Blockchain Fraud Detection for Explainable Forensic Investigations

Mario Trerotola[1], Davide Calvaresi[2], and Mimmo Parente[3]

[1] Politecnico di Torino, Torino, Italy
mario.trerotola@polito.it
[2] University of Applied Sciences and Arts Western Switzerland (HES-SO), Sion, Switzerland
davide.calvaresi@hevs.ch
[3] Università degli Studi di Salerno, Salerno, Italy
parente@unisa.it

Abstract. In 2024, cryptocurrency-related fraud exceeded $40 billion, challenging UN Sustainable Development Goal 16.4 on illicit financial flows. Existing methods face a recurring trade-off: deep models are accurate but opaque, while interpretable pipelines are often chain-specific. We propose an explainable graph-temporal Machine Learning framework whose methodology is reusable across heterogeneous ledgers by relying on chain-specific feature extractors, instantiated in this work on Bitcoin and Ethereum. The method combines architecture-neutral structural, monetary, and temporal features; Balanced Random Forests for severe class imbalance; and a three-layer XAI stack (SHAP, CIU, DEXiRE) for global, local, and symbolic auditing. ChainAbuse intelligence (15k+ reports) supports periodic retraining as fraud patterns evolve. Evaluated on 31,000 Bitcoin wallets and 157,000 Ethereum tokens, the framework achieves macro-F_1 up to 0.92 and 0.985, respectively, while preserving full transparency.

Keywords: Blockchain Forensics · Explainable AI · Graph-Temporal Features · Fraud Detection.

1 Introduction

Blockchain technology enables large-scale value transfers without intermediaries, but pseudonymity also facilitates illicit activity. In 2024, cryptocurrency-related fraud exceeded $40 billion [4], challenging UN Sustainable Development Goal 16.4 [34]. We ask whether a single forensic framework can classify licit and illicit behavior across heterogeneous ledgers while remaining interpretable, efficient, and reproducible. We focus on Bitcoin's UTXO model and Ethereum's account-based model. Deep graph models [36,27,37] are accurate but opaque; feature-engineered methods [24,5,33,15,37] are transparent but often chain-specific. Recent work highlights unresolved transferability and efficiency issues [14]. We target the regulatory trilemma of accuracy, interpretability, and deployability under

GDPR and the EU AI Act. The framework targets ex-post forensic classification (reconstruction of behavioural profiles for entities with an already-observed activity window) rather than streaming detection.

Our explainable-by-design pipeline has four stages: (i) label harmonization with Confident Learning [28]; (ii) architecture-neutral graph-temporal feature engineering; (iii) class-imbalance-aware training with Balanced Random Forests; and (iv) layered explanations via SHAP [22], CIU [1], and DEXiRE [7]. We evaluate on 31,000 Bitcoin wallets and 157,000 ERC-20 tokens and obtain performance comparable to strong neural baselines while preserving full auditability.

Our contributions are threefold: a methodologically portable forensic pipeline whose feature taxonomy and XAI stack instantiate on heterogeneous ledgers via chain-specific adapters, empirical evidence that interpretable ensembles can match deep approaches, and cross-ledger fraud signatures (dispersion, centrality concentration, temporal burstiness) that appear robust to data model differences.

Related Works. Research on illicit activity detection spans several complementary directions. *Feature-engineered pipelines* remain widely used due to their interpretability and low operational overhead. In the Bitcoin context, this includes early work on address clustering and transactional pattern analysis [23,41], unsupervised and multi-view fraud detection [24,25], transaction-level risk scoring [26], service attribution [33], entity ownership inference [5], analyses of money flow structures [32,29,12], and wallet-centric fraud mitigation [15]. Work on learning under limited labels has also been investigated [21]. For Ethereum, hand-crafted features have been used for detection of Ponzi schemes [3,6], honeypot detection [31], analysis of scam token behavior [40], and smart contract vulnerability discovery [16].

