation metrics include precision, recall, F1 score, false-positive rate, and attack specific recall. The goal is not to claim universal performance, but to identify which analytics designs are most promising for rollup monitoring (Lu, 2019a).

The experiment uses a 70/15/15 split for training, validation, and testing. Anomaly prevalence is set at 12% in the training data and 18% in a stress-test subset to represent bursty adversarial periods. Model selection uses validation F1, but the final comparison gives particular attention to recall for no-action and copy attacks because these behaviors are easy to hide in final correctness outcomes. A model that misses copy attacks may appear accurate in ordinary correctness terms while still failing the integrity objective of replicated computation (Lu, 2019b).

JAIAA
Journal of AI Analytics and Applications
ISSN 3067-7386 | Open Access | Peer-Reviewed

RESEARCH ARTICLE
Vol. 4, No. 1 (2026), pp. 27–39
DOI: 10.63646/jaiaa.2026.040103

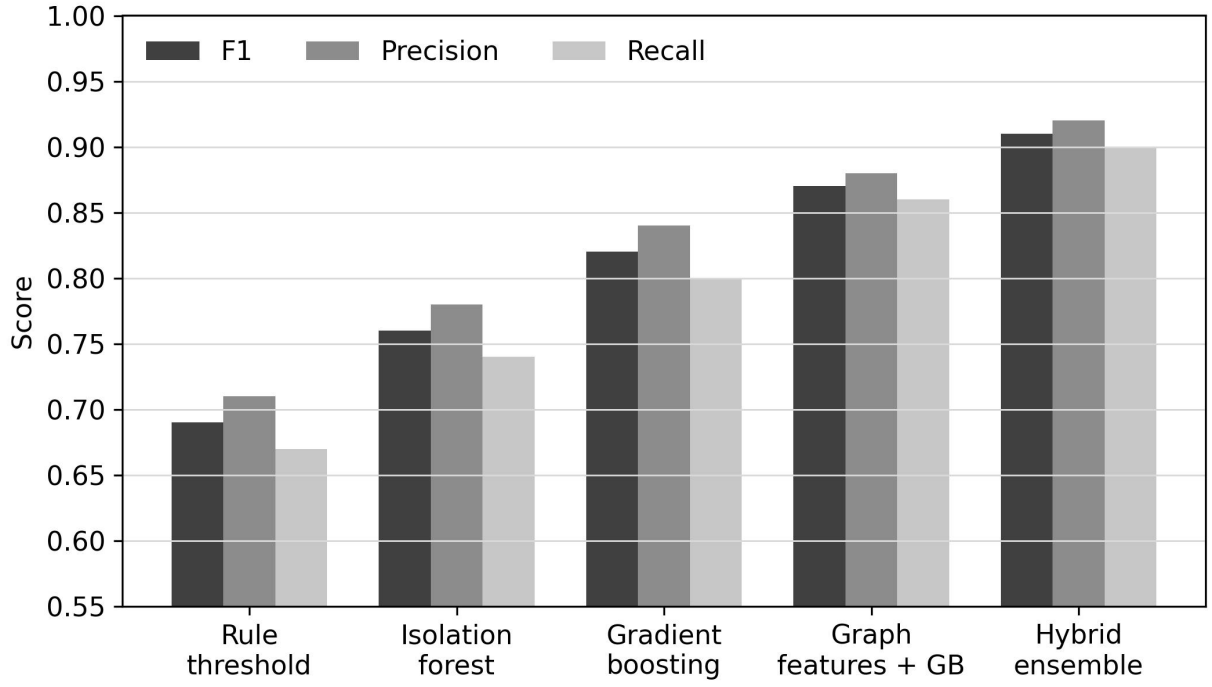

Figure 2. Detection performance across anomaly analytics configurations.

Figure 2 shows that richer feature integration improves detection performance. The rule baseline performs reasonably on obvious no-action cases but struggles when copy attacks preserve correct final results. Isolation forest improves anomaly discovery but lacks attack specific discrimination. Supervised gradient boosting performs better when labels are available. The largest performance improvement occurs when graph features are added, because copied assertions and collusive acceptance leave relational traces that are not visible in tabular event features. The hybrid ensemble achieves the best balance of precision, recall, and F1 score.

Table III. Experimental benchmark for anomaly detection models.

Model	Precision	Recall	F1	False positive rate	Best-detected class	Main limitation
Rule threshold	0.71	0.67	0.69	0.083	No-action	Rigid thresholds fail under adaptive timing
Isolation forest	0.78	0.74	0.76	0.061	Free-riding	Weak class explanation
Gradient boosting	0.84	0.80	0.82	0.045	Delay abuse	Limited relational reasoning
Graph features + GB	0.88	0.86	0.87	0.034	Copy attack	Requires stable identity graph
Hybrid ensemble	0.92	0.90	0.91	0.026	Copy and no-action	Higher governance and calibration burden

V. RESULTS AND INTERPRETATION

The benchmark suggests that anomaly analytics for Layer-2 rollups should not be treated as a generic fraud-detection task. The distinctive feature of this domain is that an attack may coexist with a correct final result. This is why simple outcome labels are insufficient. A model must examine whether the pathway to the result contains evidence of independent computation. In the simulation, free-riding is often visible through unusually short response times, reward seeking acceptance, and missing proof signals. Copy attacks are visible through ordering, similarity, and graph proximity. No-action behavior is visible through silence patterns and repeated timeout exposure (Lu and Xu, 2019；Zhang et al., 2019；Ahmed et al., 2016；He et al., 2016；Norta, 2015).

