International Journal of Recent Advances in Science and Technology, 2026; 12(1):25-32

DOI: 10.30750/ijarst.120105

Decentralized Finance and Traditional Banking: A Machine Learning-Based Risk Assessment Framework

Babaeva Asal

Senior Analyst at the Ministry of Development of Digital Technologies and Communications

Abstract

Decentralised finance (DeFi) protocols perform functions long reserved to banks, including lending, exchange and asset management, without the balance sheets, supervision and deposit insurance that structure banking risk. Risk assessment frameworks built for banks do not transfer directly, because DeFi adds risk factors that have no banking analogue, such as smart-contract code risk and oracle dependence, and removes others, such as regulatory capital. This paper develops a unified machine-learning risk assessment framework that scores both DeFi protocols and traditional banks on a common set of seven factors and examines it in a simulation of 1,200 entities. A gradient-boosting model attains an AUC of 0.83 across entity types. Permutation importance differs sharply by type: for banks, leverage (0.15) and governance (0.14) dominate, while code and oracle risk are irrelevant by construction; for DeFi protocols, leverage (0.06), collateral volatility (0.04), liquidity (0.04) and the DeFi-specific code and oracle risks (0.03 each) contribute more evenly, so that risk is distributed across more channels. The paper proposes a two-layer framework in which a shared factor model supports comparison and a type-specific layer captures the mechanisms unique to each, and discusses the implications for supervisors, for banks integrating DeFi services and for the risk governance of protocols themselves.

Keywords: decentralised finance; banking risk; machine learning; risk assessment; smart-contract risk; financial stability.

Introduction

Decentralised finance replaces intermediaries with smart contracts: lending pools match depositors and borrowers algorithmically, automated market makers provide exchange without order books, and yield protocols allocate assets by rule. The functions are familiar to banking, but the risk architecture is not. Banks hold capital against losses, are supervised, and offer insured deposits; DeFi protocols are over-collateralised but uninsured, governed by token holders rather than boards, and exposed to failures in code and in the price feeds (oracles) on which their rules depend [1], [2]. A series of protocol collapses has shown that these risks are material.

Risk assessment for the two systems has developed separately. Bank risk rests on decades of supervisory practice, with capital ratios, liquidity coverage and governance assessments as its core [3]. DeFi risk assessment is young and fragmented, drawing on code audits, on-chain metrics and community reputation. As banks begin to offer DeFi-connected products and supervisors begin to consider DeFi within the perimeter, a framework that assesses both on a comparable basis becomes necessary.

This paper develops such a framework using machine learning. It specifies a common factor set that spans both entity types, trains a single model on simulated failure outcomes, and examines how the drivers of failure differ between banks and protocols. The results inform a two-layer framework and its use by supervisors, banks and protocol governance.

Background

Banking risk

Bank failure risk is conventionally decomposed into credit, market, liquidity and operational risk, with leverage and liquidity as the principal quantitative indicators and governance as the principal qualitative one [3]. Supervisory frameworks set minimum capital and liquidity ratios and assess governance through examination. Regulatory coverage itself reduces failure risk through early intervention.

DeFi risk

DeFi protocols face analogues of leverage (collateralisation ratios), liquidity (pool depth and withdrawal dynamics) and governance (token-holder voting, admin keys), and two additional risks: code risk, the possibility that a smart contract contains an exploitable flaw, and oracle risk, the possibility that the external price feed is manipulated or fails [1], [2]. Collateral volatility matters more in DeFi because liquidations are automatic and can cascade. Regulatory coverage is minimal, so no supervisor intervenes before failure.

Machine learning in financial risk

Machine-learning classifiers are established in credit and bank-failure prediction [4], and their ability to learn interactions makes them suited to a setting in which the same factor, such as leverage, may matter differently across entity types. Permutation importance [5] computed within entity type reveals those differences.

