Journal of Animal Environment

(E-ISSN 2717-1396, ISSN 2717-1388)

Original Research Paper

Counterfactual Transaction-Motif Memory with Market-Crime Drift Attribution for Open-World Cryptocurrency Forensics

M. Kaliappan[1], S Vimal[2], Gaurav Dhiman[3,4,5], S. Santhi[6]

[1]Department of Artificial Intelligence and Data Science, Ramco Institute of Technology, Rajapalayam, Tamil Nadu 626117, India

[2]Department of Computer Science and Engineering, KIT- Kalaignar Karunanidhi Institute of Technology, Coimbatore, Tamil Nadu 641402 India

[3]Department of Computer Science and Engineering, Yuan Ze University, Tao Yuan, Taiwan

[4]University Centre for Research and Development, Chandigarh University, Mohali, India

[5]Division of Research and Development, Lovely Professional University, Phagwara, India

[6]Department of Computer Science and Engineering, KIT- Kalaignar Karunanidhi Institute of Technology, Coimbatore, Tamil Nadu 641402 India

Email: kaliappan@ritrjpm.ac.in, svimal@kitcbe.ac.in, gdhiman0001@gmail.com, ssanthi.kit@gmail.com

1 Abstract

Cryptocurrency ransomware attribution is intrinsically open world: a forensic learner must identify previously observed campaigns, reject later-emerging families, distinguish criminal-motif change from benign market drift, and prevent uncertain online samples from corrupting its memory. This paper presents CausalLedger-OWL, a counterfactual multi-view framework that separates transaction-shape and value-flow evidence from temporal market context, stores multimodal known-family prototypes, fuses prototype distance, energy, tri-view disagreement, and normalized predictive entropy, attributes drift through separate market and crime statistics, and permits only uncertainty-gated, agreement-gated, source-anchored prototype updates. The revised implementation defines a strict global protocol on the UCI BitcoinHeist corpus: training through 2014, source-only calibration in 2015, and open-world evaluation during 2016-2018 with five dynamically selected known ransomware families and four completely held-out families. The supplied revision completed end-to-end implementation validation and produced all required outputs. Empirical security evidence is reported from a version-controlled predecessor BitcoinHeist pilot containing 26,740 selected addresses. That pilot attained an H-score of 0.2184 and AUROC of 0.5666, below the calibrated MSP baseline (0.2763 and 0.5995), so superiority in unknown-family ranking is not claimed. Its trust gate nevertheless reduced final source-memory forgetting from 0.1561 to 0.0106 and retained known-family accuracy of 0.2905 under 50% unknown contamination, compared with 0.2088 for naive updating. The results localize the defensible contribution to contamination-resistant memory and identify novelty calibration as the principal confirmatory research question.

Keywords: Bitcoin ransomware; cryptocurrency forensics; open-world learning; zero-day detection; causal representation; counterfactual intervention; concept drift; prototype memory; conformal calibration; continual cybersecurity.

***Corresponding Author:**

Received: 25th July, 2026 | Revised: 26th August, 2026 | Accepted: 18th September, 2026 | Available Online: 24th September, 2026

(DOI): 10.70102/AEJ.2026.18.7s.57

2 Introduction

Bitcoin's public ledger exposes transaction structure but not verified identity. Ransomware actors exploit this separation by generating new addresses, mixing funds, and changing merging behavior across campaigns. The BitcoinHeist study demonstrated that graph-derived length, weight, count, loop, neighbor, and income descriptors encode useful obfuscation patterns [1]. Recent ransomware classifiers confirm that these variables remain useful for supervised discrimination [2, 3, 4, 5, 6]. Yet the operational task is not ordinary random-split classification. A deployed forensic model encounters temporal distribution shift, new criminal families without training labels, and uncertain samples that can poison adaptive memory.

Three errors are especially costly. First, a closed-set model assigns every novel campaign to an existing class. Second, a global novelty threshold may mistake benign market-volume changes for criminal innovation. Third, unrestricted online centroid updates can absorb zero-day transactions into known-family memory. The resulting system becomes increasingly confident in a corrupted representation. These risks motivate an event-triggered open-world formulation in which recognition, rejection, drift attribution, and memory updates are jointly controlled.

Graph-based blockchain detectors increasingly model heterogeneous relations, adaptive aggregation, and temporal links [7, 8, 9, 10, 11, 12, 13, 14]. Separately, open-set recognition studies unknown support, extreme-value boundaries, data generation, and real-world label openness [33, 34, 35, 36, 37, 38, 39]. Concept-drift research adds incremental or continual adaptation [15, 16, 17, 18, 19, 20, 21, 22]. However, these streams rarely combine: (i) an explicit causal/context split, (ii) counterfactual market intervention, (iii) multi-prototype family memory, (iv) source-only novelty calibration, (v) market-versus-crime drift attribution, and (vi) a gate that prevents rejected or uncertain transactions from changing known memory.

- A zero-day cryptocurrency-forensics formulation that holds later-emerging ransomware families out of representation learning and all threshold selection.
- A dual-view transaction encoder whose causal motif is trained to remain class-consistent after market-context swapping.
- A multi-prototype family memory with source radii, tri-view agreement, calibrated rejection, and zero-day quarantine.
- A dual MMD detector that distinguishes market-context movement from criminal-motif movement before adaptation.
- A bounded, source-anchored trust gate whose strongest observed benefit is resistance to contaminated online updates.
- A version-controlled single-cell implementation with strict global year cutoffs, leakage controls, end-to-end code-path validation, fifteen publication-resolution outputs, and serialized audit artifacts.

3 Literature Review

Recent Bitcoin-ransomware studies have moved from rule systems toward learned family classifiers. Dib et al. formulated machine-learning classification of Bitcoin transactions, Alsaif examined a related supervised formulation, and Alraizza and Algarni synthesized ransomware detection methods [2, 3, 4]. Garin and Gisin studied address classification, whereas rule-based ransomware detection remains an interpretable but rigid comparator [5, 6]. These studies justify transaction-graph descriptors but predominantly assume a fixed label inventory. The key missing operational question is whether a score calibrated without future-family labels can reject families that emerge later.

The broader illicit-finance literature increasingly exploits graph structure. Guo et al. investigated covert transactions with graph-based defenses; Kim et al. used heterogeneous transaction graphs for phishing accounts; and Li and He proposed graph-neural transaction tracking [7, 8, 9]. Adaptive sampling and aggregation improve fraud-oriented GNNs [10], while future-link prediction and heterogeneous-network link models emphasize dynamics [11, 12]. Feature-gated temporal learning and graph continual learning further indicate that temporal behavior cannot be treated as independent and identically distributed [13, 14]. CausalLedger-OWL complements these graph-heavy methods: it consumes the published BitcoinHeist graph summaries so the contribution can be isolated to causal separation, open-set calibration, and safe memory rather than end-to-end graph construction.

Open-set recognition is related to, but not identical with, out-of-distribution detection. Wang et al. show that task definitions and benchmarks can change apparent method rankings [33]. Noisy open-set environments further complicate calibration [34]. Extreme-value modeling [35], orientational distribution learning [36], unknown-support prototypes [37], generated open data [38], and real-world open-set analysis [39] demonstrate that unknown evidence may arise from geometry, logits, generated surrogates, or support modeling. This motivates the present score fusion, but also requires an ablation against individual evidence channels. The executed result makes that ablation decisive: source-

selected fusion is inferior to MSP, so fusion is treated as a hypothesis rather than an established improvement.

Drift-aware learning must decide both when and how to adapt. Casado et al. connect drift detection with federated and continual learning [15]. Intrusion-detection studies combine drift detection with online incremental learning [16] and examine continual models under evolving threats [17, 18]. Localization of drift in IoT streams [19] is especially relevant because global change can arise from benign context rather than attack semantics. Recent self-supervised adaptation, hybrid online detection, robust continual representations, and chronological leave-one-attack-family evaluation extend this direction [20, 21, 22, 45]. CausalLedger-OWL adds attribution: market MMD and crime-motif MMD are measured separately, and only crime-active, low-uncertainty, tri-view-consistent samples may update a known prototype.

