

DASN: A Decentralized Anonymous Framework for Community Intelligence and Criminal Logistics Disruption in Nigeria

Oluwaseun Adeniyi Ojerinde^{1*}, AbdulSalaam Kolawole Noibi², Ramatu Abubakar³, Joseph Adebayo Ojeniyi⁴

^{1,2,3,4} Federal University of Technology, Minna, Niger State, Nigeria

ARTICLE INFO

Article History

Received: August 2, 2026
Revised: September 16, 2026
Accepted: September 25, 2026

Keywords

Natural Language Processing;
Smart Contracts;
Community Intelligence;
Criminal Logistics Disruption;
Predictive Policing

ABSTRACT

Organized banditry and mass kidnapping in Nigeria depend on physical logistics, such as fuel, ammunition, food and communication credit, that reactive security responses largely overlook. Community members can observe these logistics anomalies, but reporting is constrained by fear of retaliation, identity exposure and the lack of automated means to separate credible reports from decoys. This paper proposes the Decentralized Anonymous Security Network (DASN), a proof-of-concept framework that combines a machine-learning and rule-based decision layer with an Ethereum-compatible blockchain ledger. DASN accepts reports through Unstructured Supplementary Service Data (USSD) and a Progressive Web Application and replaces the informant's telecommunications identifier with a salted hash at ingestion. A spaCy Named Entity Recognition component and a Random Forest classifier estimate threat probability, while a Solidity smart contract anchors report hashes on-chain and maintains informant reputation scores. On synthetic evaluation data, Random Forest achieved the highest performance among four classifiers, with 85% accuracy, 0.86 precision, 0.85 recall and 0.85 F1-score. Hybrid on-chain/off-chain storage reduced gas consumption by approximately 70%, and asynchronous processing returned user acknowledgements in approximately 50 milliseconds. The results show the technical feasibility of integrating low-bandwidth reporting, informant pseudonymity, reputation scoring and criminal-logistics analytics in a single architecture.

* Corresponding author: o.ojerinde@futminna.edu.ng

<https://doi.org/10.56427/jcbd.v5i3.1058>

© 2026 The Authors. Published by PT. Delitekno Media Mandiri. This is an open access article under a Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International License. (<https://creativecommons.org/licenses/by-nc-sa/4.0/>)

1. Introduction

Organized banditry and kidnapping-for-ransom have become a defining security challenge in Nigeria, where expanding organized-crime networks and socio-economic instability strain national security infrastructure [1], [2]. Civilians are typically the first to observe unusual activity in their communities and therefore remain the primary source of the situational intelligence that security agencies need to deploy scarce operational resources [3], [4]. In Nigeria, however, this intelligence pipeline is structurally compromised: rural hinterlands, where banditry and kidnapping are most prevalent, remain largely uncovered by technological surveillance [5]. Existing reporting channels, including police hotlines, SMS shortcodes and web-based tip lines, rely on centralized client-server architectures that grant administrators broad access to reporter metadata; combined with Nigeria's National Identity Number (NIN)-SIM linkage policy, this exposes informants to identification and retaliation and sustains a culture of silence [6]. These channels also offer no reliable mechanism to separate credible reports from decoys, so operators are overwhelmed by unverified information.

Conventional responses are also oriented toward direct confrontation with armed combatants, while the logistical supply chains that sustain these groups, namely the movement of fuel, ammunition, food and communication credit between rural markets and forest encampments, receive comparatively little analytical attention [1], [7], [8]. Field research shows that armed syndicates cannot sustain forest enclaves without continuous rural supply [1], and because such flows route through legitimate commercial channels [7], they generate observable anomalies, such as unusual bulk purchases of fuel or medical supplies, that communities are well placed to notice. Communities hold this micro-logistical intelligence but largely decline to report it, citing fear of retaliation [5]; this reluctance is linked to reporting gateways that retain raw reporter metadata at the server, so that anonymity depends on administrative policy rather than on cryptographic enforcement at the point of ingestion [3], [6].

On the technical side, permissioned-ledger systems such as CRAB [9], JusticeChain [10] and PoliceChain [11] improve the auditability of institutional security records but assume authenticated, identifiable participants rather than anonymous grassroots informants. Several studies address anonymity more directly: Mbimbi et al. [3] propose a private-blockchain, zero-knowledge-proof evidence framework, Chaurasia et al. [12] automate complaint validation for anonymous crime reporting

through smart contracts, and Kasztelnik [13] shows that a smart contract can autonomously update an on-chain trust score for anonymous nodes, although in a financial-transaction context. Blockchain reporting infrastructure also has performance limits, as Akkala et al. [14] report latency spikes and storage bloat under high-volume conditions.