Why a common framework

Separate frameworks for banks and protocols serve each community but fail at the boundary between them, which is where the risk of the coming decade lies: banks holding tokenised assets in protocols, protocols accepting bank-issued stablecoins as collateral, and supervisors asked to decide whether a protocol exposure should carry a capital charge. A common framework does not assume that the two kinds of entity fail for the same reasons; it assumes that their failure probabilities can be placed on one scale and that the reasons can be compared. Machine learning suits this task because a single model can learn that leverage interacts with regulatory coverage and that code risk matters only where it exists, without requiring separate specifications.

The Risk Assessment Framework

The framework has two layers. The shared layer scores all entities on seven factors: leverage, liquidity, governance quality, code risk, oracle risk, collateral volatility and regulatory coverage, each normalised to [0, 1], with code and oracle risk set to zero for banks. A single model maps the factors to failure probability, which places banks and protocols on one scale and allows a bank to compare a DeFi exposure with its other exposures. The type-specific layer interprets the shared score through the mechanisms relevant to each type: for banks, supervisory ratios and governance examination; for protocols, audit status, admin-key arrangements, oracle design and liquidation mechanics. The shared layer supports comparison; the type-specific layer supports action.

Simulation Design

The harness generates 1,200 entities, half DeFi protocols and half banks. Factor distributions differ by type: protocols have higher leverage and collateral volatility, lower liquidity and governance scores, positive code and oracle risk and near-zero regulatory coverage; banks the reverse. Failure follows

$$\text{logit } P(\text{failure}) = -4.2 + 3.0\,\text{lev} - 1.5\,\text{liq} - 1.0\,\text{gov} + 1.2\,\text{code} + 1.0\,\text{oracle} + 1.5\,\text{colvol} - 1.0\,\text{reg}, \quad (1)$$

with realised failure rates of 25.3% for protocols and 2.8% for banks. A gradient-boosting classifier [6] is trained on 70% of entities and evaluated on the remainder by AUC overall and by type; permutation importance is computed overall and within type. Parameters are design values encoding the risk mechanisms described in Section 2.

Results

Discrimination

The model attains an AUC of 0.83 across all entities, 0.71 within DeFi protocols and 0.78 within banks. The overall figure exceeds both within-type figures because entity type itself, through the factor distributions, separates high- and low-risk populations; the within-type figures measure the model's ability to rank entities of the same kind.

Risk-factor importance by entity type

Table 1 and Figure 1 report permutation importance within type. For banks, leverage (0.15) and governance (0.14) account for most of the model's discrimination, with collateral volatility (0.08) third and liquidity (0.02) modest; code and oracle risk are zero by construction. For DeFi protocols, leverage (0.06) leads but by less, followed by collateral volatility (0.04), liquidity (0.04), code risk (0.03), oracle risk (0.03) and governance (0.02). Risk in DeFi is spread

Table 1: Permutation importance (AUC decrease) of risk factors within entity type (360 held-out entities)

Risk factor	*Traditional banks*	*DeFi protocols*
Leverage	0.147	0.060
Liquidity	0.015	0.038
Governance quality	0.137	0.022
Code risk	0.000	0.028
Oracle risk	0.000	0.025
Collateral volatility	0.077	0.044
Regulatory coverage	−0.016	−0.007

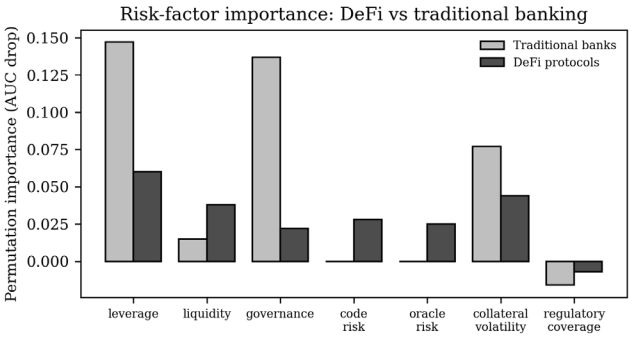


Figure 1: Permutation importance of risk factors for traditional banks and DeFi protocols

across more channels, none dominant, whereas bank risk concentrates in two.