Causal domain generalization seeks features stable under environmental intervention. Causal disentanglement has been used to reduce semantic bias [23], and generative intervention supports single-domain generalization [24]. Graph causal representations target out-of-distribution generalization [25], while cross-domain causal invariance and invariant meta-learning emphasize environment-stable mechanisms [26, 27]. Progressive causal-feature learning and deep discriminative causal generalization provide related mechanisms [28, 29]. Fine-grained decomposition, semi-supervised incremental invariance, and information bottlenecks further connect stability with compression and incremental domain exposure [30, 31, 32]. The present model operationalizes this principle by swapping learned market states across temporal environments and requiring the re-encoded criminal motif and class decision to remain invariant [42]-[63].

A rejection system must distinguish ranking quality from threshold validity. Recent work combines uncertainty with conformal prediction in dynamic visual environments [40], proposes anomaly-detection mechanisms based on conformal ideas [41], and analyzes uncertainty-aware time-series anomalies [42]. Renganathan et al. show that anomaly detection under model uncertainty requires explicit assumptions [43]. Accordingly, this paper uses class-conditional source calibration but does not claim unconditional conformal coverage after arbitrary temporal shift. The valid statement is narrower: empirical robustness of a source-calibrated threshold is measured under a chronological class-holdout protocol.

The literature supports each component separately but does not establish that their combination will improve every metric. The proposed integration is scientifically useful because it yields component-level falsifiability: counterfactual stability can be measured, prototype counts can be ablated, market and crime drift can be separated, and contamination damage can be quantified. This design prevents a favorable aggregate accuracy from concealing a defective novelty or adaptation mechanism.

4 Proposed Work

4.1 Threat model and notation

Let x_t denote an address-level observation at time t, y_t a known label in C_K or an unobserved family in C_U, and e_t a temporal market environment. The learner observes labeled source samples only from C_K. It may access unlabeled target features online but never uses held-out family identity for threshold tuning or update decisions. Adversaries may introduce previously unseen transaction motifs and may contaminate an adaptation stream. The objective is simultaneous known-class prediction, zero-day rejection, drift attribution, and source-memory retention.

$x_t = [x_t^{shape}, x_t^{flow}, x_t^{market}]$, $C_K \cap C_U = \emptyset$

4.2 Architecture

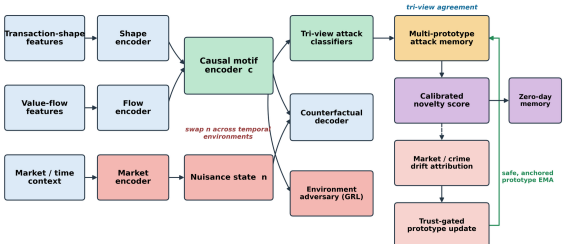


Figure 1. CausalLedger-OWL architecture. Transaction-shape and value-flow views form the causal motif; time and daily market aggregates form the nuisance state. Counterfactual swapping, calibrated novelty, dual drift attribution, and trust-gated memory updates are explicitly separated.

The shape view encodes mixing length, signed merge weight, count, loops, neighbors, loop-to-count ratio, and count-to-neighbor ratio. The flow view encodes log income, income relative to the daily median, weight per path length, and income per neighbor. The market view contains year, sinusoidal day-of-year coordinates, daily median income, daily transaction count, and transaction-to-market income contrast. Robust clipping and standardization are fit only on training positions.

Table 1. Representation views, dimensions, and operational roles.

View	Dimensions	Role
Transaction shape	7	Graph-motif structure intended to retain crime semantics
Value flow	4	Magnitude and relative-flow evidence
Market context	6	Calendar and daily-volume nuisance variables
Causal embedding	64	Known classification, prototypes, novelty, crime drift
Market embedding	24	Environment classification and market drift

4.3 Causal–nuisance decomposition and counterfactual intervention

$s = E_s(x^{shape})$, $f = E_f(x^{flow})$, $c = E_c([s; f])$, $n = E_n(x^{market})$

Equation (2) assigns transaction structure to c and explicitly routes calendar and market aggregates to n. This assignment alone does not guarantee disentanglement. Therefore, a decoder R reconstructs the input representation, and nuisance vectors are permuted across environments while c is held fixed.

$h = R([c; n])$, $h_{cf} = R([c; n_\pi])$, $\pi(i)$ chosen from another environment

$$L_{CF} = CE(g_c(E_c(h_{cf})), y) + \|E_c(h_{cf})-c\|_2^2 + \|E_n(h_{cf})-n_\pi\|_2^2 \quad (4)$$

The first term in (4) enforces label invariance; the two cycle terms require the decoder to preserve the retained causal factor and inserted nuisance factor. A gradient-reversal environment head simultaneously discourages e from being predictable from c, while a standard environment head encourages n to retain environmental information.

4.4 Multi-prototype attack memory

$P_k = \{p_{k1}, \ldots, p_{kM}\}$, $p_{km} = KMeans_m(\{c_i : y_i=k\})$

$$D_p(x) = \min_m \|c(x)-p_{km}\|_2 / (r_{km}+\varepsilon), \quad k = \arg\max_j p(y=j|x) \quad (5)$$

Multiple prototypes represent multimodal campaign behavior. The radius r_{km} is the 95th percentile of source distances assigned to prototype m. Radius normalization makes distance comparable across dense and diffuse modes. Compactness and separation losses pull samples toward a same-class mode while penalizing class-center distances below a margin.

4.5 Multi-evidence novelty and source-only calibration

$$E(x) = -\log \Sigma_k \exp(z_k(x)) \quad (6)$$

$$J(x) = \tfrac{1}{2} KL(p \,\|\, m) + \tfrac{1}{2} KL(a \,\|\, m), \quad m=(p+a)/2 \quad (7)$$

$$U(x) = -(1/\log K) \Sigma_k p_k(x) \log(p_k(x)+\varepsilon) \quad (8)$$

$$A(x) = w_p \tilde{D}_p(x) + w_e \tilde{E}(x) + w_j \tilde{J}(x) + w_u \tilde{U}(x), \quad \Sigma w=1 \quad (9)$$

Here p is the main causal-head distribution and a is the average of the shape and flow auxiliary distributions. Prototype distance, energy, Jensen-Shannon view disagreement, and normalized entropy are standardized on source calibration data. The revised implementation evaluates six non-negative convex weight vectors against pseudo-open source examples generated by suppressing the true-class logit. Single-component corners are intentionally excluded so the calibration search cannot collapse to an unregularized one-score detector. No held-out family label is used for scaling, weight selection, threshold fitting, drift calibration, or adaptation.

$$q_k^{\alpha} = Quantile_{1-\alpha}\{A_i : \hat{y}_i=k, y_i=k\}, \quad \alpha=0.05$$

$$\hat{y}_{open}(x) = \hat{y}(x) \text{ if } A(x) \le q^{\alpha}_{\hat{y}}(x); \text{ otherwise UNKNOWN}$$

4.6 Dual drift attribution

$$MMD^2(X,Y)=n^{-2}\Sigma_{i,j} \kappa(x_i,x_j)+m^{-2}\Sigma_{i,j} \kappa(y_i,y_j)-2(nm)^{-1}\Sigma_{i,j} \kappa(x_i,y_j) \quad (2)$$

$$\Delta_{market}=.75\ MMD^2(N_s,N_t)+.25\ \|\mu(N_t)-\mu(N_s)\|_2/\sqrt{d_n} \quad (3)$$

$$\Delta_{crime}=.65\ MMD^2(C_s,C_t)+.35\ mean_x \min_{k,m} \|c_x-p_{km}\|_2/(r_{km}+\varepsilon) \quad (4)$$

An RBF kernel with a median-distance bandwidth is used. The 95th percentile of calibration-window scores defines each drift threshold. Market-only alarms freeze criminal memory. A crime-drift alarm merely permits evaluation of the trust gate; it does not itself authorize an update.