Graph representation learning approaches [36,27] enhance detection performance, but this improvement comes at the expense of diminished auditability. TokenScout [37] utilizes Temporal GNNs for early scam detection. Approaches including FlowScope [17], DenseFlow [19], and HoloScope [20] are specifically designed for laundering detection. Subsequent work extends these ideas to EVM-based tracing [14,13,38], mixer investigations [8,11,30,39], phishing [10], giveaway scams [18], wash trading [35], and DeFi exploit analysis [42].

Our work contributes by integrating architecture-agnostic graph-temporal features with interpretable ensemble models to enable transfer across different ledgers. The paper is organized as follows: Section 2 details the methodology; Sections 3 and 4 present Bitcoin and ERC-20 case studies; Section 5 discusses cross-architecture insights; Section 6 concludes.

2 Methodology

Our central hypothesis is that illicit behavior exhibits architecture-invariant signatures (structural concentration, temporal burstiness, value dispersion), regardless of ledger mechanics.

Data Harmonization and Label Quality. We merge labels from TokenScout [37], ChainAbuse, and Elliptic++ with a conservative rule: an entity e is fraud if any source flags it as fraud ($\widetilde{L}_e = \text{fraud} \iff \exists l \in L_e : l = \text{fraud}$), thus minimizing false negatives [9]. To reduce label noise, we apply Confident Learning and compute $s_i = P(y_i \mid \mathbf{x}_i)$ via k-fold cross-validation; samples with $s_i < 0.3$ are removed as likely mislabeled.

Graph-Temporal Feature Engineering. We model transactions as directed graphs and extract three feature families: *structural* (centrality and topology), *monetary* (value concentration and fragmentation), and *temporal* (velocity, burstiness, lifetime). Fraud profiles typically combine abnormal flow concentration and short activity windows. Feature semantics remain stable across ledgers; only extraction differs (UTXO link tracing for Bitcoin, `Transfer` log parsing for ERC-20).

Ensemble Classification with Class Balancing. Since fraud data are highly imbalanced, we train Balanced Random Forests, which rebalance bootstrap samples internally. We tune hyperparameters with stratified five-fold cross-validation, optimizing macro-F_1 and AUROC.

Multi-Resolution Explainability Stack. We combine three complementary layers: SHAP for global feature attribution (path-dependent TreeSHAP under residual feature dependence), CIU for instance-level contextual reasoning, and DEXiRE for symbolic IF–THEN rules readable by auditors. This stack supports transparency requirements under GDPR Article 22 and the EU AI Act. For reproducibility, the implementation is available on an anonymous link [2].

3 Case Study I: Bitcoin Wallet Classification

Bitcoin's UTXO structure complicates forensics because addresses are transient states rather than persistent entities. We therefore cluster addresses into wallets and trace multi-hop flows to detect darknet payments, ransomware, mixers, and scam activity.

3.1 Data Acquisition and Feature Engineering

Ground-truth labels come from Elliptic++ (14,266 fraudulent and 7,378 licit addresses), ChainAbuse (6,911 validated fraud reports), and a recent Q1 2025 sample (8,834 addresses). We retain only addresses with at least two incoming and two outgoing transactions; for the recent sample, an Isolation Forest (5% contamination) removes suspicious outliers. Table 1 summarizes the per-source composition; after the connectivity filter and outlier removal, the final working dataset comprises 31,389 wallets.

For each address, we extract 20 features in three groups: structural, monetary, and temporal. The most discriminative are `unique_out_ratio` and `time_interval_ratio`, which capture rapid dispersion and bursty behavior typical of peel-chain laundering. Table 2 compares this vocabulary with prior work.

Table 1: Bitcoin wallet dataset composition by source, before connectivity filter (≥2 in/out transactions) and outlier removal (Isolation Forest, 5% contamination) that yield the final 31,389-wallet working dataset. Data spans 2019–2025.