Interpretation

The pattern reflects the architectures. In banking, leverage is the master variable because capital absorbs losses from every source, and governance determines whether leverage is managed prudently. In DeFi, over-collateralisation substitutes for capital but is undermined by collateral volatility and thin liquidity, and the protocol can fail through channels that have no banking analogue even when its collateralisation is sound. Regulatory coverage shows no within-type importance because it does not vary within type in the simulation; its effect appears in the gap between the two failure rates.

Sensitivity

The within-type importance pattern follows from the factor distributions and the common coefficients: a factor matters within a type in proportion to its variance there and its coefficient. Banks' low variance in collateral volatility and absence of code and oracle risk concentrate their importance in leverage and governance; protocols' wider variance across all factors spreads it. Changing the coefficients would re-weight the factors but not alter the qualitative contrast, which is that DeFi risk is multi-channel. The within-DeFi AUC of 0.71 is modest and would rise with factors not modelled here, such as dependency structure and audit recency.

Implications

For supervisors

A shared factor model lets supervisors place DeFi exposures on the same scale as bank exposures, which is the prerequisite for setting capital or limits against them. The type-specific layer indicates that supervision of DeFi cannot rely on the two levers, capital and governance examination, that dominate bank supervision; it must address code assurance, oracle robustness and liquidation design, which are engineering rather than financial controls.

For banks integrating DeFi

Banks offering DeFi-connected products inherit DeFi risk channels. Their risk functions should extend factor models to include code and oracle risk for protocol exposures, require audit and oracle-design evidence as conditions of integration, and recognise that diversification across protocols provides less protection than across banks when protocols share oracles or collateral assets.

For protocol governance

The distributed risk profile of DeFi implies that no single control suffices. Protocols should maintain audit programmes, redundant oracles, conservative collateral parameters calibrated to volatility, and governance arrangements that can respond to emerging risks without admin-key concentration. The framework provides a scorecard against which such measures can be assessed.

For research

The model's propositions, that factor importance differs by entity type in the directions shown and that a shared model can discriminate within both, are testable with protocol-failure datasets assembled from on-chain data and audit records, combined with bank-failure data. The main empirical challenge is the construction of comparable factor measures across types.

Contagion and systemic risk

The entity-level framework is the foundation for, not a substitute for, systemic assessment. DeFi protocols are composable: one protocol's tokens are another's collateral, and many share oracles. A failure in one can propagate through liquidations and oracle manipulation to others in minutes, faster than any supervisory intervention. The type-specific layer should therefore record each protocol's dependencies, and systemic assessment should simulate shocks across the dependency graph. Banks are also interconnected, but through slower channels and with central-bank backstops; the speed difference is itself a risk characteristic that the framework should flag.

Data for the shared layer

Bank factors come from regulatory filings; protocol factors must be constructed from on-chain data (collateralisation, pool depth, governance token distribution) and off-chain evidence (audits, oracle architecture). Constructing comparable measures is the main practical obstacle to the shared layer, and standardised protocol disclosure, analogous to bank reporting, would remove it. Until then, the framework can operate with proxies, provided their limitations are recorded.

Governance of the model

A risk model that informs capital charges or integration decisions is itself a supervised model in the regulatory sense and should be validated, documented and monitored accordingly. Its failure modes include training on a short history of protocol failures that over-represents particular exploit types, and drift as protocol designs change. Periodic retraining, out-of-time validation and the retention of the type-specific layer as an expert check on model outputs are the appropriate safeguards.

Limitations

Failure is generated from a single logistic model with independent factors, and the factor distributions by type are stylised. Contagion among protocols sharing oracles or collateral, which is central to DeFi systemic risk, is not modelled. The importance values are simulation outputs reflecting the design coefficients and are not estimates for any real population of banks or protocols.