4.7 Trust-gated update and theoretical properties

$$g_t = 1[A_t \le q_{\hat{y}}^{\alpha}] \cdot 1[U_t \le \tau_u] \cdot 1[\text{tri-view agreement}] \cdot 1[\Delta_{crime}>\tau_c] \cdot 1[\neg \text{market-only drift}] \quad (5)$$

$$\eta_t = g_t \eta_0 (1-U_t) \exp(-D_p(x_t)) \quad (6)$$

$$p_{km}^{t+1}=(1-\eta_t)p_{km}^{t}+\eta_t c_t; \quad \bar{p}_{km}^{t+1}=(1-\gamma)p_{km}^{t+1}+\gamma p_{km}^{0} \quad (7)$$

Proposition 1 (counterfactual class invariance). If the decoder and re-encoders satisfy $E_c(R(c,n_\pi))=c$ and the classifier depends only on c, then $g_c(E_c(R(c,n_\pi)))=g_c(c)$ for every admissible nuisance intervention. This follows by substitution. Loss (4) is an empirical relaxation of the equality; causal-domain studies motivate the invariant-mechanism assumption [23, 24, 26].

Proposition 2 (bounded one-step prototype displacement). For $0\le\eta_t\le\eta_0\le1$, the unanchored update satisfies $\|p^{t+1}-p^t\|_2=\eta_t\|c_t-p^t\|_2\le\eta_0\|c_t-p^t\|_2$. After anchoring, $\|\bar{p}^{t+1}-p^0\|_2\le(1-\gamma)(\|p^t-p^0\|_2+\eta_0\|c_t-p^t\|_2)$. Hence $g_t=0$ gives zero data-driven displacement and γ contracts memory toward its source state. This explains why

uncertainty and anchoring can limit contamination, although it does not guarantee correctness of accepted pseudo-labels.

Proposition 3 (conditional calibration statement). Under exchangeability within predicted class k, the empirical quantile in (11) bounds the source-domain false-rejection rate near α up to finite-sample quantile discretization. Arbitrary temporal or adversarial shift breaks exchangeability; consequently no unconditional target coverage is claimed. The experiment measures, rather than assumes, robustness under shift [40, 41, 42].

4.8 Training objective and algorithms

$$L = L_cls + \lambda_1 L_env + \lambda_2 L_adv + \lambda_3 L_rec + \lambda_4 L_ind + \lambda_5 L_CF + \lambda_6 L_proto + \lambda_7 L_sep + \lambda_8 L_Brier \quad (19)$$

Cross-entropy trains the main and auxiliary class heads. L_env makes n environment-informative; gradient reversal implements L_adv so c becomes environment-uninformative. Reconstruction avoids destructive factorization; cross-covariance penalizes dependence; L_CF implements intervention invariance; prototype losses shape memory geometry; and Brier loss discourages overconfident probabilities.

4.9 Complexity

For batch size B, causal dimension d_c, K classes and M prototypes, novelty-distance evaluation costs O(BKMd_c) and stores O(KMd_c) prototype values. Windowed kernel drift costs $O(W^2 d)$ per evaluated window; the implementation evaluates every max(16,W/4) transactions with W=128. This is independent of the 2.9-million-row source corpus after training. DBSCAN is periodic and bounded to quarantined-memory analysis rather than the per-transaction path.

4.10 Experimental protocol

The benchmark contains 2,916,697 Bitcoin addresses, 10 published variables, 29 labels, and observations from 2011-2018 [1, 44]. The uploaded implementation treats white as a known class, then identifies criminal families with at least 30 pre-2015 observations and at least 15 observations from 2016 onward. It ranks eligible families by the smaller of their early and late supports and then by total support; the five best-supported families become known campaigns. Four additional families with the strongest late-period support are withheld in their entirety. Training uses data through 2014. Calibration uses 2015 source-known data; when a known class has fewer than 20 calibration instances, the latest 20% of that class's early history becomes a fallback source calibration set. Testing uses 2016-2018 known and held-out families in global chronological order. Sampling caps preserve tractability while retaining class-year coverage. Held-out labels never enter training, calibration, drift thresholds, or prototype updates.

Table 2. Strict global temporal protocol implemented by the uploaded revision.

Partition	Current-code eligibility	Label access	Sampling / operational rule
Training	Through 2014	White + five dynamically selected known ransomware families	Up to 6,000 white addresses; up to 3,000 per known family
Calibration	2015; fallback: latest 20% of early history if sparse	Known classes only	Up to 1,200 per class; novelty and drift thresholds
Evaluation	2016-2018	Known classes + four completely held-out families	Up to 1,200 per class-year; global chronological ordering
Full UCI corpus	2011-2018	29 labels; 2,916,697 addresses	Held-out labels used only for final metric calculation

Table 3. Principal FAST_MODE hyperparameters in the uploaded implementation.

Hyperparameter	FAST_MODE value	Purpose
Warm-up / full epochs	3 / 9	Representation then prototype-aware optimization
Batch size	512	CPU/GPU throughput
Causal / market dimensions	64 / 24	Compact factorization
Prototypes per class	3	Multimodal memory
Calibration α	0.05	Source false-rejection target
Drift window	128	Windowed attribution

Prototype EMA / anchor	0.04 / 0.01	Plasticity and retention
Seed	42	Fixed sampling/model seed; GPU kernels may still vary
Known / held-out ransomware families	5 / 4	Open-world class protocol

$$H = 2\,Acc_K\,Acc_U / (Acc_K + Acc_U)$$

$$F_i = \max_{t<T} A_i^t - A_i^T, \quad F = (T-1)^{-1} \Sigma_i F_i$$

Metrics include known accuracy and macro-F1, unknown recall, H-score, AUROC, AUPR, FPR at approximately 95% TPR, OSCR area, expected calibration error (ECE), Brier score, contamination robustness, continual macro-F1, and source-memory forgetting. H-score prevents a reject-all detector from appearing successful.

4.11 Version-controlled BitcoinHeist pilot results

Table 4. Artifact-verified predecessor BitcoinHeist pilot metrics; these values are not attributed to the revised strict year-cutoff protocol.

Known Acc.	Known F1	Unknown recall	H-score	AUROC	AUPR	FPR95	OSCR	ECE
0.2554	0.2136	0.1908	0.2184	0.5666	0.5294	0.8615	0.1797	0.1526

The benchmark figures in Sections 4.11-4.14 come from the archived predecessor build and quantify a related family-stratified pilot, not the revised strict year-cutoff protocol. The pilot is retained as auditable component evidence. Known accuracy is 0.2554, unknown-family recall 0.1908, H-score 0.2184, AUROC 0.5666, FPR95 0.8615, ECE 0.1526, and Brier score 0.7468. These values are inadequate for deployment but define an honest baseline for the confirmatory rerun.

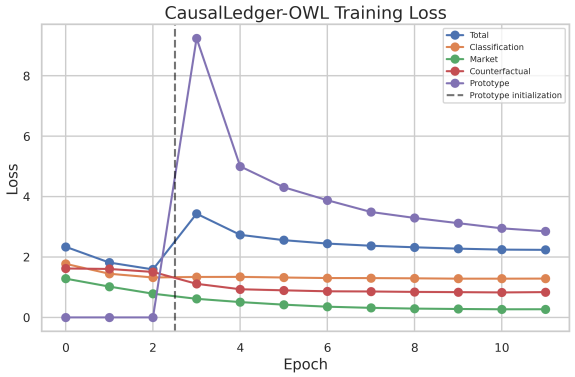


Figure 2. Training objectives in the predecessor FAST_MODE BitcoinHeist pilot. Prototype initialization after epoch 3 creates the expected loss discontinuity; subsequent prototype and counterfactual terms decline.