The hybrid model improves recall because it combines three perspectives. Supervised learning captures nonlinear combinations of timing, gas, and proof variables. Graph scoring captures relational patterns among managers. Rule constraints preserve domain knowledge, such as the requirement that acceptance should be accompanied by evidence of actual computation in stronger

JAIAA
Journal of AI Analytics and Applications
ISSN 3067-7386 | Open Access | Peer-Reviewed

RESEARCH ARTICLE
Vol. 4, No. 1 (2026), pp. 27–39
DOI: 10.63646/jaiaa.2026.040103

replicated-computation protocols. This combination is particularly useful for copy attacks, where a purely statistical model may misclassify a copied correct result as normal because the final output matches the honest computation (Lu, 2018).

Attack-specific recall reveals the most important operational lesson. Free-riding recall reaches 0.89 under the hybrid model, while copy-attack recall reaches 0.93 and no-action recall reaches 0.91. The lower recall for delay abuse, at 0.86, reflects the difficulty of separating malicious delay from legitimate challenge behavior. In a real protocol, this distinction should not be made by AI alone. Challenge quality, dispute outcome, historical pattern, and governance review should be combined before imposing penalties (Zheng et al., 2018).

The false-positive rate is also significant. A 2.6% false-positive rate may be acceptable for generating audit tickets, but it may be too high for automatic slashing or exclusion. Therefore, the paper recommends a tiered response structure. Low-risk anomalies should create monitoring labels. Medium-risk anomalies should require additional proof sampling. High-risk repeated anomalies should trigger governance review, temporary reward holdback, or enhanced dispute requirements. The analytics layer should support human and protocol governance rather than operate as an unchecked punitive mechanism (Yli-Huumo et al., 2016； Schär, 2021； Buczak and Guven, 2016； LeCun et al., 2015； Wust and Gervais, 2018).

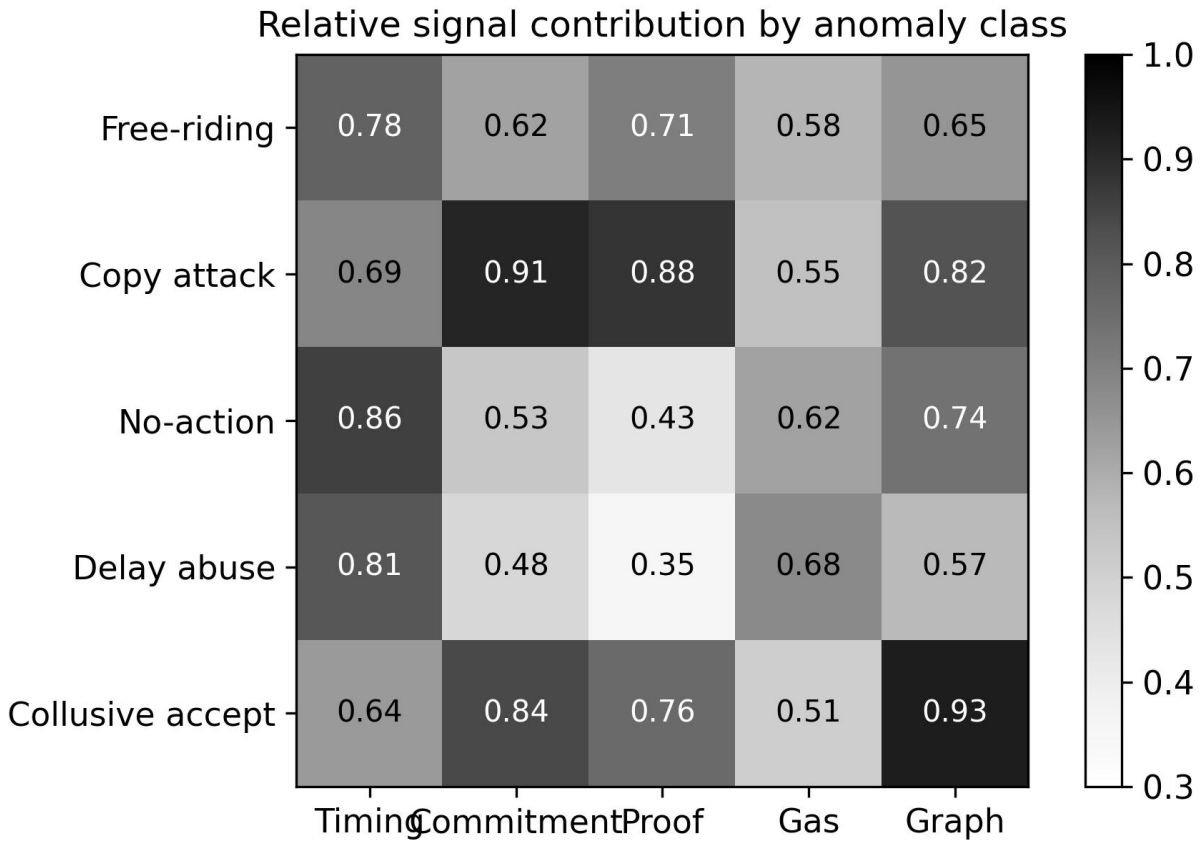

Figure 3. Relative signal contribution by anomaly class in the hybrid analytics model.