Conclusion

A machine-learning framework that scores DeFi protocols and traditional banks on a common factor set discriminated failure risk within both types in simulation, and revealed that the drivers differ: bank risk concentrates in leverage and governance, while DeFi risk is distributed across leverage, collateral volatility, liquidity and the code and oracle channels unique to it. Supervisors, integrating banks and protocol governors should use a shared layer for comparison and a type-specific layer for control, recognising that DeFi requires engineering assurance alongside financial prudence.

References

1. Bhaskara Raju Rallabandi. (2024). 6G Cross-Layer Intent Engine: Automated Translation from Business Requirements to Network Policies and Real-Time Scheduling Controls. International Journal of Data Science and IoT Management System, 3(1), 44–48. https://doi.org/10.64751/ijdim.2024.v3.n1.pp44-48
2. Kumar, Ashok, and Arpit Jain. "Image smog restoration using oblique gradient profile prior and energy minimization." Frontiers of Computer Science 15, no. 6 (2021): 156706.
3. Bhaskara Raju Rallabandi. (2019). Comprehensive Design and Field Performance Assessment of Early 5G NSA Private Networks for Industrial IoT Campuses. International Journal of Engineering Research and Science & Technology, 15(1), 22–26. https://doi.org/10.62643/ijerst.2019.v15.n1.pp22-26
4. Bhanushali, B. (2023). Blockchain for trade finance: Enhancing cross-border payment transparency, reducing processing time, and mitigating fraud through distributed ledger systems. International Journal on Recent and Innovation Trends in Computing and Communication, 11(9), 5690–5697. https://www.ijritcc.org/index.php/ijritcc/article/view/11932
5. Khan, S. (2019). Sales force security compliance: An in-depth study of GDPR, HIPAA, and PCI-DSS enforcement in cloud-based CRM systems and their implications for global enterprises. International Journal of Advanced Research in Electrical, Electronics and Instrumentation

Engineering, 8(9), 2211–2219. https://doi.org/10.15662/IJAREEIE.2019.0809010

6. Ahuja, R. (2023). Implementation of AI-driven input validation in user interfaces for preventing injection attacks and ensuring data sanitation through adaptive filtering and pattern recognition. International Journal on Recent and Innovation Trends in Computing and Communication, 11(8), 884–891. https://www.ijritcc.org/index.php/ijritcc/article/view/11936
7. Rangappa, A. S. (2021). Database security in cloud environments: Exploring multi-tenant risks, data isolation challenges, and compliance mechanisms in hybrid and public clouds. International Journal on Recent and Innovation Trends in Computing and Communication, 9(2), 50–57. https://www.ijritcc.org/index.php/ijritcc/article/view/11921
8. Mahajan, R. A., Dey, R., Khan, M., Mohd Su'ud, M., Alam, M. M., & Jadhav, P. (2025). Enhancing personalization in IoT-based health monitoring via generative AI and transfer learning. Egyptian Informatics Journal, 32, Article 100788. https://doi.org/10.1016/j.eij.2025.100788