On the analytical side, machine learning has been combined with blockchain-secured logging for fraud detection [15], applied through SVM and Random Forest to historical police data [16], and extended to CNN-LSTM crime-rate forecasting [17], while network analysis of organized crime highlights the disruptive value of targeting structurally central actors [18]. A systematic review [19] notes that most crime-prediction models are trained on retrospective police records and rarely model the logistics dimension of organized crime. Recent AI-blockchain hybrids for IIoT security-event verification [20] and Web 4.0 security [21] integrate automated assessment with trust mechanisms and decentralized logging, but neither addresses anonymous low-bandwidth community reporting or criminal-logistics intelligence.

On the accessibility side, a USSD-based data-collection portal for disconnected rural communities [22] demonstrated that offline submission over GSM signalling is feasible without smartphones or mobile data, consistent with evidence that low-bandwidth mobile channels drive digital inclusion in Sub-Saharan Africa [23]; that portal, however, transmitted data in plain text to a centralized database. Table 1 summarizes the limitations of the reviewed studies relative to the requirements of DASN.

Table 1. Comparison of Related Studies and the Proposed DASN Framework

Study	Focus	Limitation relative to DASN
[1]	Supply-chain dependency of kidnapping syndicates (qualitative field research)	No computational or real-time tracking framework
[5]	Community policing and HUMINT (mixed-methods survey)	No technological solution to the anonymity problem
[3]	Anonymous whistleblower evidence reporting (SHA-256, zero-knowledge proofs)	Susceptible to decoy/flooding attacks; no AI verification layer
[13]	Automated trust computation via smart contracts	Financial-transaction context; no geospatial or text analytics
[16]	Crime-rate prediction with SVM and Random Forest	Trained on retrospective, potentially biased police data
[18]	Structural mapping of organized-crime networks	Manual network construction; no real-time data pipeline
[22]	USSD-based rural data-collection portal	Plain-text transmission; vulnerable to interception
[14]	Stress capacity of blockchain reporting systems	No AI filtering layer to reduce load before ledger writes
[11]	PoliceChain: Hyperledger Fabric with attribute-based access control	Assumes authenticated actors; no anonymous ingestion or analytics
[20]	AI-assisted blockchain verification of IIoT security events	No community intelligence or criminal-logistics analysis
[21]	AI-blockchain hybrid for Web 4.0 security and trust	No low-bandwidth anonymous reporting or logistics intelligence
This study	Anonymous community intelligence and criminal-logistics analysis (USSD/Web, blockchain, Random Forest)	Synthetic data, sandbox USSD and testnet evaluation; field and Layer-2 validation required

Collectively, the reviewed literature provides four separate capabilities, namely tamper-evident storage, cryptographic anonymization with reputation scoring, machine-learning crime prediction and offline low-bandwidth collection, but no reviewed framework integrates (i) anonymous, privacy-preserving community reporting, (ii) automated credibility and threat assessment, (iii) low-bandwidth accessibility and (iv) criminal-logistics-focused intelligence analysis within a single architecture. This study therefore proposes the Decentralized Anonymous Security Network (DASN) as an integrated proof-of-concept framework. Its contributions are (a) an architecture combining anonymous reporting, AI-assisted threat assessment and blockchain-based evidence management; (b) a dual USSD/Web reporting mechanism that cryptographically protects informant identifiers under low-connectivity conditions; (c) the coupling of machine-learning threat assessment with smart-contract-based reputation management to validate reports and penalize decoys; and (d) a prototype evaluation of threat-classification performance, smart-contract gas consumption and user-facing latency.

2. Research Methodology

2.1 System Design and Actors

DASN was developed through an iterative, modular process combining Agile development with object-oriented analysis and design, so that its three subsystems, namely USSD telecommunications, a distributed ledger and predictive analytics, could be developed and tested independently before integration. The design responds to three weaknesses of centralized reporting channels: concentrated administrative privilege, mitigated by anchoring report metadata to a blockchain instead of a relational database; procedural rather than cryptographic anonymity, mitigated by hashing identifying metadata at ingestion; and manual triage that does not scale and is vulnerable to decoy flooding, mitigated by automated classification and a blockchain-based reputation layer. Four actors interact with the system: the grassroots informant (farmers, traders, transport operators and other

community members), who submits reports via USSD or the Progressive Web Application (PWA); the smart contract, which records pseudonymous identifiers, anchors report hashes and maintains reputation scores; the AI analytics engine, which parses report text, extracts entities, computes a threat score and maps events onto a criminal-logistics network model; and the tactical commander, an authenticated security-agency user who receives validated alerts, authorizes deployment and supplies feedback that updates informant reputation. Figure 1 summarizes the resulting processing pipeline.