Figure 3 illustrates why different anomaly classes require different evidence sources. No-action behavior depends heavily on timing and graph signals because silence is a behavioral pattern rather than a proof inconsistency. Copy attacks depend most strongly on commitment, proof, and graph features because copying leaves similarity and ordering traces. Collusive acceptance depends most strongly on graph structure. This result supports a multi-view analytics strategy rather than a single universal anomaly score.

VI. GOVERNANCE, PRIVACY, AND DEPLOYMENT IMPLICATIONS

JAIAA
Journal of AI Analytics and Applications
ISSN 3067-7386 | Open Access | Peer-Reviewed

RESEARCH ARTICLE
Vol. 4, No. 1 (2026), pp. 27–39
DOI: 10.63646/jaiaa.2026.040103

AI-driven anomaly analytics must be deployed carefully in Layer-2 systems because monitoring can itself create privacy and centralization risks. The framework should not require public disclosure of private computation traces. It should use commitments, sampled proof evidence, and metadata whenever possible. When richer off-chain telemetry is needed, access should be limited to auditors or decentralized monitoring committees operating under transparent rules. This design respects the outsider privacy objective emphasized in privacy-preserving rollup research while still creating accountability for managers (Androulaki et al., 2018).

Governance integration should be explicit. A risk score has no meaning unless it maps to a response. The proposed response ladder includes observation, warning, proof escalation, reward holdback, temporary manager review, and protocol-level amendment. Each step should have due-process safeguards. For example, a manager flagged for no-action behavior should have an opportunity to provide delayed proof evidence if network conditions affected the event. A manager repeatedly linked to copy attacks, by contrast, may be required to submit stronger independent evidence or face exclusion from reward distribution (Dinh et al., 2017).

The framework is also relevant to protocol design. If analytics consistently identifies no-action behavior under a particular reward scheme, then the problem may not be model accuracy but incentive design. In that case, the protocol should require acceptance votes to include randomly sampled proof evidence or commitment openings. This aligns with the formal insight that accepting without computing should be treated as detectable deviation. AI can reveal the prevalence and distribution of such behavior, but the permanent fix may require protocol modification (Xu et al., 2016； Werner et al., 2022； Sommer and Paxson, 2010； Silver et al., 2016).

Decentralization is another concern. A monitoring system controlled by a single operator could become a soft referee, undermining the trust-minimized ethos of rollups. A better design is plural monitoring: independent analytics nodes compute risk scores from common public or permissioned evidence, and governance acts only when scores converge or when cryptographic evidence supports the alert. This approach treats AI as a decision-support mechanism, not as an ultimate authority (Casino et al., 2019) (Douceur, 2002).

Table IV. Governance response ladder for anomaly analytics in Layer-2 rollups.

Risk level	Trigger condition	Recommended action	Human review	Protocol safeguard
Low	Single weak signal or rare timing deviation	Record observation label	No immediate review	No penalty
Moderate	Repeated weak evidence across sessions	Request additional proof sampling	Optional audit	Reward remains available
High	Strong copy or no-action score with corroborating evidence	Hold reward and open audit queue	Required review	Temporary proof escalation
Critical	Repeated high-risk behavior or collusive cluster	Suspend manager eligibility pending review	Governance committee review	Transparent appeal and evidence log

VII. DISCUSSION: RELATION TO FORMAL VERIFICATION AND ROLLUP DESIGN

The proposed framework should be understood as complementary to formal verification. Formal security models define ideal behavior and prove whether a protocol realizes that behavior under specified assumptions. AI analytics observes deployed behavior and estimates whether participants are acting consistently with those assumptions. The two approaches answer different questions. Formal analysis asks whether a protocol can be secure. Behavioral analytics asks whether the live protocol environment is behaving as though the security assumptions are being respected (Kshetri, 2017).

This distinction matters for Layer-2 smart contracts because the attack surface is socio-technical. A rollup protocol includes smart contracts, off-chain managers, sequencers, validators, rewards, deposits, network timing, and governance rules. A cryptographic proof may ensure that a specific computation step is valid, but it does not by itself reveal whether reward design is encouraging lazy verification across thousands of sessions. Conversely, an anomaly model may flag suspicious manager behavior, but it cannot prove correctness in the cryptographic sense. Trustworthy rollup operations require both layers (Saberi et al., 2019； Qin et al., 2021； Ribeiro et al., 2016； Sarker, 2021).

The study also shows why final correctness is an incomplete metric. A protocol can output correct results while still rewarding non-computation. Over time, this creates a negative learning effect: managers discover that not computing is profitable as long as another party computes. This is the exact pathway through which free-riding becomes a systemic integrity risk. Monitoring systems should therefore evaluate contribution evidence, not only final output agreement (Treiblmaier, 2018).

Finally, the framework supports future research on reactive smart contracts, cross-rollup monitoring, and privacy-preserving telemetry. Reactive contracts create additional complexity because inputs arrive during execution, making independent trace

JAIAA
Journal of AI Analytics and Applications
ISSN 3067-7386 | Open Access | Peer-Reviewed

RESEARCH ARTICLE
Vol. 4, No. 1 (2026), pp. 27–39
DOI: 10.63646/jaiaa.2026.040103

comparison harder. Cross-rollup monitoring introduces heterogeneous event schemas and different dispute windows. Privacy-preserving telemetry will require commitments, secure aggregation, and possibly zero-knowledge attestations that prove monitoring features without revealing sensitive computation states (Frizzo-Barker et al., 2020).