Source	**Fraud**	**Licit**
Elliptic++	14,266	7,378
ChainAbuse	6,911	–
Recent Sample (filtered)	–	8,834
Total	21,177	16,212

Table 2: Bitcoin feature comparison with previous research. Letters show which studies employed each feature; bold entries mark novel, highly discriminative contributions.

Feature	**Used in**
Structural (7)	
In/Out-degree	A,B,C,E
Unique counterparties (in/out)	B,E
Local clustering coefficient	A,E
Total wallet degrees	C,E
Monetary (6)	
Average transaction amount (in/out)	B,C,D,E
Net balance (address/wallet)	B,C,E
Total received/sent volumes	C,D,E
Weighted average transaction	C,D,E
Temporal (7)	
Inter-transaction intervals	B,C,E
Address lifetime	C,E
Average active duration	C,E
Activity index (tx/lifetime)	B,C,D,E
In/Out transaction count ratio	B,C,E
Unique-out ratio (dispersion)	**A,E**
Time-interval ratio (burstiness)	**C,E**
Total feature count per study	**A:3, B:7, C:12, D:4, E:20**

Legend: A=Chang et al. (2018), B=Weber et al. (2019), C=Chen et al. (2021), D=Işcan et al. (2023), E=Our work. Gray rows highlight novel dispersion-specific features critical for detecting peel chains and automated laundering.

3.2 Experimental Results

We evaluate three provenance setups: (1) ChainAbuse vs recent on-chain sample, (2) ChainAbuse vs Elliptic++ licit addresses, and (3) Elliptic++ only. Each scenario uses a held-out 80:20 stratified split with SMOTE applied within training only; 5-fold CV on Scenario 2 yields 0.918 ± 0.006 macro-F_1 at tuning. Table 3 reports held-out metrics.

Table 3 shows that Random Forest performs best in Scenario 2, with macro-$F_1 = 0.92$ on the held-out test set. Confident Learning is applied only to the ChainAbuse fraud subset of Scenario 2's training partition before SMOTE, flagging 294 records; re-evaluated on the same test partition, macro-F_1 improves from 0.912 to 0.920.

The Scenario 1 vs. 2–3 gap (F_1 0.79 vs. 0.92) is driven by the licit reference: Scenario 1 replaces the vetted Elliptic++ set with a Q1 2025 on-chain sample filtered only by a 5% Isolation Forest. We thus read Scenario 1 as a low-curation cold-start and Scenarios 2–3 as the curated steady state, bracketing the operational range.

Table 3: Bitcoin fraud classification across three data source scenarios. Metrics: macro-averaged precision, recall, F_1, and accuracy. Best per-scenario results in bold.

Sc.	Model	Prec	Rec	F_1	Acc
1	RF	0.78	0.79	0.79	0.82
	MLP	0.71	0.73	0.72	0.75
	SVM	0.69	0.70	0.69	0.73
2	RF	**0.93**	**0.91**	**0.92**	**0.92**
	MLP	0.84	0.85	0.85	0.85
	SVM	0.81	0.82	0.82	0.82
3	RF	0.92	0.92	0.92	0.90
	MLP	0.85	0.84	0.84	0.84
	SVM	0.80	0.81	0.80	0.81

Table 4: Positioning within the literature. Metrics for prior methods are reproduced from the corresponding original publications and are evaluated on the respective original datasets and splits; baseline architectures are not re-trained on our data. XAI column: ✓=full explainability, ∼=partial, none=no explainability. BA=Balanced Accuracy.