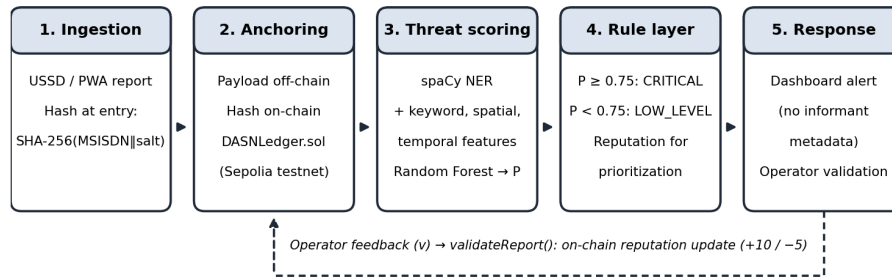


Figure 1. Conceptual Processing Pipeline of the DASN Prototype

2.2 Data-Flow Architecture

Figure 2 presents the Level-1 data-flow view of DASN. Process 1.0 ingests the raw submission and hashes the identifying metadata, so that no plain-text identifier is passed to later stages. Process 2.0 executes the smart contract and anchors the hashed payload to the blockchain ledger (D1). Process 3.0 performs AI threat scoring and anomaly detection without using informant reputation as an input and, after operator feedback, writes the reputation update to the reputation store (D2). Process 4.0 maps validated reports onto the criminal-logistics network model, consults the active tactical threat store (D3), and routes the resulting alert, stripped of informant metadata, to the tactical commander. The informant's current reputation accompanies the alert only for post-classification prioritization, and field outcomes are fed back to refine the classification and reputation models.

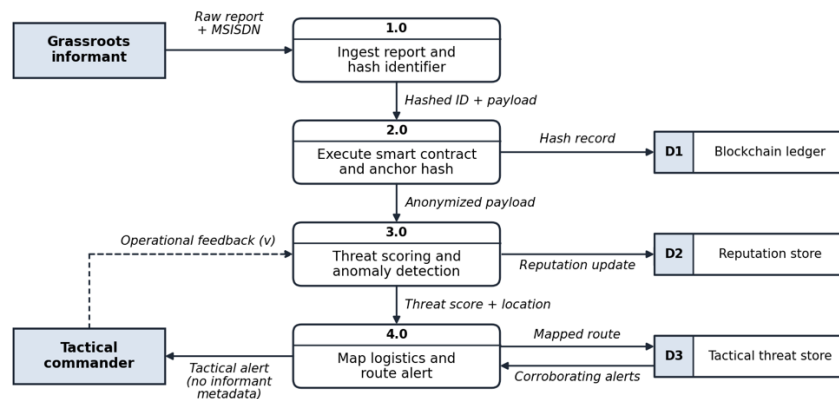


Figure 2. Level-1 Data-Flow Diagram of the DASN Framework

2.3 Hybrid Intelligence Layer and Trust-Scoring Model

The intelligence layer combines data-driven classification with explicit decision rules. Report text is processed with spaCy using the pretrained `en_core_web_sm` English model, and the entity-count feature is the number of named-entity spans recognized in a preprocessed report; no fine-tuning or adaptation to Nigerian place names, Nigerian Pidgin or indigenous-language entities was performed, so this component is a baseline implementation. A Random Forest classifier [24] then estimates a threat probability, after which a lightweight rule layer applies the escalation threshold and the reputation-update rule. The informant's reputation score is used only as a post-classification decision variable for validation and prioritization and is not an input to the classifier.

Let id denote an informant's persistent pseudonymous identifier, generated on first contact as a salted SHA-256 hash of the device's telecommunications identifier (MSISDN), so that the plain-text MSISDN is not stored or processed beyond the ingestion process (Equation 1).

$$id = SHA - 256(MSISDN \parallel salt) \quad (1)$$

This protection amounts to pseudonymity with a limited scope. Because the space of valid MSISDNs is small, any party holding the salt can recover an identifier by exhaustive hashing; the mobile network operator and the USSD aggregator observe the MSISDN before hashing; and because id is persistent and anchored on a public test network, all reports from the same informant are linkable. DASN therefore shields informant identity from downstream data consumers, such as dashboard users and ledger observers, but not from telecommunications-level or salt-holding adversaries.