VIII. IMPLEMENTATION BLUEPRINT AND PRACTICAL DATA ANALYSIS

A deployable anomaly analytics system for rollup protocols should begin with a minimal event contract rather than a complex machine learning platform. The monitoring layer needs a stable schema for assertions, commitments, votes, challenge windows, proof fragments, gas records, and dispute outcomes. Once these data elements are standardized, feature pipelines can compute manager-level baselines, pairwise similarity, delayed response patterns, and proof-availability scores without exposing private computation traces.

The practical implementation also requires a governance-aware data pipeline. Raw telemetry should be stored with tamper-evident hashes, while sensitive operational fields should be transformed into aggregate indicators before model training. This design keeps the analytics layer compatible with outsider privacy and reduces the risk that security monitoring becomes a new source of information leakage. The recommended deployment path is therefore incremental: schema stabilization, offline benchmarking, shadow-mode monitoring, human-reviewed alerting, and only then limited automatic reward holdback under explicit protocol rules.

IX. LIMITATIONS AND FUTURE RESEARCH

This article has several limitations. First, the benchmark dataset is simulated. It is useful for controlled evaluation but cannot capture every feature of real rollup traffic, adversarial adaptation, network congestion, or implementation-specific behavior. Future work should test the framework on open rollup event logs, permissioned validator telemetry, and controlled testnet deployments. A public benchmark for Layer-2 anomaly analytics would accelerate reproducible comparison across models (Li et al., 2020； Daian et al., 2020； Kipf and Welling, 2017； Miers et al., 2013).

Second, the model does not claim cryptographic soundness. A high anomaly score is not a proof that a manager cheated. It is a probabilistic signal that should trigger additional evidence collection. Future research should integrate machine learning with formal proof obligations, for example by using AI to select which sessions require stronger Merkle-proof sampling or which manager pairs should enter enhanced dispute verification (Conti et al., 2018).

Third, adaptive adversaries may learn to imitate normal timing and proof behavior. This risk is common in fraud analytics. Countermeasures include rotating random proof indices, adversarial training, concealed monitoring thresholds, and multi-party analytics. However, concealed thresholds must be balanced against transparency and governance fairness. Protocol participants should understand the categories of behavior that are prohibited even if exact risk-model parameters remain protected (Gervais et al., 2016).

Fourth, the framework focuses on non-reactive computations and replicated rollup settings. Reactive smart contracts, zero-knowledge rollups, validity proofs, decentralized sequencers, and shared rollup ecosystems may require different feature sets. Future work should extend the taxonomy to hybrid systems where optimistic dispute logic and validity proofs coexist (Meiklejohn et al., 2013； Cong and He, 2019； Vaswani et al., 2017； Ben-Sasson et al., 2014).

X. CONCLUSION

Layer-2 smart-contract protocols need more than final result correctness. They need operational assurance that the participants rewarded for verification actually performed the work required by the protocol. Inspired by formal analysis of Arbitrum0 and vArbitrum, this paper developed an AI-driven anomaly analytics framework for detecting free-riding, copy attacks, and no-action behaviors in rollup computation. The framework translates formal security concerns into observable feature groups, evaluates multiple detection models on a simulated protocol-event benchmark, and proposes a governance response ladder for practical deployment.Table V translates the conceptual framework into a deployable roadmap. The first stage is instrumentation, because no model can compensate for missing or inconsistent protocol events. The second stage is lightweight screening, which provides immediate value even before a full training dataset exists. The third stage introduces behavioral learning. The fourth stage connects model outputs to evidence escalation rather than automatic punishment. The fifth stage uses confirmed audits to improve the next

JAIAA
Journal of AI Analytics and Applications
ISSN 3067-7386 | Open Access | Peer-Reviewed