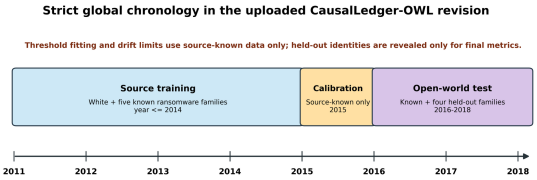


Figure 3. Strict global chronology in the uploaded revision. Known classes are learned through 2014, thresholds and drift limits are fitted on source-known data from 2015, and open-world evaluation covers 2016-2018. Held-out-family labels are accessed only for final metrics.

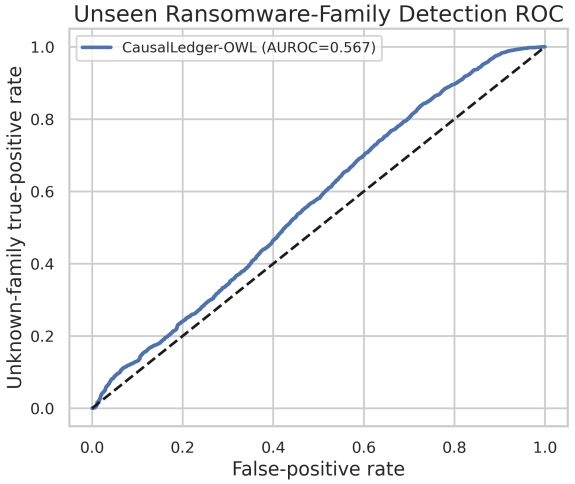


Figure 4. Unknown-family ROC curve in the predecessor BitcoinHeist pilot (AUROC 0.567). Ranking is only moderately above random and does not support operational deployment.

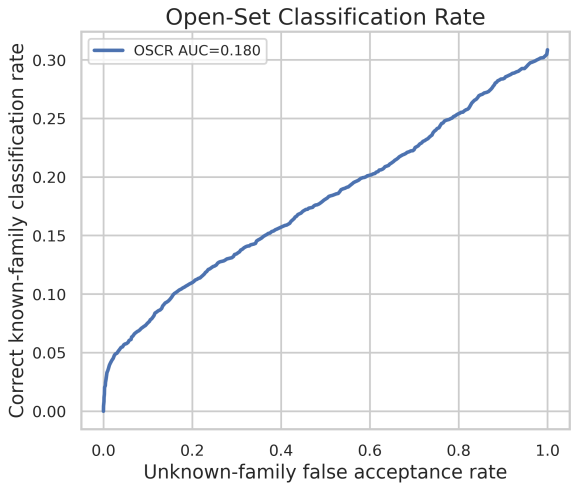


Figure 5. Open-set classification-rate curve in the predecessor pilot. OSCR area 0.180 reflects the joint difficulty of correct known-family classification and zero-day rejection.

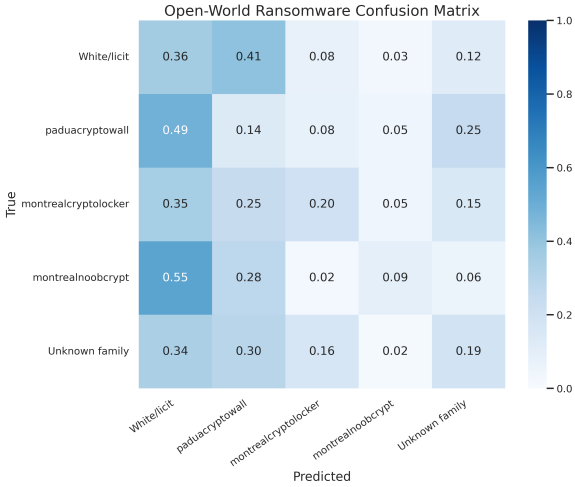


Figure 6. Normalized open-world confusion matrix in the predecessor pilot. Errors are localized across white, established ransomware campaigns, and the unified unknown-family decision.

4.12 Representation, prototypes, drift, and calibration

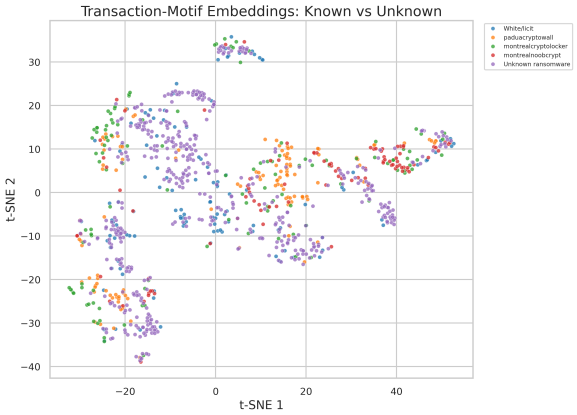


Figure 7. t-SNE projection of causal transaction-motif embeddings in the predecessor pilot. Held-out families overlap known modes, explaining the limited novelty AUROC.

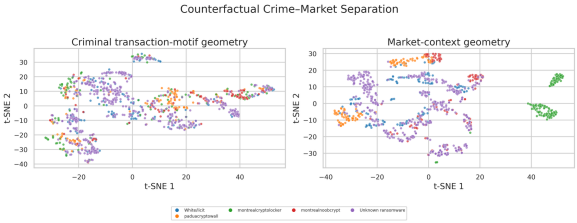


Figure 8. Causal-versus-nuisance embedding geometry in the predecessor pilot. Visual separation is qualitative; confirmatory invariance requires multi-seed statistics under the revised protocol.

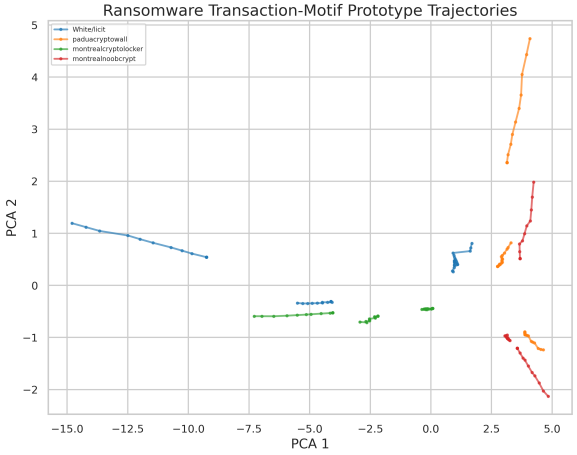


Figure 9. PCA projection of source and online prototype trajectories in the predecessor pilot. Source anchoring limits most trajectories, while selected modes move during crime-drift events.

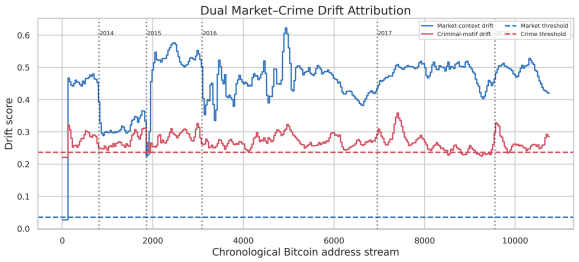


Figure 10. Dual market-crime drift timeline in the predecessor pilot. Market-context drift is persistently high whereas crime-motif drift is more localized, supporting attribution rather than one undifferentiated alarm.

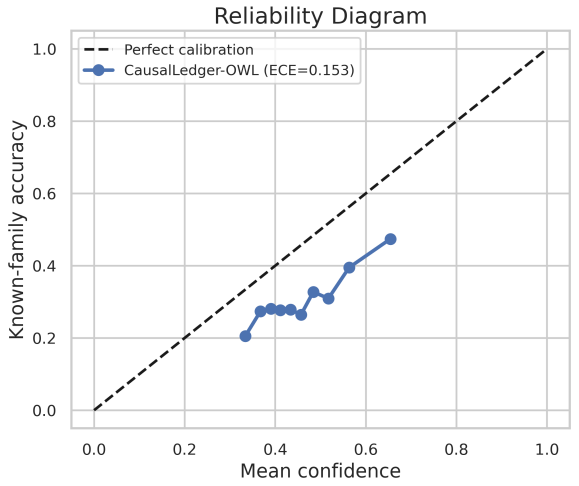


Figure 11. Known-class reliability diagram in the predecessor pilot. Accuracy is below confidence across bins, consistent with ECE 0.153 and motivating temperature scaling or adaptive calibration.