9. Dwivedi, D. (2020). Adoption of secure software updates in application lifecycle for preventing supply chain tampering and maintaining trust through cryptographically signed patches. Journal of Critical Reviews, 7(10), 6974–6980. https://doi.org/10.69980/jcr.v7:i10.13406
10. Harshitha, G., Kumar, S., Rani, S., & Jain, A. (2021, November). Cotton disease detection based on deep learning techniques. In 4th Smart Cities Symposium (SCS 2021) (Vol. 2021, pp. 496–501). IET.
11. Anchala, S. (2023). Integration of confidential computing in multi-cloud environments for enhancing data privacy and regulatory compliance through encrypted processing and isolated execution. International Journal on Recent and Innovation Trends in Computing and Communication, 11(9), 5698–5709. https://www.ijritcc.org/index.php/ijritcc/article/view/11933
12. Gupta, A. (2021). Deployment of advanced endpoint detection in remote work environments for preventing malware intrusions and data exfiltration through behavioral analytics and AI-driven responses. NeuroQuantology, 19(9), 1206–1224. https://doi.org/10.48047/nq.2021.19.9.NQ21199
13. Rangappa, A. S. (2020). Deployment of cross-platform code scanning tools in polyglot DevSecOps projects for uniform security testing and language-agnostic vulnerability coverage. Journal of Critical Reviews, 7(11), 5000–5006. https://doi.org/10.69980/jcr.v7:i11.13402
14. Mahadasu, S. S. (2024). Open RAN-based on-demand 6G agricultural networks: Adaptive edge infrastructure for autonomous farming systems. International Journal of Data Science and IoT Management System, 3(4), 96–102. https://doi.org/10.64751/ijdim.2024.v3.n4.1289
15. Bhaskara Raju Rallabandi. (2019). Enhancing 5G NSA Performance with Lightweight Network Data Analytics Loops. International Journal of Engineering Research and Science & Technology, 15(4), 57–61. https://doi.org/10.62643/ijerst.2019.v15.n4.pp57-61
16. Aggarwal, A. (2022). Cloud security in artificial intelligence workloads: Protecting model training, data confidentiality, and AI-generated insights in cloud-based machine learning pipelines. NeuroQuantology, 20(3), 1367–1382. https://doi.org/10.48047/nq.2022.20.3.NQ23004
17. Khan, S. (2022). The use of artificial intelligence in ESG (environmental, social, and governance) investing: A study on AI models for sustainability analytics in asset and wealth management. International Journal of Multidisciplinary and Scientific Emerging Research, 10(1), 431–439. https://doi.org/10.15662/IJMSERH.2022.1001046
18. Jain, Arpit, Adesh Kumar, and Sanjeev Sharma. "Comparative design and analysis of mesh, torus and ring NoC." Procedia Computer Science 48 (2015): 330–337.
19. Chatrath, A. (2024). The role of encryption and key management in cloud security: Balancing performance, scalability, and confidentiality in multi-tenant and distributed data storage systems. International Journal of Innovative Research in Science, Engineering and Technology, 13(7). https://doi.org/10.15680/IJIRSET.2024.1307187
20. Dwivedi, D. (2022). Application of AI-powered vulnerability scanners in legacy applications for