RESEARCH ARTICLE
Vol. 4, No. 1 (2026), pp. 27–39
DOI: 10.63646/jaiaa.2026.040103

model version. This roadmap avoids the common mistake of treating AI as a plug-in classifier detached from protocol governance.An implementation should also include model-risk controls. Drift detection is essential because adversaries adapt. A copy attacker may initially wait several seconds before copying and later learn to randomize the delay. A no-action manager may alternate between silence and minimal proof submission. The monitoring system should therefore track feature distribution shifts, class-recall changes, calibration error, and unexplained increases in audit reversal. When model drift is detected, the safest response is not immediate retraining alone, but temporary tightening of proof requirements for high-risk sessions.The broader implication is that AI analytics should make Layer-2 security more measurable. Formal models define properties such as correctness, privacy, and deviation detection. Protocol implementations define events and proofs. AI analytics connects these abstract properties to daily evidence: who computed, who copied, who stayed silent, who challenged constructively, and who repeatedly received rewards without independent contribution. This bridge between theory and operations is where anomaly analytics can make the strongest contribution to trustworthy smart-contract ecosystems.Another practical issue is explainability. Rollup governance participants may not accept a model that simply reports a risk score. The system should provide a compact explanation for each alert, such as abnormal acceptance latency, missing proof evidence, repeated agreement with a known source manager, or a sudden change in challenge behavior. Explanations should be stored with the alert record so that later dispute review can reconstruct why the model acted. This requirement is especially important when a manager is pseudonymous, because reputation damage can occur even without a legal identity.The model should also support counterfactual review. For example, governance may ask whether a manager would still be flagged if network latency were adjusted to the median condition for that block interval, or whether the risk score depends mainly on a single graph feature. Counterfactual review does not eliminate uncertainty, but it reduces arbitrary enforcement. It also makes the analytics layer more useful for protocol design, because repeated counterfactual patterns can reveal whether the problem is a dishonest manager, a weak reward rule, or a poorly calibrated dispute window.A production deployment should maintain a separation between raw evidence, derived features, model outputs, and governance decisions. Raw evidence includes protocol events and cryptographic commitments. Derived features include timing, proof coverage, and graph metrics. Model outputs include calibrated probabilities and class labels. Governance decisions include warnings, reward holds, or manager suspension. Keeping these layers separate improves auditability and allows a protocol community to change one layer without rewriting the entire monitoring stack.From a systems perspective, storage design matters. Raw event logs may be large, but most anomaly features are compact. A practical archive can store full evidence for high-risk sessions, sampled evidence for ordinary sessions, and aggregate statistics for long-term trend analysis. This reduces storage cost while preserving enough information for dispute review. It also supports privacy, because detailed trace-adjacent metadata is retained only when justified by risk or audit requirements.The approach is also useful for comparative protocol evaluation. Designers can test how different reward schemes, dispute windows, proof-sampling rules, and manager-selection policies affect anomaly prevalence. If a reward scheme creates high free-riding risk in simulation, the protocol can be redesigned before deployment. If a proof-sampling rule reduces no-action behavior but increases cost, governance can choose a cost-risk balance transparently. AI analytics therefore becomes an experimental tool for mechanism design, not only a monitoring tool for deployed networks.Finally, anomaly analytics should be evaluated under adversarial adaptation. A static benchmark may overstate performance if attackers learn the monitoring logic. Future testnets should include red-team scenarios in which managers deliberately randomize timing, copy only partial evidence, alternate between honest and lazy behavior, or form temporary collusive groups. These scenarios would provide more realistic measurements of resilience and would help determine whether hybrid models remain reliable when adversaries respond strategically to detection incentives (Tsankov et al., 2018；Easley et al., 2019；Devlin et al., 2019；Goldreich et al., 1989).

The central conclusion is that trustworthy rollup operations require an integration of formal methods, cryptographic evidence, AI analytics, and accountable governance. Formal models define the security target. Protocol design embeds proof and dispute mechanisms. AI analytics monitors behavioral deviations at scale. Governance converts evidence into proportionate responses. When these layers are aligned, Layer-2 smart contracts can preserve scalability benefits while reducing the hidden incentive failures that threaten replicated verification (Nikolic et al., 2018).

DECLARATIONS

Conflicts of interest: The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this manuscript.

Data availability: The experimental dataset used in this article is a simulated benchmark generated for methodological evaluation. No proprietary blockchain dataset is redistributed. Aggregated simulation settings and performance summaries are available from the corresponding author upon reasonable request.

Funding: This research received no external funding.

Ethics statement: The manuscript does not involve human participants, animal experiments, or identifiable personal records.

ABOUT THE AUTHORS

Daniel Osei is affiliated with the University of Ghana, Ghana. His research focuses on blockchain systems, distributed computing, and applied AI for trustworthy digital infrastructure.

Miriam Mensah is a researcher at Ghana Communication Technology University, Ghana. Her interests include cyber-physical security, intelligent information systems, and analytics for decentralized platforms.

Kwame Boateng is affiliated with Accra Technical University, Ghana. His research addresses smart-contract security, anomaly detection, and governance mechanisms for emerging computing systems.

REFERENCES

Zheng, Z., Xie, S., Dai, H. N., Chen, X., & Wang, H. (2018). Blockchain challenges and opportunities: A survey. International Journal of Web and Grid Services, 14(4), 352-375. https://doi.org/10.1504/IJWGS.2018.095647

Yli-Huumo, J., Ko, D., Choi, S., Park, S., & Smolander, K. (2016). Where is current research on blockchain technology? A systematic review. PLOS ONE, 11(10), e0163477. https://doi.org/10.1371/journal.pone.0163477

Androulaki, E., Barger, A., Bortnikov, V., Cachin, C., Christidis, K., De Caro, A., et al. (2018). Hyperledger Fabric: A distributed operating system for permissioned blockchains. Proceedings of the Thirteenth EuroSys Conference, 1-15. https://doi.org/10.1145/3190508.3190538

Yang, L., Hou, Q., Zhu, X., Lu, Y., & Xu, L. D. (2025). Potential of large language models in blockchain-based supply chain finance. Enterprise Information Systems, 19(11), 2541199. https://doi.org/10.1080/17517575.2025.2541199

Dinh, T. T. A., Wang, J., Chen, G., Liu, R., Ooi, B. C., & Tan, K. L. (2017). Blockbench: A framework for analyzing private blockchains. Proceedings of the ACM International Conference on Management of Data, 1085-1100. https://doi.org/10.1145/3035918.3064033

Xu, X., Pautasso, C., Zhu, L., Gramoli, V., Ponomarev, A., Tran, A. B., & Chen, S. (2016). The blockchain as a software connector. 2016 13th Working IEEE/IFIP Conference on Software Architecture, 182-191. https://doi.org/10.1109/WICSA.2016.21

Casino, F., Dasaklis, T. K., & Patsakis, C. (2019). A systematic literature review of blockchain-based applications: Current status, classification and open issues. Telematics and Informatics, 36, 55-81. https://doi.org/10.1016/j.tele.2018.11.006

Zhang, H., & Lu, Y. (2025). Web 3.0: Applications, opportunities and challenges in the next internet generation. Systems Research and Behavioral Science, 42(4), 996-1015. https://doi.org/10.1002/sres.3151