4.13 Sensitivity, comparison, and contamination analysis

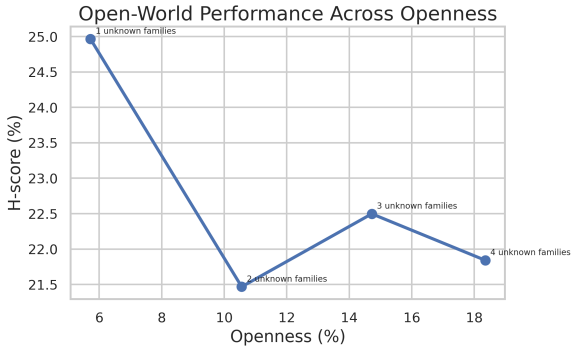


Figure 12. H-score versus openness in the predecessor pilot. Performance generally decreases as additional unseen families are introduced, with family-specific non-monotonicity.

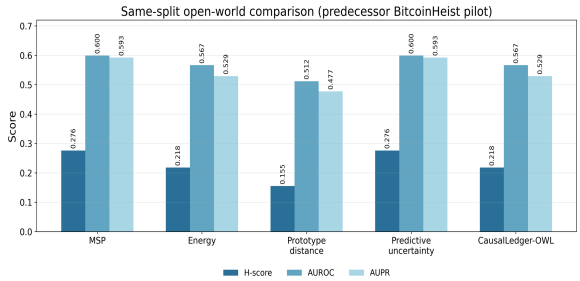


Figure 13. Same-split open-world comparison in the predecessor BitcoinHeist pilot. Calibrated MSP and predictive uncertainty exceed the predecessor composite score in H-score, AUROC, and AUPR; novelty-fusion superiority is therefore rejected.

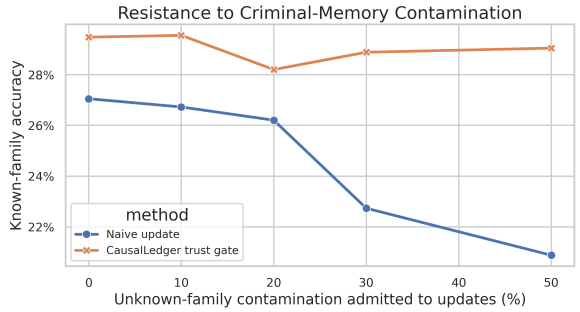


Figure 14. Unknown-contamination stress test in the predecessor pilot. At 50% contamination, the trust gate retains known accuracy of 0.2905 compared with 0.2088 for naive updating, an 8.17 percentage-point gain.

The contamination experiment provides the clearest positive result. At 20%, 30%, and 50% admitted unknowns, gated accuracy exceeds naive updating by 2.00, 6.15, and 8.17 percentage points, respectively. The gate is also higher at 10% in this run, but the substantive benefit appears as contamination increases. This pattern is consistent with Proposition 2: fewer low-trust samples produce data-driven prototype displacement.

4.14 Continual recognition and forgetting

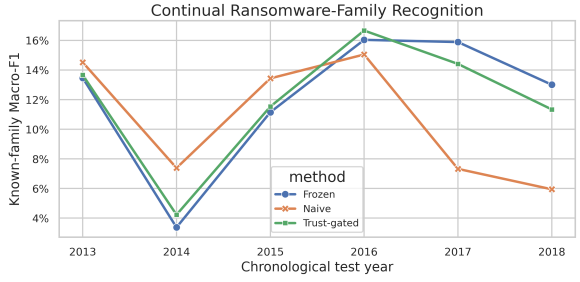


Figure 15. Chronological known-family macro-F1 in the predecessor pilot. Naive adaptation improves some early stages but later collapses; trust-gated memory remains close to the frozen model.

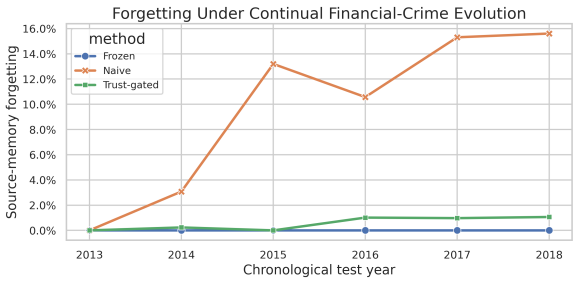


Figure 16. Source-memory forgetting in the predecessor pilot. Final forgetting is 0.1561 for naive adaptation and 0.0106 for trust-gated adaptation, a reduction of 0.1454 absolute.

The continual curves reveal a stability–plasticity trade-off. Naive updates can raise macro-F1 in 2014–2015 but later degrade both current performance and source retention. The trust gate sacrifices some short-term plasticity yet keeps final forgetting near 0.011. This supports the memory-protection mechanism, not broad recognition superiority.

5 Comparison with Recent Works

5.1 Capability comparison

Cross-paper numerical rankings are inappropriate because datasets, graph construction, unknown definitions, and temporal splits differ. Table 5 therefore compares capabilities, whereas Table 6 and Figure 13 report only same-split measurements from the archived predecessor pilot. The revised code changes the temporal protocol and novelty-weight search; its future benchmark results must be reported separately.

Table 5. Capability comparison with recent literature.

Method	Primary setting	Open family	Causal split	Drift attribution	Safe memory
Dib et al. [2]	Bitcoin ransomware classification	No	No	No	No
Guo et al. [7]	Graph covert-transaction detection	No	No	No	No
Tian et al. [10]	Adaptive GNN fraud detection	No	No	No	No
Shen et al. [14]	Graph continual anomaly detection	Limited	No	General	Continual
Wang et al. [33]	OOD/OSR benchmark analysis	Yes	No	No	No
Jiang et al. [37]	Unknown-support prototypes	Yes	No	No	Prototype
Yang et al. [20]	Self-supervised drift adaptation	Threat drift	No	General	Adaptive
Jin et al. [23]	Causal domain generalization	OOD domains	Yes	Environment	No
Obaidat et al. [45]	Chronological drift-aware IoT IDS	One held-out attack	No	Statistical drift	Conditional hybrid
CausalLedger-OWL	Bitcoin zero-day families	Yes	Yes	Market vs crime	Trust-gated

5.2 Same-split predecessor pilot baselines

Table 6. Same-split measurements from the predecessor BitcoinHeist pilot.

Method	Known Acc.	Unknown recall	H-score	AUROC	AUPR	FPR95
MSP	0.2590	0.2961	0.2763	0.5995	0.5926	0.9070
Energy	0.2554	0.1908	0.2184	0.5666	0.5294	0.8615
Prototype distance	0.2847	0.1069	0.1554	0.5121	0.4775	0.8984
Predictive uncertainty	0.2590	0.2961	0.2763	0.5995	0.5926	0.9070

CausalLedger-OWL	0.2554	0.1908	0.2184	0.5666	0.5294	0.8615

In the predecessor pilot, MSP is the strongest measured baseline: H-score 0.2763 and AUROC 0.5995, compared with 0.2184 and 0.5666 for the predecessor composite score. Prototype distance is weaker still (H-score 0.1554). The composite score improves FPR95 relative to MSP (0.8615 versus 0.9070), but this isolated benefit does not offset lower balanced detection. This negative finding directly motivated one revision in the uploaded code: pure single-component weight corners are removed, forcing the calibration search to evaluate mixed evidence. That design change remains a hypothesis until rerun on the UCI benchmark.

5.3 Why memory protection improves

The measured contamination benefit follows from three aligned constraints. Rejection removes samples above a class-conditional threshold; normalized entropy and tri-view agreement reduce unreliable pseudo-labels among accepted samples; and source anchoring contracts every accepted update toward its original prototype. Naive EMA treats proximity as sufficient evidence, so repeated unknown samples can pull a known prototype toward a novel-family mode. Market-only drift also freezes attack memory. These mechanisms explain the predecessor forgetting reduction from 0.1561 to 0.0106 without invoking an unsupported claim about novel-family ranking.