Each report is represented by a feature vector F_v of n extracted features (Equation 2), namely keyword frequency, spatial proximity, temporal frequency and entity count ($n = 4$).

$$F_v = [f_1, f_2, \dots, f_n] \quad (2)$$

An ensemble of K decision trees h_k takes F_v as input and outputs the threat probability as the mean of the individual tree votes (Equation 3).

$$P(T | F_v) = \frac{1}{K} \sum_{k=1}^K h_k(F_v) \quad (3)$$

The probability is discretized into an operational threat level using a validation threshold of 0.75 (Equation 4).

$$\begin{aligned} TL &= \text{CRITICAL_THREAT}, & \text{if } P(T | F_v) \geq 0.75 \\ TL &= \text{LOW_LEVEL}, & \text{if } P(T | F_v) < 0.75 \end{aligned} \quad (4)$$

After a tactical operator confirms ($v = \text{TRUE}$) or refutes ($v = \text{FALSE}$) a report, the informant's reputation score is updated from its previous value hs to ts (Equation 5).

$$\begin{aligned} ts &= hs + 10, & \text{if } v = \text{TRUE} & \text{ (report verified)} \\ ts &= hs - 5, & \text{if } v = \text{FALSE} & \text{ (report identified as a decoy)} \end{aligned} \quad (5)$$

The asymmetric reward and penalty allow a historically reliable source to recover after an isolated false report, while sustained decoy submission progressively lowers a source's standing until it is filtered from the dashboard. Reputation scores and transaction receipts are managed by the DASNLedger.sol smart contract, so every change to an informant's standing is executed through `validateReport()` and recorded on-chain as an auditable transaction; the contract cannot, however, verify the operator's verdict, so the tactical operator remains a trusted party in the reputation mechanism. The 0.75 threshold and the +10/-5 update values are heuristic proof-of-concept parameters chosen to favour conservative escalation; no sensitivity analysis was performed, so they should not be regarded as operationally optimal. Algorithm X summarizes the complete report-processing and reputation-update procedure.

Algorithm X DASN Report Processing and Reputation Update

Input: report r , informant MSISDN m , salt s ; **Output:** threat level TL , updated reputation ts

- 1: $id \leftarrow \text{SHA-256}(m \parallel s)$
 - 2: store r off-chain; `anchorIntelligence(id, hash(r))` on-chain
 - 3: $F_v \leftarrow$ [keyword frequency, spatial proximity, temporal frequency, entity count] of r
 - 4: $p \leftarrow \text{RandomForest}(F_v)$
 - 5: if $p \geq 0.75$ then $TL \leftarrow \text{CRITICAL_THREAT}$; route alert and reputation of id to dashboard
 - 6: else $TL \leftarrow \text{LOW_LEVEL}$
 - 7: on operator feedback v : `validateReport(id, v)`
 - 8: if $v = \text{TRUE}$ then $ts \leftarrow hs + 10$ else $ts \leftarrow hs - 5$
 - 9: return TL, ts
-

2.4 Prototype Implementation and Experimental Setup

The prototype was implemented as a multi-tier system comprising a Python/FastAPI backend, a React.js tactical dashboard, a Neo4j AuraDB graph database for spatial-temporal network analysis, and a deployment of DASNLedger.sol on the Ethereum Sepolia test network accessed through an Alchemy RPC endpoint. The USSD channel was implemented and tested using the Africa's Talking sandbox API rather than a live telecommunications network.

Because access to real operational intelligence and telecommunications metadata is restricted under Nigerian law, the classifier was trained and evaluated on a programmatically generated synthetic dataset structured around indicators drawn from the Armed Conflict Location & Event Data Project (ACLED). Classifier inputs comprised keyword frequency (weapon- and threat-related terms), spatial proximity, temporal frequency of incidents and NLP-derived entity count, and records were labelled according to the prototype's threat-classification criteria. A synthetic informant trust score was generated separately for evaluating the reputation mechanism but was excluded from classifier training to prevent circularity and target leakage. Pearson correlation analysis of the classifier inputs showed the strongest association with the threat label for entity count ($r = 0.37$), followed by keyword frequency ($r = 0.24$), with weaker associations for spatial proximity ($r = -0.10$) and temporal frequency ($r = -0.06$); pairwise correlations among the four inputs were negligible ($|r| \leq 0.05$), so no dimensionality reduction was applied. Numerical features were standardized, and the data were split into 80% training and 20% held-out test sets using stratified random sampling. The original implementation did not preserve the random seed or class-generation counts, which limits strict reproducibility; future evaluations will release the generation script, parameter settings, class distribution and fixed random seed.