Kshetri, N. (2017). Can blockchain strengthen the Internet of Things? IT Professional, 19(4), 68-72. https://doi.org/10.1109/MITP.2017.3051335

Saberi, S., Kouhizadeh, M., Sarkis, J., & Shen, L. (2019). Blockchain technology and its relationships to sustainable supply chain management. International Journal of Production Research, 57(7), 2117-2135. https://doi.org/10.1080/00207543.2018.1533261

Treiblmaier, H. (2018). The impact of blockchain on the supply chain: A theory-based research framework. Supply Chain Management, 23(6), 545-559. https://doi.org/10.1108/SCM-01-2018-0029

Kou, G., & Lu, Y. (2025). FinTech: A literature review of emerging financial technologies and applications. Financial Innovation, 11(1), 1-34. https://doi.org/10.1186/s40854-024-00668-6

Frizzo-Barker, J., Chow-White, P. A., Adams, P. R., Mentanko, J., Ha, D., & Green, S. (2020). Blockchain as a disruptive technology for business: A systematic review. International Journal of Information Management, 51, 102029. https://doi.org/10.1016/j.ijinfomgt.2019.10.014

Li, X., Jiang, P., Chen, T., Luo, X., & Wen, Q. (2020). A survey on the security of blockchain systems. Future Generation Computer Systems, 107, 841-853. https://doi.org/10.1016/j.future.2017.08.020

Conti, M., Kumar, E. S., Lal, C., & Ruj, S. (2018). A survey on security and privacy issues of Bitcoin. IEEE Communications Surveys & Tutorials, 20(4), 3416-3452. https://doi.org/10.1109/COMST.2018.2842460

Lu, Y. (2025). The current status and developing trends of Industry 4.0: A review. Information Systems Frontiers, 27(1), 215-234. https://doi.org/10.1007/s10796-021-10221-w

Gervais, A., Karame, G. O., Wust, K., Glykantzis, V., Ritzdorf, H., & Capkun, S. (2016). On the security and performance of proof of work blockchains. Proceedings of the ACM SIGSAC Conference on Computer and Communications Security, 3-16. https://doi.org/10.1145/2976749.2978341

Meiklejohn, S., Pomarole, M., Jordan, G., Levchenko, K., McCoy, D., Voelker, G. M., & Savage, S. (2013). A fistful of bitcoins: Characterizing payments among men with no names. Proceedings of the Internet Measurement Conference, 127-140. https://doi.org/10.1145/2504730.2504747

Atzei, N., Bartoletti, M., & Cimoli, T. (2017). A survey of attacks on Ethereum smart contracts. Principles of Security and Trust, 164-186. https://doi.org/10.1007/978-3-662-54455-6_8

Wu, H. P., Liu, Z., Dong, H. Y., Lu, Y., & Xu, L. D. (2025). Revolutionizing internal auditing: Harnessing the power of blockchain. Enterprise Information Systems, 19(1-2), 2448003. https://doi.org/10.1080/17517575.2024.2448003

Luu, L., Chu, D. H., Olickel, H., Saxena, P., & Hobor, A. (2016). Making smart contracts smarter. Proceedings of the ACM SIGSAC Conference on Computer and Communications Security, 254-269. https://doi.org/10.1145/2976749.2978309

Tsankov, P., Dan, A., Drachsler-Cohen, D., Gervais, A., Buenzli, F., & Vechev, M. (2018). Securify: Practical security analysis of smart contracts. Proceedings of the ACM SIGSAC Conference on Computer and Communications Security, 67-82. https://doi.org/10.1145/3243734.3243780

Nikolic, I., Kolluri, A., Sergey, I., Saxena, P., & Hobor, A. (2018). Finding the greedy, prodigal, and suicidal contracts at scale. Proceedings of the 34th Annual Computer Security Applications Conference, 653-663. https://doi.org/10.1145/3274694.3274743

Lu, W., Lu, Y., Li, J., Sigov, A., Ratkin, L., & Ivanov, L. A. (2024). Quantum machine learning: Classifications, challenges, and solutions. Journal of Industrial Information Integration, 42, 100736. https://doi.org/10.1016/j.jii.2024.100736

Chen, T., Li, X., Luo, X., & Zhang, X. (2017). Under-optimized smart contracts devour your money. IEEE 24th International Conference on Software Analysis, Evolution and Reengineering, 442-446. https://doi.org/10.1109/SANER.2017.7884650

Chen, W., Zheng, Z., Cui, J., Ngai, E., Zheng, P., & Zhou, Y. (2018). Detecting Ponzi schemes on Ethereum: Towards healthier blockchain technology. WWW 2018 Companion, 1409-1418. https://doi.org/10.1145/3184558.3191616

Bartoletti, M., & Pompianu, L. (2017). An empirical analysis of smart contracts: Platforms, applications, and design patterns. Financial Cryptography and Data Security, 494-509. https://doi.org/10.1007/978-3-319-70278-0_31

Lu, Y., & Yang, J. (2024). Quantum financing system: A survey on quantum algorithms, potential scenarios and open research issues. Journal of Industrial Information Integration, 41, 100663. https://doi.org/10.1016/j.jii.2024.100663

Rouhani, S., & Deters, R. (2019). Security, performance, and applications of smart contracts: A systematic survey. IEEE Access, 7, 50759-50779. https://doi.org/10.1109/ACCESS.2019.2911031