Chain	Method	Model	Acc/BA	F_1	XAI
BTC	Chang (2018)	RF/SVM	89%	–	✓
	Nerurkar (2021)	CNN	91%	–	–
	Işcan (2023)	LightGBM	93%	–	∼
	Ours	RF+XAI	92%	0.92	✓
ETH	TokenScout	T-GNN	98.4%	–	–
	Chen (2021)	GCN	94.2%	0.93	–
	Ours	RF+XAI	98.1%	0.985	✓

3.3 Explainability Analysis

SHAP [22] decomposes predictions into additive feature effects, i.e., $f(x) = \phi_0 + \sum_i \phi_i$, where positive ϕ_i values move the decision toward fraud. In Fig. 1, red points on the right indicate that high feature values increase fraud risk, while blue-right patterns indicate variables where low values are suspicious. The dominant signal is `unique_out_ratio`: high dispersion strongly increases fraud probability. `time_interval_ratio` captures temporal burstiness, and low `weighted_avg_tx` plus low `avg_active_duration` characterize short-lived wallets with small repetitive transfers. Overall, the model highlights patterns consistent with peel-chain and mixer operations. Table 5 reports the top features.

For instance-level analysis, CIU yields $\overline{\text{CI}} \approx 0.067$ and $\overline{\text{CU}} \approx 0.856$. CI (Contextual Importance) quantifies how much a feature can influence the local prediction, while CU (Contextual Utility) measures whether its observed value supports the predicted class. These values indicate that top features provide stable local support for fraud classifications. To make this evidence operational for audits, DEXiRE [7] extracts symbolic IF–THEN rules from the trained ensemble. We obtain 77 rules (44 fraud, 33 licit) with 89% surrogate accuracy, showing good fidelity to the original model. A representative rule is: IF `unique_out_ratio`> 0.7 AND `lifetime`$<$ 48h THEN fraud (confidence 0.94).

Operational Implications. The Bitcoin results show that explainable ensembles can support day-to-day forensic workflows without deep-model opacity. In practice, investigators can use SHAP to prioritize high-risk wallets, CIU to test

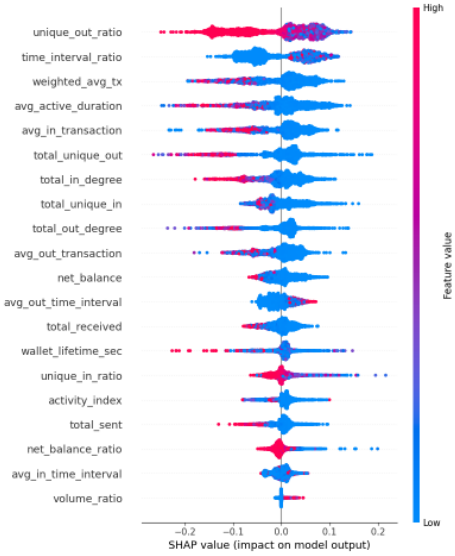

Fig. 1: SHAP beeswarm plot for classifying Bitcoin wallets. Each point corresponds to one wallet; horizontal position is SHAP impact and color encodes feature value (red = high, blue = low).

Table 5: Top-10 Bitcoin features by SHAP importance.

#	**Feature**	$\lvert\bar{\phi}\rvert$	**Fraud Signal**
1	unique_out_ratio	0.142	High ↑
2	time_interval_ratio	0.118	High ↑
3	weighted_avg_tx	0.097	Low ↓
4	avg_active_duration	0.089	Low ↓
5	avg_in_transaction	0.081	Low ↓
6	total_unique_out	0.076	High ↑
7	total_in_degree	0.068	Low ↓
8	total_unique_in	0.062	Low ↓
9	total_out_degree	0.058	High ↑
10	avg_out_transaction	0.054	Low ↓

alternative hypotheses at case level, and DEXiRE rules to produce courtroom-readable justifications. This is particularly relevant for financial intelligence units that need reproducible evidence, low false negatives, and transparent decision logic under regulatory scrutiny.

4 Case Study II: ERC-20 Token Scam Detection

Unlike Bitcoin's stateless UTXOs, Ethereum maintains persistent account states and programmable contracts. This programmability creates richer behavioral traces but also a broader attack surface. Since ERC-20 compliance is easy to satisfy syntactically, malicious actors can deploy tokens that look legitimate at interface level while embedding deceptive economic behavior (rug pulls, honeypots, or Ponzi-like incentives). As a result, detection must rely less on static metadata and more on dynamic transaction-graph behavior over time.