5.4 Validity threats and required paper-mode experiments

- Confirmatory execution: run the uploaded strict year-cutoff revision on the full UCI artifact for at least five seeds and report means, standard deviations, 95% confidence intervals, and paired significance tests.
- Label uncertainty: UCI states that white addresses are not known to be ransomware, not guaranteed licit. Sensitivity analyses should model asymmetric label noise.
- Protocol versioning: predecessor pilot metrics must not be attributed to the revised global-cutoff protocol. A final submission should replace or supplement them with an artifact archived from the exact uploaded code hash.
- Baseline breadth: evaluate calibrated logistic regression, random forest/XGBoost, Isolation Forest, one-class SVM, MSP/energy/OpenMax, a temporal GNN, and a graph-continual baseline under the identical family/time split.
- Calibration: compare global, class-conditional, temperature-scaled, EVT, adaptive, and conformal thresholds without using target-family identities.
- Graph access: address-level summaries are efficient but discard relational neighborhoods; reproduce the causal/context split on Elliptic or another public temporal transaction graph.
- Novelty selection: compare the revised mixed-weight search with leave-one-known-family-out source episodes and monotone score fusion; preregister the selection rule before inspecting held-out families.

6 Conclusion and Future Work

This study formulates cryptocurrency ransomware forensics as a joint problem of temporal known-family recognition, unknown-family rejection, drift attribution, and contamination-safe continual adaptation. CausalLedger-OWL separates transaction motifs from market context, applies counterfactual nuisance swapping, stores multimodal class prototypes, calibrates novelty using source-known evidence, distinguishes market from crime drift, and permits only bounded source-anchored memory updates. The uploaded revision further strengthens evaluation by enforcing global year cutoffs, five known ransomware families, four completely held-out families, normalized entropy, and non-degenerate mixed novelty weights. Syntax validation and full code-path execution produced all fifteen expected figures and artifacts.

The empirical conclusion remains deliberately bounded. The archived BitcoinHeist predecessor pilot supports the safe-memory mechanism: at 50% unknown contamination, gated known accuracy is 0.2905 versus 0.2088 for naive adaptation, and final forgetting falls from 0.1561 to 0.0106. It does not support superior zero-day ranking, because calibrated MSP achieves higher H-score and AUROC. Consequently, the paper contributes a falsifiable security architecture, a leakage-controlled confirmatory protocol, and evidence that update governance can be valuable even when novelty ranking is weak; it does not claim a deployment-ready ransomware detector.

Future work will execute the exact strict-cutoff revision in multi-seed paper mode; replace pseudo-open suppression with leave-one-family-out source episodes; learn monotone score fusion under nested source validation; add EVT and adaptive conformal calibration; integrate temporal graph encoders while preserving the causal/context split; maintain class-balanced replay; and evaluate analyst-assisted promotion of persistent DBSCAN clusters. Statistical tests should quantify both detection quality and memory displacement. Hardware-independent inference latency, energy cost, analyst workload, and

evidentiary traceability should also be reported before operational deployment.

Declarations

Funding: The author received no specific funding for this work.

Conflict of interest: The author declares no competing interests.

Data availability: The BitcoinHeist Ransomware Address dataset is publicly available from the UCI Machine Learning Repository under CC BY 4.0, DOI 10.24432/C5BG8V.

Code availability: The single-cell Python implementation corresponding to SHA-256 89614e1a93394976fc5a4f9293b97879e9400625229e11b6152d61dc1e25d8f2 is intended as the reproducible supplementary artifact. A permanent repository, environment lock file, benchmark output archive, and release DOI should accompany submission.

Ethics statement: The study uses public ledger-derived aggregate features and public ransomware-family labels; it does not attempt to deanonymize individual Bitcoin users. Operational use requires legal review, evidentiary safeguards, and human analyst confirmation.

References

[1] C. G. Akcora, Y. Li, Y. R. Gel, and M. Kantarcioglu. "BitcoinHeist: Topological Data Analysis for Ransomware Prediction on the Bitcoin Blockchain." Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, 4439-4445 (2020). https://doi.org/10.24963/ijcai.2020/612

[2] Omar Dib, Zhenghan Nan, Jinkua Liu. "Machine learning-based ransomware classification of Bitcoin transactions." Journal of King Saud University - Computer and Information Sciences 36(1), 101925 (2024). https://doi.org/10.1016/j.jksuci.2024.101925

[3] Suleiman Ali Alsaif. "Machine Learning-Based Ransomware Classification of Bitcoin Transactions." Applied Computational Intelligence and Soft Computing 2023, 1-10 (2023). https://doi.org/10.1155/2023/6274260

[4] Amjad Alraizza, Abdulmohsen Algarni. "Ransomware Detection Using Machine Learning: A Survey." Big Data and Cognitive Computing 7(3), 143 (2023). https://doi.org/10.3390/bdcc7030143

[5] Leonid Garin, Vladimir Gisin. "Machine learning in classifying bitcoin addresses." The Journal of Finance and Data Science 9, 100109 (2023). https://doi.org/10.1016/j.jfds.2023.100109

[6] Hardi Sabah Talabani, Hezha M.TAREQ Abdulhadi. "Bitcoin Ransomware Detection Employing Rule-Based Algorithms." Science Journal of University of Zakho 10(1), 5-10 (2022). https://doi.org/10.25271/sjuoz.2022.10.1.865

[7] Zhenyu Guo, Xin Li, Jiamou Liu, Zijian Zhang, Meng Li, Jingjing Hu, Liehuang Zhu. "Graph-Based Covert Transaction Detection and Protection in Blockchain." IEEE Transactions on Information Forensics and Security 19, 2244-2257 (2024). https://doi.org/10.1109/tifs.2023.3347895

[8] Jaehyeon Kim, Sejong Lee, Yushin Kim, Seyoung Ahn, Sunghyun Cho. "Graph Learning-Based Blockchain Phishing Account Detection with a Heterogeneous Transaction Graph." Sensors 23(1), 463 (2023). https://doi.org/10.3390/s23010463

[9] Zhiyuan Li, Enhan He. "Graph Neural Network-Based Bitcoin Transaction Tracking Model." IEEE Access 11, 62109-62120 (2023). https://doi.org/10.1109/access.2023.3288026

[10] Yue Tian, Guanjun Liu, Jiacun Wang, Mengchu Zhou. "ASA-GNN: Adaptive Sampling and Aggregation-Based Graph Neural Network for Transaction Fraud Detection." IEEE Transactions on Computational Social Systems 11(3), 3536-3549 (2024). https://doi.org/10.1109/tcss.2023.3335485

[11] Yong Zhou, Ding Bao, Ruoting Xiong, Wei Ren, Chenhuang Wu, Huimin Li, Hua Sun, Sheng Chen. "Next One: Predicting the Future Transaction of Illicit Address in Bitcoin Transaction Network by Graph Neural Network." Blockchain: Research and Applications, 100415 (2025). https://doi.org/10.1016/j.bcra.2025.100415

[12] Zening Zhao, Jinsong Wang, Jiajia Wei. "Graph neural network-based transaction link prediction method for public blockchain in heterogeneous information networks." Blockchain: Research and Applications 6(2), 100265 (2025). https://doi.org/10.1016/j.bcra.2024.100265

[13] Na Han, Ruixian Zhang, Xiaoyun Liu, Haining Zhang. "Illicit Bitcoin transaction detection via feature-gated temporal graph learning." Scientific Reports 16(1) (2026). https://doi.org/10.1038/s41598-026-53783-y

[14] Xiaodong Shen, Chang Xu, Liehuang Zhu. "Blockchain Anomaly Transaction Detection Method Based on Graph Continual Learning." IEEE Transactions on Network Science and Engineering 13, 6059-6078 (2026). https://doi.org/10.1109/tnse.2026.3653459

[15] Fernando E. Casado, Dylan Lema, Marcos F. Criado, Roberto Iglesias, Carlos V. Regueiro, Senén Barro. "Concept drift detection and adaptation for federated and continual learning."