Wang, W., Hoang, D. T., Hu, P., Xiong, Z., Niyato, D., Wang, P., et al. (2019). A survey on consensus mechanisms and mining strategy management in blockchain networks. IEEE Access, 7, 22328-22370. https://doi.org/10.1109/ACCESS.2019.2896108

Zhang, R., Xue, R., & Liu, L. (2019). Security and privacy on blockchain. ACM Computing Surveys, 52(3), 1-34. https://doi.org/10.1145/3316481

Xu, R., Zhu, J., Yang, L., Lu, Y., & Xu, L. D. (2024). Decentralized finance (DeFi): A paradigm shift in the FinTech. Enterprise Information Systems, 18(9), 2397630. https://doi.org/10.1080/17517575.2024.2397630

Schar, F. (2021). Decentralized finance: On blockchain- and smart-contract-based financial markets. Federal Reserve Bank of St. Louis Review, 103(2), 153-174. https://doi.org/10.20955/r.103.153-74

Werner, S. M., Perez, D., Gudgeon, L., Klages-Mundt, A., Harz, D., & Knottenbelt, W. J. (2022). SoK: Decentralized finance (DeFi). Proceedings of the 4th ACM Conference on Advances in Financial Technologies, 30-46. https://doi.org/10.1145/3558535.3559780

Qin, K., Zhou, L., Afonin, Y., Lazzaretti, L., & Gervais, A. (2021). CeFi vs. DeFi: Comparing centralized to decentralized finance. arXiv. https://doi.org/10.48550/arXiv.2106.08157

Chen, Y., Lu, Y., Bulysheva, L., & Kataev, M. Y. (2024). Applications of blockchain in Industry 4.0: A review. Information Systems Frontiers, 26(5), 1715-1729. https://doi.org/10.1007/s10796-022-10248-7

Daian, P., Goldfeder, S., Kell, T., Li, Y., Zhao, X., Bentov, I., et al. (2020). Flash Boys 2.0: Frontrunning in decentralized exchanges, miner extractable value, and consensus instability. IEEE Symposium on Security and Privacy, 910-927. https://doi.org/10.1109/SP40000.2020.00040

Cong, L. W., & He, Z. (2019). Blockchain disruption and smart contracts. Review of Financial Studies, 32(5), 1754-1797. https://doi.org/10.1093/rfs/hhz007

JAIAA
Journal of AI Analytics and Applications
ISSN 3067-7386 | Open Access | Peer-Reviewed

RESEARCH ARTICLE
Vol. 4, No. 1 (2026), pp. 27–39
DOI: 10.63646/jaiaa.2026.040103

Douceur, J. R. (2002). The Sybil attack. Peer-to-Peer Systems, 251-260. https://doi.org/10.1007/3-540-45748-8_24

Easley, D., O'Hara, M., & Basu, S. (2019). From mining to markets: The evolution of Bitcoin transaction fees. Journal of Financial Economics, 134(1), 91-109. https://doi.org/10.1016/j.jfineco.2019.03.004

Lu, Y., Sigov, A. S., Ratkin, L., Ivanov, L. A., & Zuo, M. (2023). Quantum computing and industrial information integration: A review. Journal of Industrial Information Integration, 35, 100511. https://doi.org/10.1016/j.jii.2023.100511

Breiman, L. (2001). Random forests. Machine Learning, 45, 5-32. https://doi.org/10.1023/A:1010933404324

Chen, T., & Guestrin, C. (2016). XGBoost: A scalable tree boosting system. Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 785-794. https://doi.org/10.1145/2939672.2939785

Liu, F. T., Ting, K. M., & Zhou, Z. H. (2008). Isolation forest. 2008 Eighth IEEE International Conference on Data Mining, 413-422. https://doi.org/10.1109/ICDM.2008.17

Lu, Y. (2022). Implementing blockchain in information systems: A review. Enterprise Information Systems, 16(12), 1876-1907. https://doi.org/10.1080/17517575.2021.2008513

Chandola, V., Banerjee, A., & Kumar, V. (2009). Anomaly detection: A survey. ACM Computing Surveys, 41(3), 1-58. https://doi.org/10.1145/1541880.1541882

Akoglu, L., Tong, H., & Koutra, D. (2015). Graph based anomaly detection and description: A survey. Data Mining and Knowledge Discovery, 29, 626-688. https://doi.org/10.1007/s10618-014-0363-y

Ahmed, M., Mahmood, A. N., & Hu, J. (2016). A survey of network anomaly detection techniques. Journal of Network and Computer Applications, 60, 19-31. https://doi.org/10.1016/j.jnca.2015.11.016

Zheng, X. R., & Lu, Y. (2022). Blockchain technology: Recent research and future trend. Enterprise Information Systems, 16(12), 1939895. https://doi.org/10.1080/17517575.2021.1939895

Buczak, A. L., & Guven, E. (2016). A survey of data mining and machine learning methods for cyber security intrusion detection. IEEE Communications Surveys & Tutorials, 18(2), 1153-1176. https://doi.org/10.1109/COMST.2015.2494502

Sommer, R., & Paxson, V. (2010). Outside the closed world: On using machine learning for network intrusion detection. IEEE Symposium on Security and Privacy, 305-316. https://doi.org/10.1109/SP.2010.25

Ribeiro, M. T., Singh, S., & Guestrin, C. (2016). Why should I trust you? Explaining the predictions of any classifier. Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 1135-1144. https://doi.org/10.1145/2939672.2939778