Random Forest was compared with three baselines under identical features, preprocessing and data partitioning: Support Vector Machine (SVM), XGBoost and Logistic Regression. Fixed hyperparameters were used without systematic tuning: Random Forest with 100 trees, Gini criterion, unrestricted depth and `min_samples_split = 2`; SVM with an RBF kernel, `C = 1.0` and `gamma = 'scale'`; XGBoost with 100 estimators, learning rate 0.1, maximum depth 6 and subsample 1.0; and Logistic Regression with L2 regularization, `C = 1.0`, the `lbfgs` solver and 100 iterations. Blockchain performance was assessed through gas consumption and latency measurements. The USSD component was evaluated as a functional sandbox prototype covering submission, request handling, acknowledgement generation and asynchronous forwarding to the blockchain layer, not as a controlled telecommunications performance study.

3. Results and Discussion

3.1 Classification Performance

Table 2 reports the performance of the four classifiers on the held-out test set.

Table 2. Machine-Learning Model Performance Comparison

Model	Precision	Recall	F1-Score	Accuracy
Random Forest Classifier	0.86	0.85	0.85	85%
Support Vector Machine (RBF kernel)	0.82	0.81	0.82	81%
XGBoost Classifier	0.81	0.81	0.81	81%
Logistic Regression	0.78	0.78	0.78	78%

Random Forest achieved the strongest performance, with 85% accuracy, 0.86 precision, 0.85 recall and 0.85 F1-score, followed by SVM and XGBoost (81% accuracy) and Logistic Regression (78%). This is consistent with the non-linear structure of the synthetic data, in which threat severity depends on interactions among keyword, spatial, temporal and entity features, and Random Forest was therefore selected for the threat-scoring pipeline. Because the evaluation used synthetic data and a single train–test split without cross-validation, confidence intervals or significance testing, these results demonstrate proof-of-concept performance rather than statistically established superiority.

3.2 Smart-Contract Gas Efficiency

An initial implementation that stored full report payloads on-chain consumed approximately 150,058 gas per `anchorIntelligence()` call, a cost that scales with payload size. The contract was therefore refactored into a hybrid model in which the full payload is stored off-chain and only its cryptographic hash is anchored on-chain. Gas reduction was computed using Equation 6, and the results are summarized in Table 3.

$$\text{Gas Reduction}(\%) = \frac{G_{\text{naive}} - G_{\text{optimized}}}{G_{\text{naive}}} \times 100 \quad (6)$$

Table 3. Smart-Contract Gas Optimization (Naive versus Hybrid Off-Chain Storage)

Function	Access Type	Naive Gas (units)	Optimized Gas (units)	Reduction
<code>anchorIntelligence()</code>	Write (state change)	150,058	≈44,962	≈70%
<code>validateReport()</code>	Write (state change)	≈84,650	≈34,926	≈59%
<code>getReputationScore()</code>	Read-only (view)	0	0	—

The hybrid design reduced the cost of `anchorIntelligence()` by approximately 70% and that of `validateReport()`, which executes the reputation update in Equation 5, by approximately 59%, while read-only calls such as `getReputationScore()` incur no gas cost. Storing compact cryptographic commitments rather than complete intelligence payloads therefore substantially reduces on-chain transaction costs.

3.3 Latency and Projected Scalability

On Sepolia, write transactions were confirmed in approximately 12.4 s (`anchorIntelligence()`) and 11.8 s (`validateReport()`), and two block confirmations required approximately 24 s (Table 4); this confirmation depth is a prototype criterion rather than Ethereum's protocol-level finality, which under proof-of-stake requires about two epochs (approximately 12.8 min). This is acceptable for evidentiary archiving but would exceed the session-timeout window of typical USSD gateways (approximately 2.5 s), which was treated as an architectural constraint rather than an experimentally measured sandbox limit. An asynchronous fire-and-forget queue therefore decouples the interface from block confirmation, returning an application-level acknowledgement in approximately 50 ms while the transaction is confirmed in the background; this value reflects perceived responsiveness, not blockchain finality.