Multimedia Tools and Applications 81(3), 3397-3419 (2021). https://doi.org/10.1007/s11042-021-11219-x

[16] Farah Jemili, Khaled Jouini, Ouajdi Korbaa. "Intrusion detection based on concept drift detection and online incremental learning." International Journal of Pervasive Computing and Communications 21(1), 81-115 (2024). https://doi.org/10.1108/ijpcc-12-2023-0358

[17] Sai Prasath, Kamalakanta Sethi, Dinesh Mohanty, Padmalochan Bera, Subhransu Ranjan Samantaray. "Analysis of Continual Learning Models for Intrusion Detection System." IEEE Access 10, 121444-121464 (2022). https://doi.org/10.1109/access.2022.3222715

[18] Chaoqun Guo, Xihan Li, Jubao Cheng, Shunjie Yang, Huiquan Gong. "Continual Learning for Intrusion Detection Under Evolving Network Threats." Future Internet 17(10), 456 (2025). https://doi.org/10.3390/fi17100456

[19] Renjie Chu, Peiyuan Jin, Hanli Qiao, Quanxi Feng. "Intrusion detection in the IoT data streams using concept drift localization." AIMS Mathematics 9(1), 1535-1561 (2023). https://doi.org/10.3934/math.2024076

[20] Shuo Yang, Xinran Zheng, Jinze Li, Jinfeng Xu, Xinchen Zhang, Edith C. H. Ngai. "Self-Supervised Adaptation Method to Concept Drift for Network Intrusion Detection." IEEE Transactions on Dependable and Secure Computing 22(6), 7632-7646 (2025). https://doi.org/10.1109/tdsc.2025.3599321

[21] Alessandra De Paola, Salvatore Drago, Pierluca Ferraro, Giuseppe Lo Re. "HOIDS: Concept drift aware hybrid online intrusion detection system." Journal of Network and Computer Applications 254, 104556 (2026). https://doi.org/10.1016/j.jnca.2026.104556

[22] Yan Liu, Weina Niu, Hongkai Li, Mingze He, Xuyang Ding, Jiacheng Gong, Ran Yan, Zhenglong Li, Xiaosong Zhang. "RepShield: Robust knowledge representation in continual learning for network intrusion detection." Computer Networks 283, 112308 (2026). https://doi.org/10.1016/j.comnet.2026.112308

[23] Xuanyu Jin, Ni Li, Wanzeng Kong, Jiajia Tang, Bing Yang. "Unbiased Semantic Representation Learning Based on Causal Disentanglement for Domain Generalization." ACM Transactions on Multimedia Computing, Communications, and Applications 20(8), 1-20 (2024). https://doi.org/10.1145/3659953

[24] Youjia Shao, Shaohui Wang, Wencang Zhao. "CaRGI: Causal semantic representation learning via generative intervention for single domain generalization." Applied Soft Computing 173, 112910 (2025). https://doi.org/10.1016/j.asoc.2025.112910

[25] Xianglin Zuo, Baohang Wei, Hao Yuan, Ying Wang. "Graph causal representation learning for out-of-distribution generalization." Neural Networks 195, 108208 (2026). https://doi.org/10.1016/j.neunet.2025.108208

[26] Dianlong You, Bingxin Liu, Dongyan Wang, Xiaoyi Ge, Zhen Chen, Xindong Wu. "Cross-domain representation learning with causal invariance." Engineering Applications of Artificial Intelligence 162, 112699 (2025). https://doi.org/10.1016/j.engappai.2025.112699

[27] Chen Jia, Yue Zhang. "Meta-learning the invariant representation for domain generalization." Machine Learning 113(4), 1661-1681 (2022). https://doi.org/10.1007/s10994-022-06256-y

[28] Yuxuan Wang, Muli Yang, Aming Wu, Cheng Deng. "Progressive Invariant Causal Feature Learning for Single Domain Generalization." IEEE Transactions on Image Processing 34, 2694-2706 (2025). https://doi.org/10.1109/tip.2025.3563772

[29] Shanshan Li, Qingjie Zhao, Changchun Zhang, Yuanbing Zou. "Deep discriminative causal domain generalization." Information Sciences 645, 119335 (2023). https://doi.org/10.1016/j.ins.2023.119335

[30] Shanshan Li, Qingjie Zhao, Baosheng Sun, Xin Wang, Yuanbing Zou. "Domain generalization via causal fine-grained feature decomposition and learning." Computers and Electrical Engineering 119, 109548 (2024). https://doi.org/10.1016/j.compeleceng.2024.109548

[31] Ning Wang, Huiling Wang, Shaocong Yang, Huan Chu, Shi Dong, Wattana Viriyasitavat. "Semi-supervised incremental domain generalization learning based on causal invariance." International Journal of Machine Learning and Cybernetics 15(10), 4815-4828 (2024). https://doi.org/10.1007/s13042-024-02199-z

[32] Jiao Zhang, Xu-Yao Zhang, Chuang Wang, Cheng-Lin Liu. "Deep representation learning for domain generalization with information bottleneck principle." Pattern Recognition 143, 109737 (2023). https://doi.org/10.1016/j.patcog.2023.109737

[33] Hongjun Wang, Sagar Vaze, Kai Han. "Dissecting Out-of-Distribution Detection and Open-Set Recognition: A Critical Analysis of Methods and Benchmarks." International Journal of Computer Vision 133(3), 1326-1351

(2024). https://doi.org/10.1007/s11263-024-02222-4

[34] Rundong He, Zhongyi Han, Xiushan Nie, Yilong Yin, Xiaojun Chang. "Visual Out-of-Distribution Detection in Open-Set Noisy Environments." International Journal of Computer Vision 132(11), 5453-5470 (2024). https://doi.org/10.1007/s11263-024-02139-y

[35] Ziheng Xia, Penghui Wang, Ganggang Dong, Hongwei Liu. "Radar HRRP Open Set Recognition Based on Extreme Value Distribution." IEEE Transactions on Geoscience and Remote Sensing 61, 1-16 (2023). https://doi.org/10.1109/tgrs.2023.3257879

[36] Zhun-ga Liu, Yi-min Fu, Quan Pan, Zuo-wei Zhang. "Orientational Distribution Learning with Hierarchical Spatial Attention for Open Set Recognition." IEEE Transactions on Pattern Analysis and Machine Intelligence, 1-16 (2022). https://doi.org/10.1109/tpami.2022.3227913

[37] Guosong Jiang, Pengfei Zhu, Bing Cao, Dongyue Chen, Qinghua Hu. "Unknown Support Prototype Set for Open Set Recognition." International Journal of Computer Vision 133(7), 4366-4383 (2025). https://doi.org/10.1007/s11263-025-02384-9

[38] Shu Kong, Deva Ramanan. "OpenGAN: Open-Set Recognition via Open Data Generation." IEEE Transactions on Pattern Analysis and Machine Intelligence 47(5), 3233-3243 (2025). https://doi.org/10.1109/tpami.2022.3184052

[39] Zhen Yang, Jun Yue, Pedram Ghamisi, Shiliang Zhang, Jiayi Ma, Leyuan Fang. "Open Set Recognition in Real World." International Journal of Computer Vision 132(8), 3208-3231 (2024). https://doi.org/10.1007/s11263-024-02015-9

[40] Arya Saboury, Mustafa Kemal Uyguroglu. "Uncertainty-Aware Real-Time Visual Anomaly Detection With Conformal Prediction in Dynamic Indoor Environments." IEEE Robotics and Automation Letters 10(5), 4468-4475 (2025). https://doi.org/10.1109/lra.2025.3552318

[41] Jian Zheng, Jingjing Sun, Qiankun Liao, Mengyun Xie, Shengye Wang. "An anomaly detection method for conformal mechanism." Egyptian Informatics Journal 32, 100824 (2025). https://doi.org/10.1016/j.eij.2025.100824