identifying unpatched security flaws and weak dependencies through semantic code analysis. International Journal on Recent and Innovation Trends in Computing and Communication, 10(12), 617–623. https://www.ijritcc.org/index.php/ijritcc/article/view/11918

21. Kumar, S., Jain, A., & Moparthi, N. R. (2022). Enhanced method of object tracing using extended Kalman filter via binary search algorithm.
22. Dwivedi, D. (2019). The use of homomorphic encryption in cloud security: Enabling secure data processing, confidential query execution, and privacy-preserving cloud applications. International Journal of Innovative Research in Science, Engineering and Technology, 8(4), 4690–4698. https://doi.org/10.15680/IJIRSET.2019.0804136
23. Talasila, D. (2023). Enhancing cloud security with intrusion detection and prevention systems (IDPS): Comparative evaluation of signature-based, anomaly-based, and AI-powered detection models. International Journal on Recent and Innovation Trends in Computing and Communication, 11(3), 828–836. https://www.ijritcc.org/index.php/ijritcc/article/view/11945
24. Mahadasu, S. S. (2025). Distributed AI-RAN inference placement for joint radio optimization and edge workload acceleration. International Journal of AI Electronics and Nexus Energy, 1(1), 39–44. https://doi.org/10.64751/7ep35m50
25. Khan, S. (2022). Sales force security in mobile and remote work environments: Addressing authentication challenges, secure synchronization, and device-level threats. International Journal of Innovative Research in Computer and Communication Engineering, 10(12), 8735–8743. https://doi.org/10.15680/IJIRCCE.2022.1012040
26. Jain, Jaishree, Arpit Jain, Saurabh Kumar Srivastava, Chaman Verma, Maria Simona Raboaca, and Zoltán Illés. "Improved security of e-healthcare images using hybridized robust zero-watermarking and hyper-chaotic system along with RSA." Mathematics 10, no. 7 (2022): 1071.
27. Bhanushali, B. (2024). Tokenization of real-world assets through block chain for improving liquidity, enabling fractional ownership, and reducing transaction costs in global financial markets. International Journal of Advanced Research in Education and Technology, 11(4), 1809–1816. https://doi.org/10.15680/IJARETY.2024.1104071
28. Chatrath, A. (2021). Integration of AI-powered forensic analysis in cloud breach investigations for accelerating root cause identification and evidence collection through log correlation techniques. International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering, 10(8). https://doi.org/10.15662/IJAREEIE.2021.1008051
29. Kumar, S., Sharma, K., Kumar, P. A., Jain, A., Bhagat, S. K., & Singh, P. (2024, September). An improved particle swarm approach for energy-aware location-aided routing in mobile ad-hoc network. In 2024 7th International Conference on Contemporary Computing and Informatics (IC3I) (Vol. 7, pp. 1119–1124). IEEE.
30. Mahadasu, S. S. (2025). AI-enhanced service management and orchestration for rApp and xApp lifecycle automation in Open RAN. Communications on Applied Nonlinear Analysis, 32(2s), 794–807. https://doi.org/10.52783/cana.v32.6716
31. Aggarwal, A. (2021). Implementation of machine learning in cloud SIEM solutions for enhancing threat detection accuracy and reducing alert fatigue through correlation-based anomaly detection. NeuroQuantology, 19(10). https://doi.org/10.48047/nq.2021.19.10.NQ21226
32. Ahuja, R. (2022). Application of static application security testing (SAST) in CI/CD pipelines for early detection of insecure coding practices through syntax tree analysis and custom rule sets. International Journal on Recent and Innovation Trends in Computing and Communication, 10(2), 80–87. https://www.ijritcc.org/index.php/ijritcc/article/view/11922
33. Jain, A. (2024, December). Energy-Efficient Signal Processing Architectures for IoT Networks Design. In 2024 International Conference on Emerging Technologies and Innovation for Sustainability (EmergIN) (pp. 362–367). IEEE.
34. Devulapalli, N. (2020). The evolution of database security from centralized architectures to

distributed cloud-native systems: A comparative and historical perspective. Turkish Online Journal of Qualitative Inquiry, 11(4), 2954–2963. https://doi.org/10.69980/j1064z97