Table 4. Blockchain and User-Facing Latency

Operation	Measured latency
anchorIntelligence() confirmation	≈12.4 s
validateReport() confirmation	≈11.8 s
Two-block confirmation	≈24 s
User-facing acknowledgement	≈50 ms

Table 5 compares Layer-1 (Sepolia) reference values with target values for Arbitrum One, an Ethereum-compatible Layer-2 network. The Layer-2 values are projections based on expected performance characteristics and were not measured in this study. The Layer-2 confirmation time refers to sequencer (soft) confirmation; settlement on Ethereum still depends on batch posting to Layer 1 and, for optimistic rollups, on the fraud-proof challenge period. Migration to Layer-2 is projected to reduce block time about 48-fold and raise throughput more than 100-fold, consistent with reported Layer-2 scaling gains [25], but empirical benchmarking under realistic transaction loads is required to validate these projections.

Table 5. Throughput and Scalability: Layer-1 Testnet versus Projected Layer-2 Deployment

Metric	Ethereum L1 (Sepolia Testnet)	L2 Target (Arbitrum One)	Projected Improvement
Block generation time	≈12.0 s	≈0.25 s	≈48× faster
Maximum throughput	≈17 reports/s	≈2,000+ reports/s	>100× capacity
Time to confirmation	≈24.0 s (2 blocks)	≈1.5 s (soft)	≈16× faster

3.4 Discussion

The results indicate that combining machine-learning classification and rule-based decisions with a blockchain trust and anonymization layer is technically feasible for grassroots crime reporting in Nigeria. The balanced precision (0.86) and recall (0.85) of Random Forest suggest that credible logistics-related threats can be identified while limiting false alarms, although this has only been shown on synthetic data. Hashing identifiers at ingestion reduces identity exposure to downstream data consumers and complements institution-oriented systems such as PoliceChain [11], while hybrid storage and asynchronous queuing reduce on-chain cost and perceived latency, subject to the smart-contract security risks documented in [26]. The evaluation has five main limitations: the classifier was trained on synthetic rather than operational data, so generalizability is not established; the NLP component lacks support for Nigerian Pidgin and local languages [27]; the USSD channel was tested only in a sandbox and Layer-2 performance was not benchmarked; informant protection amounts to pseudonymity against downstream parties rather than anonymity against telecommunications-level adversaries (Section 2.3); and robustness against large-scale adversarial or decoy attacks was not evaluated [28]. These limitations motivate validation with appropriately anonymized real-world data and larger-scale deployment experiments.

4. Conclusion

This paper presented DASN, a blockchain- and AI-based framework for anonymous community intelligence reporting and criminal-logistics disruption designed for the connectivity and trust constraints of rural Nigeria. DASN combines a dual USSD/Web reporting channel, a blockchain layer that records pseudonymized informant identifiers and maintains on-chain reputation scores, and a hybrid machine-learning and rule-based layer that converts unstructured civilian reports into structured threat intelligence. In prototype evaluation, Random Forest achieved the highest performance among four classifiers on synthetic data (85% accuracy, 0.86 precision, 0.85 recall and 0.85 F1-score), hybrid on-chain/off-chain storage reduced gas consumption by approximately 70%, and asynchronous queuing reduced perceived latency to approximately 50 ms. These findings demonstrate technical feasibility rather than operational effectiveness at national scale. Future work will validate the framework with anonymized real-world intelligence, evaluate the USSD channel in large-scale field trials, adapt the NLP component to Nigerian linguistic and geographic contexts, and benchmark the architecture on Layer-2 infrastructure.

References

- [1] J. S. Ojo, S. Oyewole, and F. Aina, “Forces of terror: Armed banditry and insecurity in North-west Nigeria,” *Democracy and Security*, vol. 19, no. 4, pp. 319–346, 2023, doi: [10.1080/17419166.2023.2164924](https://doi.org/10.1080/17419166.2023.2164924).
- [2] O. Ojewale, “Shadow of death: The criminal economy of banditry and kidnapping in northwest Nigeria,” *Security Journal*, vol. 37, no. 4, pp. 1680–1710, 2024, doi: [10.1057/s41284-024-00436-2](https://doi.org/10.1057/s41284-024-00436-2).
- [3] B. Mbimbi, D. Murray, and M. Wilson, “Preserving whistleblower anonymity through zero-knowledge proofs and private blockchain: A secure digital evidence management framework,” *Blockchains*, vol. 3, no. 2, Art. no. 7, 2025, doi: [10.3390/blockchains3020007](https://doi.org/10.3390/blockchains3020007).
- [4] U. Ismail, “Application of big data for intelligence-led policing among law enforcement agencies in Nigeria,” *International Journal of Police Science*, vol. 4, no. 2, 2025, doi: [10.56331/ijps.v4i2.9115](https://doi.org/10.56331/ijps.v4i2.9115).