[42] Ranjit, K. C., Shrestha, B., & Ghimire, U. (2026). Optimizing CNNArchitectures for Steering Angle Prediction for Self-Driving Vehicles in Unstructured Roads: A Comparative Study of Activation Functions and Model Complexity. ICCK Transactions on Machine Intelligence, 2(2), 88–99. https://doi.org/10.62762/TMI.2025.759110

[43] Kumar, A., & Lehal, G. S. (2026). Detection of Newspaper Layouts Using YOLO12. ICCK Transactions on Machine Intelligence, 2(2), 77–87. https://doi.org/10.62762/TMI.2025.846033

[44] Wang,X., & Zhao,R.(2026). FedTLRec: FederatedRecommendation with Transformer-based Parameter Aggregation and LoRA Compression. ICCK Transactions on Machine Intelligence, 2(1), 65–76. https://doi.org/10.62762/TMI.2025.882476

[45] Arya, S., Anju, & Onah, J. N. (2026). A Data-Driven Framework for Methane Emission Prediction Using Machine Learning Methods. ICCK Transactions on Machine Intelligence, 2(1), 53–64. https://doi.org/10.62762/TMI.2025.782852

[46] Bai, Y., Hu, J., Tursun, E. & Yimit, H. (2026). Data-Driven Operational Assessment Method and Digital Twin System for Unmanned Surface Vehicles. ICCK Transactions on Machine Intelligence, 2(1), 38–52. https://doi.org/10.62762/TMI.2025.444910

[47] Agarwal, S. K., & Gupta, Y. K. (2026). A Novel Approach of Progressive Transfer Learning for MRI Brain Tumor Classification Using VGG16 and MobileNet Architectures. ICCK Transactions on Machine Intelligence, 2(1), 28–37. https://doi.org/10.62762/TMI.2025.367009

[48] Kumar, V., Saxena, A., & Chawla, L. (2026). Adaptive Learning Density Estimators for Tsallis Entropy and Kapur Entropy with Applications in System Training. ICCK Transactions on Machine Intelligence, 2(1), 12–27. https://doi.org/10.62762/TMI.2025.317970

[49] Agarwal, S., Kandpal, M., Gupta, V. K., Aggarwal, S., & Jain, P. (2025). Secure and Decentralized Heart Sound Analysis using Federated Learning and Blockchain Technology. ICCK Transactions on Machine Intelligence, 2(1), 1–11. https://doi.org/10.62762/TMI.2025.567350

[50] Pereira, M. S., de Azevedo, T. M., Ribeiro, C. T., Freire, A. I., Francisco, M. B., Pereira, J. L. J., & de Paiva, A. P. (2025). Discriminating Planted Capsicum Spp. Varieties via Machine Learning and Multivariate Data Reduction. ICCK Transactions on Machine Intelligence, 1(3), 166–185. https://doi.org/10.62762/TMI.2025.385133

[51] Kumar, A. (2025). Clinical Text Analytics: Techniques, Deep Learning Models, and the Future of Medical Text Analytics. ICCK Transactions on Machine Intelligence, 1(3),

148–165. https://doi.org/10.62762/TMI.2025.451731

[52] Malik, I. A., & Habib, G. (2025). AI Enabled Resource-Constrained Computing Architectures for IoT Devices. ICCK Transactions on Machine Intelligence, 1(3), 138–147. https://doi.org/10.62762/TMI.2025.225921

[53] Bmotra, M., Kumar, R., & Dhiman, G. (2025). Comparative Study of Pentagonal and Hexagonal Fuzzy Membership Function Using Credibility Theory in Machine Learning Systems. ICCK Transactions on Machine Intelligence, 1(3), 127–137. https://doi.org/10.62762/TMI.2025.922612

[54] Aggarwal, V., Aggarwal, S., Bhardwaj, A., & Gupta, V. K. (2025). Quantum Computing Essentials: Bridging Theory and Practice for New Learners. ICCK Transactions on Machine Intelligence, 1(3), 117–126. https://doi.org/10.62762/TMI.2025.173543

[55] Sarvakar, K., & Rana, K. (2025). A Hybrid Framework Combining CNN, LSTM, and Transfer Learning for Emotion Recognition. ICCK Transactions on Machine Intelligence, 1(2), 103–116. https://doi.org/10.62762/TMI.2025.572412

[56] Singh, S., Aggarwal, S., Singh, A., & Sharma, A. (2025). Integrating Artificial Intelligence and Machine Learning in Autism Detection via Gut Microbiome Analysis. ICCK Transactions on Machine Intelligence, 1(2), 90–102. https://doi.org/10.62762/TMI.2025.682666

[57] Sharma, S., Devgan, K., Jangra, D., Bhardwaj, A., & Aggarwal, S. (2025). Neuro-Inspired Alert System for Air Quality Prediction Using Ensemble Preprocessing and SNN Classification. ICCK Transactions on Machine Intelligence, 1(2), 69–79. https://doi.org/10.62762/TMI.2025.403059

[58] Khan, A., & Singh, J. (2025). A Novel Image Captioning Technique Using Deep Learning Methodology. ICCK Transactions on Machine Intelligence, 1(2), 52–68. https://doi.org/10.62762/TMI.2025.886122

[59] Khan, M., Kumar, R., & Dhiman, G. (2025). A Hybrid Machine Learning Fuzzy Non-linear Regression Approach for Neutrosophic Fuzzy Set. ICCK Transactions on Machine Intelligence, 1(1), 42–51. https://doi.org/10.62762/TMI.2025.561363

[60] Garg, M. (2025). Enhanced Reinforcement Learning-Based Resource Scheduling for Secure Blockchain Networks in IIoT. ICCK Transactions on Machine Intelligence, 1(1), 29–41. https://doi.org/10.62762/TMI.2024.529242

[61] Gupta, S., Adhikari, U., Roy, D., & Hazra, S. (2025). IoT-Integrated Reinforcement Learning-Based Mine Detection System for Military and Humanitarian Applications. ICCK Transactions on Machine Intelligence, 1(1), 17–28. https://doi.org/10.62762/TMI.2025.235880

[62] Myakala, P. K., Kamatala, S., & Bura, C. (2025). Privacy-Preserving Federated Learning for IoT Botnet Detection: A Federated Averaging Approach. ICCK Transactions on Machine Intelligence, 1(1), 6–16. https://doi.org/10.62762/TMI.2025.796490

[63] Tiwari, S., Kumar, D., Mahajan, A., & Sachar, S. (2025). Emotion Detection from Speech Using CNN-BiLSTM with Feature Rich Audio Inputs. ICCK Transactions on Machine Intelligence, 1(2), 80–89. https://doi.org/10.62762/TMI.2025.306750

Appendix A. Reproducibility Audit

Table A1. Version-controlled implementation and reproducibility audit.

Audit item	Status
Uploaded source syntax	Passed (python -m py_compile)
Uploaded strict-protocol logic	Reviewed: train <= 2014; calibrate 2015; test >= 2016
End-to-end implementation smoke test	Passed on 2,658 schema-faithful rows; checkpoint, CSVs, ZIP generated
Required plots	15/15 generated and file-verified in the smoke execution
Held-out labels in representation training	None by code path
Held-out labels in novelty/drift calibration	None by code path
Held-out labels in online update decisions	None by code path
Current UCI benchmark execution in this workspace	External dataset/package retrieval unavailable; no benchmark score fabricated

Audit item	Status
Archived predecessor UCI pilot	Executed on 26,740 selected BitcoinHeist addresses; figures and tables retained as pilot evidence
Confirmatory paper mode	Full current-code benchmark, >=5 seeds, confidence intervals and paired tests required

The uploaded code fixes the random seed, fits all feature scalers on training positions, calibrates novelty and drift only on source-known data, and uses held-out identities solely for final evaluation. A reduced schema-faithful fixture was used only to execute every code path after this workspace could not retrieve the external UCI artifact; its metric values are intentionally excluded from the scientific results. The predecessor benchmark figures remain separately versioned and are never described as outputs of the revised global-cutoff code.