Xu, L. D., Lu, Y., & Li, L. (2021). Embedding blockchain technology into IoT for security: A survey. IEEE Internet of Things Journal, 8(13), 10452-10473. https://doi.org/10.1109/JIOT.2021.3060508

Kipf, T. N., & Welling, M. (2017). Semi-supervised classification with graph convolutional networks. arXiv. https://doi.org/10.48550/arXiv.1609.02907

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., et al. (2017). Attention is all you need. arXiv. https://doi.org/10.48550/arXiv.1706.03762

Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2019). BERT: Pre-training of deep bidirectional transformers for language understanding. arXiv. https://doi.org/10.48550/arXiv.1810.04805

Zhang, C., & Lu, Y. (2021). Study on artificial intelligence: The state of the art and future prospects. Journal of Industrial Information Integration, 23, 100224. https://doi.org/10.1016/j.jii.2021.100224

Kingma, D. P., & Ba, J. (2015). Adam: A method for stochastic optimization. arXiv. https://doi.org/10.48550/arXiv.1412.6980

Mnih, V., Kavukcuoglu, K., Silver, D., Rusu, A. A., Veness, J., Bellemare, M. G., et al. (2015). Human-level control through deep reinforcement learning. Nature, 518, 529-533. https://doi.org/10.1038/nature14236

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., et al. (2014). Generative adversarial networks. arXiv. https://doi.org/10.48550/arXiv.1406.2661

Lu, Y. (2019). Artificial intelligence: A survey on evolution, models, applications and future trends. Journal of Management Analytics, 6(1), 1-29. https://doi.org/10.1080/23270012.2019.1570365

Shapley, L. S. (1953). A value for n-person games. Contributions to the Theory of Games, 2, 307-317. https://doi.org/10.1515/9781400881970-018

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., et al. (2011). Scikit-learn: Machine learning in Python. Journal of Machine Learning Research, 12, 2825-2830. https://doi.org/10.48550/arXiv.1201.0490

He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. IEEE Conference on Computer Vision and Pattern Recognition, 770-778. https://doi.org/10.1109/CVPR.2016.90

Lu, Y. (2019). The blockchain: State-of-the-art and research challenges. Journal of Industrial Information Integration, 15, 80-90. https://doi.org/10.1016/j.jii.2019.04.002

LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. Nature, 521, 436-444. https://doi.org/10.1038/nature14539

Silver, D., Huang, A., Maddison, C. J., Guez, A., Sifre, L., van den Driessche, G., et al. (2016). Mastering the game of Go with deep neural networks and tree search. Nature, 529, 484-489. https://doi.org/10.1038/nature16961

Sarker, I. H. (2021). Machine learning: Algorithms, real-world applications and research directions. SN Computer Science, 2, 160. https://doi.org/10.1007/s42979-021-00592-x

Lu, Y., & Xu, L. D. (2019). Internet of Things (IoT) cybersecurity research: A review of current research topics. IEEE Internet of Things Journal, 6(2), 2103-2115. https://doi.org/10.1109/JIOT.2018.2869847

Miers, I., Garman, C., Green, M., & Rubin, A. D. (2013). Zerocoin: Anonymous distributed e-cash from Bitcoin. IEEE Symposium on Security and Privacy, 397-411. https://doi.org/10.1109/SP.2013.34

Ben-Sasson, E., Chiesa, A., Garman, C., Green, M., Miers, I., Tromer, E., & Virza, M. (2014). Zerocash: Decentralized anonymous payments from Bitcoin. IEEE Symposium on Security and Privacy, 459-474. https://doi.org/10.1109/SP.2014.36

Goldreich, O., Micali, S., & Wigderson, A. (1989). How to prove all NP statements in zero-knowledge, and a methodology of cryptographic protocol design. Proceedings on Advances in Cryptology, 171-185. https://doi.org/10.1007/0-387-34805-0_17

Lu, Y. (2018). Blockchain and the related issues: A review of current research topics. Journal of Management Analytics, 5(4), 231-255. https://doi.org/10.1080/23270012.2018.1516523

Castro, M., & Liskov, B. (1999). Practical Byzantine fault tolerance. Proceedings of the Third Symposium on Operating Systems Design and Implementation, 173-186. https://doi.org/10.5555/296806.296824

Lamport, L. (1998). The part-time parliament. ACM Transactions on Computer Systems, 16(2), 133-169. https://doi.org/10.1145/279227.279229

Bonneau, J., Miller, A., Clark, J., Narayanan, A., Kroll, J. A., & Felten, E. W. (2015). SoK: Research perspectives and challenges for Bitcoin and cryptocurrencies. IEEE Symposium on Security and Privacy, 104-121. https://doi.org/10.1109/SP.2015.14

Croman, K., Decker, C., Eyal, I., Gencer, A. E., Juels, A., Kosba, A., et al. (2016). On scaling decentralized blockchains. Financial Cryptography and Data Security, 106-125. https://doi.org/10.1007/978-3-662-53357-4_8

Eyal, I., & Sirer, E. G. (2018). Majority is not enough: Bitcoin mining is vulnerable. Communications of the ACM, 61(7), 95-102. https://doi.org/10.1145/3212998

Norta, A. (2015). Creation of smart-contracting collaborations for decentralized autonomous organizations. Perspectives in Business Informatics Research, 3-17. https://doi.org/10.1007/978-3-319-21915-8_1

Wust, K., & Gervais, A. (2018). Do you need a blockchain? 2018 Crypto Valley Conference on Blockchain Technology, 45-54. https://doi.org/10.1109/CVCBT.2018.00011