Table 6: ERC-20 token dataset by provenance. Fraudulent tokens from community reports; legitimate tokens from market-cap ranking and academic verification.

Category	Source	Tokens
Fraudulent	TokenScout (EOA-linked)	142,275
	ChainAbuse (direct)	11,636
Legitimate	CoinMarketCap (top MCap)	2,003
	TokenScout (academic)	1,805
Total unique tokens		**157,719**

4.1 Data Acquisition and Feature Engineering

We build the ERC-20 dataset from four sources (Table 6), grouped by provenance and trust assumptions. Fraud labels come from TokenScout EOA-linked tracing (142,275 tokens) and direct ChainAbuse reports (11,636 tokens). The target scam typologies are deployer- and contract-level fraud (rug pulls, honeypots, Ponzi-like deployments, pump-and-dump schemes) rather than intra-token market manipulation. Labeled-licit references come from CoinMarketCap top-cap assets (2,003 tokens), used as a conservative market-quality proxy, and TokenScout's academic benchmark subset (1,805 tokens) [37]. The choice of CoinMarketCap top-cap is deliberate: target typologies have short token lifetimes (days to a few months), whereas top-cap tokens have been actively traded for over twelve months at sampling. This design lets us explicitly test how source curation and class purity affect downstream classification performance.

For each token, we build a directed transfer graph and define 47 semantic feature descriptors in three families: structural (17), monetary (18), and temporal (12). Descriptors marked with $*$ are expanded through aggregate statistics (min, max, mean, std), yielding a 145-column matrix; iterative collinearity pruning (drop the weaker-target feature in each highest-$|\rho|$ pair until $|\rho| < 0.7$) converges to a 17-feature representation used for the final ERC-20 model and XAI. Temporal splits between short- and long-horizon activity are especially informative because rug pulls are typically short-lived. Table 7 compares our semantic feature vocabulary with TokenScout [37].

4.2 Experimental Results

Data provenance strongly affects performance. We evaluate four source combinations to isolate the impact of fraud-source quality and licit-reference quality, then rank scenarios by $(\text{AUC-PR} + F_2)/2$. This composite criterion is recall-oriented, reflecting a realistic operational preference: missing active scams is generally more costly than temporarily flagging a legitimate token for manual review. For all ERC-20 scenarios, model training and evaluation are performed after the collinearity-pruning step, so the reported metrics refer to the final 17-feature representation rather than to the full candidate feature matrix.

The best setup (CA + CMC) reaches $F_1 = 0.985$, balanced accuracy = 98.1%, and FPR = 1%; the collinearity-aware reduction preserves predictive performance while removing variables that could distort attribution-based explanations. As shown in Table 4, this places our framework within the top tier of

Table 7: ERC-20 semantic feature descriptors compared against TokenScout. Gray rows mark descriptors introduced in this work; ∗ indicates aggregate-expanded descriptors.

Feature Group	TS	Ours
Structural (17 descriptors)		
Graph topology (nodes, edges, transactions)		✓
Centrality* (degree, betweenness, closeness)	✓	✓
Centrality* (eigenvector, Katz)	✓	✓
Clustering coefficient*	✓	✓
Entity distribution (unique addresses, ratios)		✓
Monetary (18 descriptors)		
Value statistics (total, avg, std, median, quantiles)		✓
Multi-scale transforms* (normalized, log, harmonic)	✓	✓
Flow accumulation* (in/out value & count)	✓	✓
Long/short-term avg values*	✓	✓
Concentration metrics (Gini-proxy, volume/address)		✓
Temporal (12 descriptors)		
Lifetime statistics (span, inter-arrival times)		✓
Long/short-term dynamics* (in/out counts)	✓	✓
Accumulated frequency* (long/short-term)	✓	✓
Transfer frequency & recency*	✓	✓
Total descriptors	**30**	**47 (+17 novel)**

TS=TokenScout. Gray rows=novel descriptors beyond TokenScout. *=descriptor expanded through aggregate statistics (mean, std, min, max). The final ERC-20 model is trained on 17 retained features after collinearity pruning.

values reported in the literature for ERC-20 scam detection, while adding full end-to-end interpretability that prior deep-graph models do not provide. Notably, the result also improves operational precision: low FPR limits unnecessary escalation workload for exchange compliance teams and investigators.