- [5] C. L. Ezeji, "Innovative and disruptive policing: Considering intelligence-driven community policing as a remodeled strategy for addressing crime in contemporary society," *International Journal of Business Ecosystem & Strategy*, vol. 6, no. 5, pp. 254–268, 2024, doi: [10.36096/ijbes.v6i5.688](https://doi.org/10.36096/ijbes.v6i5.688).
- [6] S. Banaeian Far and M. Rajabzadeh Asaar, "A blockchain-based anonymous reporting system with no central authority: Architecture and protocol," *Cyber Security and Applications*, vol. 2, Art. no. 100032, 2024, doi: [10.1016/j.csa.2023.100032](https://doi.org/10.1016/j.csa.2023.100032).
- [7] N. Magliocca et al., "Comparative analysis of illicit supply network structure and operations: Cocaine, wildlife, and sand," *Journal of Illicit Economies and Development*, vol. 3, no. 1, pp. 50–73, 2021, doi: [10.31389/jied.76](https://doi.org/10.31389/jied.76).
- [8] O. Ojewale, "The bandits' world: Recruitment strategies, command structure and motivations for mass casualty attacks in northwest Nigeria," *Small Wars & Insurgencies*, vol. 35, no. 2, pp. 228–255, 2024, doi: [10.1080/09592318.2024.2301713](https://doi.org/10.1080/09592318.2024.2301713).
- [9] M. A. Tasnim, A. Al Omar, M. S. Rahman, and M. Z. A. Bhuiyan, "CRAB: Blockchain based criminal record management system," in *Security, Privacy, and Anonymity in Computation, Communication, and Storage (SpaCCS 2018)*, *Lecture Notes in Computer Science*, vol. 11342, Cham: Springer, 2018, pp. 294–303, doi: [10.1007/978-3-030-05345-1_25](https://doi.org/10.1007/978-3-030-05345-1_25).
- [10] R. Belchior, M. Correia, and A. Vasconcelos, "JusticeChain: Using blockchain to protect justice logs," in *On the Move to Meaningful Internet Systems: OTM 2019 Conferences*, *Lecture Notes in Computer Science*, vol. 11877, Cham: Springer, 2019, pp. 318–325, doi: [10.1007/978-3-030-33246-4_21](https://doi.org/10.1007/978-3-030-33246-4_21).
- [11] A. Mukherjee and R. Halder, "PoliceChain: Blockchain-based smart policing system for smart cities," in *Proc. 13th Int. Conf. Security of Information and Networks (SIN 2020)*, 2020, pp. 1–5, doi: [10.1145/3433174.3433618](https://doi.org/10.1145/3433174.3433618).
- [12] P. K. Chaurasia, D. Rana, R. V. V, and S. Srividhya, "Anonymous crime reporting using blockchain and smart contract," in *Proc. 4th Int. Conf. Pervasive Computing and Social Networking (ICPCSN 2024)*, 2024, pp. 1059–1064, doi: [10.1109/ICPCSN62568.2024.00176](https://doi.org/10.1109/ICPCSN62568.2024.00176).
- [13] K. Kasztelnik, "Blockchain technology and smart contracts for fraud detection and deterrence in cryptocurrency markets," *Journal of Forensic Accounting Research*, vol. 10, no. 1, pp. 102–128, 2025, doi: [10.2308/JFAR-2024-023](https://doi.org/10.2308/JFAR-2024-023).
- [14] S. Akkala, S. Gupta, and P. V. Sai, "Secure and tamper-proof blockchain-enabled cybercrime reporting and evidence management system," *ITM Web of Conferences*, vol. 81, Art. no. 01007, 2026, doi: [10.1051/itmconf/20268101007](https://doi.org/10.1051/itmconf/20268101007).
- [15] T. Ashfaq et al., "A machine learning and blockchain based efficient fraud detection mechanism," *Sensors*, vol. 22, no. 19, Art. no. 7162, 2022, doi: [10.3390/s22197162](https://doi.org/10.3390/s22197162).