35. Ahuja, R. (2023). The role of artificial intelligence and machine learning in DevSecOps: Leveraging predictive analytics, automated threat detection, and anomaly identification for secure software delivery pipelines. International Journal on Recent and Innovation Trends in Computing and Communication, 11(2), 381–389. https://www.ijritcc.org/index.php/ijritcc/article/view/11946
36. Devulapalli, N. (2020). Implementation of compliance-as-code in regulated DevSecOps environments for automating audit readiness and reducing human error through scripted control validation. Journal of Critical Reviews, 7(1), 2651–2657. https://doi.org/10.69980/jcr.v7:i1.13405
37. Anchala, S. (2020). Cloud security challenges in supply chain systems: Addressing vendor risks, data integrity, and third-party service provider vulnerabilities. Journal of Critical Reviews, 7(2). https://doi.org/10.69980/jcr.v7:i2.13401
38. Aggarwal, A. (2021). Utilization of continuous vulnerability scanning in DevSecOps pipelines for preemptive threat detection and secure code deployment through automated security checks. NeuroQuantology, 19(1), 427–442. https://doi.org/10.48047/nq.2021.19.1.NQ21055
39. Devulapalli, N. (2019). Post-quantum cryptography for database security: Preparing for the impact of quantum computing on encrypted data storage and secure queries. Journal of Critical Reviews, 6(1), 614–620. https://doi.org/10.69980/jcr.v6:i1.13396
40. Kaur, M., AlZubi, A. A., Jain, A., Singh, D., Yadav, V., & Alkhayyat, A. (2023). DSCNet: deep skip connections-based dense network for ALL diagnosis using peripheral blood smear images. Diagnostics, 13(17), 2752.
41. Talasila, D. (2020). Deployment of dynamic data masking in cloud-based applications for protecting sensitive information and enhancing privacy controls through contextual access filtering. Journal of Critical Reviews, 7(9), 4056–4062. https://doi.org/10.69980/jcr.v7:i9.13403
42. Anchala, S. (2023). Utilization of blockchain technology in cloud storage systems for enhancing data integrity and preventing unauthorized modifications through decentralized validation. International Journal on Recent and Innovation Trends in Computing and Communication, 11(5), 629–637. https://www.ijritcc.org/index.php/ijritcc/article/view/11942
43. Rangappa, A. S. (2020). Integration of behavioral analytics in developer workflows for detecting insider threats and malicious activities through activity monitoring and contextual profiling. International Journal of Innovative Research in Science, Engineering and Technology, 9(1), 13375–13383. https://doi.org/10.15680/IJIRSET.2020.0901064
44. Gupta, A. (2021). Application of cyber deception strategies in military networks for enhancing operational security and misleading attackers through adaptive honeypots and decoy systems. NeuroQuantology, 19(4), 317–333. https://doi.org/10.48047/nq.2021.19.4.NQ21068
45. Mahadasu, S. S. (2024). Real-time network digital twin synchronization for predictive capacity planning in multi-vendor private 5G systems. International Journal of Engineering Research and Science & Technology, 20(4), 483–488. https://doi.org/10.62643/ijerst.2024.v20.n4.4405
46. Jain, J., & Jain, A. (2022). Securing e-Healthcare Images Using an Efficient Image Encryption Model. Scientific Programming, 2022(1), 6438331.
47. Pandit, T., Mahendru, S., Raval, M., & Upadhyay, D. (2026). The evolution of reranking models in information retrieval: From heuristic methods to large language models. In X. Ding & Y. Dong (Eds.), Computational linguistics and natural language processing: CLNLP 2025 (Communications in Computer and Information Science, Vol. 2775, pp. 56–67). Springer. https://doi.org/10.1007/978-981-95-4788-3_6
48. Chatrath, A. (2021). Implementation of AI in recruitment platforms for improving candidate screening and diversity inclusion through bias-free resume parsing and skill matching. International Journal of Innovative Research in Computer and Communication Engineering, 9(12). https://doi.

org/10.15680/IJIRCCE.2021.0912046

49. Gupta, A. (2021). Implementation of cyber threat hunting in hybrid cloud environments for early breach detection and threat containment through behavior-based analysis tools. NeuroQuantology, 19(12), 1312–1329. https://doi.org/10.48047/nq.2021.19.12.NQ21342
50. Bhaskara Raju Rallabandi. (2023). AGENTIC-AI xAPPS: O-RAN MULTI-AGENT CONTROL FOR ENTERPRISE POLICY, SCHEDULING & SPECTRUM. International Journal of Data Science and IoT Management System, 2(4), 54–58. https://doi.org/10.64751/ijdim.2023.v2.n4.pp54-58
51. Talasila, D. (2023). Implementation of cloud-based disaster recovery models for minimizing business downtime and ensuring operational continuity through geographically distributed backup systems. International Journal on Recent and Innovation Trends in Computing and Communication, 11(8), 867–874. https://www.ijritcc.org/index.php/ijritcc/article/view/11934
52. Bhanushali, B. (2022). Blockchain-driven carbon credit trading systems for improving environmental transparency, reducing double counting, and ensuring accountability in climate finance. International Journal on Recent and Innovation Trends in Computing and Communication, 10(11), 292–300. https://www.ijritcc.org/index.php/ijritcc/article/view/11927
53. Jain, Arpit, Nageswara Rao Moparthi, A. Swathi, Yogesh Kumar Sharma, Nitin Mittal, Ahmed Alhussen, Zamil S. Alzamil, and MohdAnul Haq. “Deep Learning-Based Mask Identification System Using ResNet Transfer Learning Architecture.” Computer Systems Science & Engineering 48, no. 2 (2024): 341.