- [16] A. A. Ibrahim, Y. M. Malgwi, and Y. Ali, "A hybrid machine learning model for crime rate prediction," *FUDMA Journal of Sciences*, vol. 8, no. 6, pp. 101–106, 2024, doi: [10.33003/fjs-2024-0806-2789](https://doi.org/10.33003/fjs-2024-0806-2789).
- [17] M. Muthamizharasan and R. Ponnusamy, "Forecasting crime event rate with a CNN-LSTM model," in *Innovative Data Communication Technologies and Application*, *Lecture Notes on Data Engineering and Communications Technologies*, vol. 96, Singapore: Springer, 2022, pp. 461–470, doi: [10.1007/978-981-16-7167-8_33](https://doi.org/10.1007/978-981-16-7167-8_33).
- [18] E. R. Kleemans, "Organized crime, transit crime, and racketeering," *Crime and Justice*, vol. 35, no. 1, pp. 163–215, 2007, doi: [10.1086/501509](https://doi.org/10.1086/501509).
- [19] N. Iqbal, A. Hassan, and T. Waheed, "AI-driven crime prediction: A systematic literature review," *Journal of Computational Social Science*, vol. 8, no. 2, Art. no. 53, 2025, doi: [10.1007/s42001-025-00373-z](https://doi.org/10.1007/s42001-025-00373-z).
- [20] I. Ortiz-Gárce, P. Palacios, J. Guna-Moya, and W. Villegas-Ch., "Distributed verification of security events in IIoT networks using low-latency blockchain consensus assisted by artificial intelligence," *Frontiers in Computer Science*, vol. 8, Art. no. 1800175, 2026, doi: [10.3389/fcomp.2026.1800175](https://doi.org/10.3389/fcomp.2026.1800175).
- [21] S. R. Sabbah et al., "An AI-blockchain hybrid model to enhance security and trust in Web 4.0," *Computers, Materials & Continua*, vol. 88, no. 1, Art. no. 41, 2026, doi: [10.32604/cmc.2026.079241](https://doi.org/10.32604/cmc.2026.079241).
- [22] S. Anunobi, M. Memon, C. Howard, and B. Lawal, "Mitigating Security Vulnerabilities in Offline USSD Payments in non-Smartphones," in *2024 11th International Conference on Software Defined Systems (SDS)*, IEEE, Dec. 2024, pp. 42–49, doi: [10.1109/sds64317.2024.10883928](https://doi.org/10.1109/sds64317.2024.10883928).
- [23] L. Grzybowski, V. Lindlacher, and O. Mothobi, "Mobile money and financial inclusion in Sub-Saharan Africa," *Information Economics and Policy*, vol. 65, Art. no. 101064, 2023, doi: [10.1016/j.infoecopol.2023.101064](https://doi.org/10.1016/j.infoecopol.2023.101064).
- [24] L. Breiman, "Random forests," *Machine Learning*, vol. 45, no. 1, pp. 5–32, 2001, doi: [10.1023/A:1010933404324](https://doi.org/10.1023/A:1010933404324).
- [25] M. H. Nasir, J. Arshad, M. M. Khan, M. Fatima, K. Salah, and R. Jayaraman, "Scalable blockchains — A systematic review," *Future Generation Computer Systems*, vol. 126, pp. 136–162, 2022, doi: [10.1016/j.future.2021.07.035](https://doi.org/10.1016/j.future.2021.07.035).
- [26] S. S. Kushwaha, S. Joshi, D. Singh, M. Kaur, and H.-N. Lee, "Systematic review of security vulnerabilities in Ethereum blockchain smart contract," *IEEE Access*, vol. 10, pp. 6605–6621, 2022, doi: [10.1109/ACCESS.2021.3140091](https://doi.org/10.1109/ACCESS.2021.3140091).
- [27] D. I. Adelani et al., "MasakhaNER: Named entity recognition for African languages," *Transactions of the Association for Computational Linguistics*, vol. 9, pp. 1116–1131, 2021, doi: [10.1162/tac1_a_00416](https://doi.org/10.1162/tac1_a_00416).
- [28] X. Ma, D. Yu, Y. Du, L. Li, W. Ni, and H. Lv, "A blockchain-based incentive mechanism for sharing cyber threat intelligence," *Electronics*, vol. 12, no. 11, Art. no. 2454, 2023, doi: [10.3390/electronics12112454](https://doi.org/10.3390/electronics12112454).