Three takeaways emerge: (i) label quality beats volume—the 12× smaller ChainAbuse subset outperforms TokenScout by 2.4 F_1 points; (ii) the negative class matters—CoinMarketCap is a cleaner licit reference than TokenScout; (iii) naive source mixing can inject noise and reduce separability. These conclusions remain stable after pruning, so the provenance effect is not an artefact of redundant features.

4.3 Explainability Analysis

SHAP highlights two dominant signal families: transactional activity and centrality. Table 9 reports the top features with CIU scores.

Lower transaction counts push predictions toward scam, consistent with a limited on-chain footprint (Fig. 2). After collinearity pruning, `num_transactions` dominates ($|\bar{\phi}| = 0.21$, CI= 0.52), with compact-neighborhood features (`num_edges`, `unique_receivers`, `receiver_ratio`, `num_nodes`) reinforcing

Table 8: ERC-20 scam detection across four data source scenarios after collinearity-aware feature pruning. BA=Balanced Accuracy, FPR=False Positive Rate. Scenarios are ranked by (AUC-PR + F_2)/2 composite score.

Sc.	Scenario	F_1	BA	ROC	FPR
1	CA + CMC	**.985**	**.981**	**.992**	**1.0%**
2	TO + CMC	.961	.945	.982	1.8%
3	CA + TS	.957	.927	.971	3.2%
4	TO + TS	.918	.868	.938	4.9%

Table 9: Leading ERC-20 features with SHAP importance and CIU scores, computed via path-dependent TreeSHAP on the 17-feature collinearity-aware representation. CI and CU are computed over 200 stratified scam-class test instances.

Feature	$\lvert\bar{\phi}\rvert$	CI	CU
`num_transactions`	**0.210**	**0.520**	0.97
`node_eigenvector_centrality_mean`	0.063	0.052	0.95
`edge_transfer_value_normalized_mean`	0.038	0.034	0.95
`num_edges`	0.034	0.032	0.95
`unique_receivers`	0.028	0.038	0.95
`receiver_ratio`	0.023	0.012	0.94
`num_nodes`	0.022	0.035	0.95
`node_katz_centrality_mean`	0.021	0.026	0.94

the signal. Centrality features provide a complementary structural signal. In particular, `node_eigenvector_centrality_mean` and `node_katz_centrality_mean` indicate that risk reflects both activity volume and local-graph position. CU near 1.0 across top variables indicates consistent directional support.

Stability was confirmed against target correlation and DEXiRE antecedents. DEXiRE on the 17-feature surrogate, distilled through a compact MLP trained on the predictions of the production Balanced Random Forest, yields 18 propositional rules with 99% fidelity to the Random Forest and 95.9% balanced accuracy on covered samples, suggesting that part of the original rule complexity reflected collinear inputs rather than genuine decision boundaries. A representative rule, `IF total_transactions > 2.68 AND edge_transfer_recency_mean > 0.71 THEN scam`, captures an activity–recency interaction beyond univariate SHAP.

Operational Implications. For ERC-20 monitoring, the model can support a staged workflow: high-throughput RF scoring for early warning, followed by CIU-based inspection of borderline or high-risk cases. Since explanations are feature-based and reproducible, they can be integrated into compliance or supervisory reports without relying on opaque post-hoc narratives. The main warning profile combines limited history, compact graph structure, restricted receiver diversity, recency, transfer intensity, and centrality signals.

5 Discussion

Having tested our framework on two fundamentally different ledgers, we can now synthesize the results. Table 10 presents the key metrics, organized into four blocks.

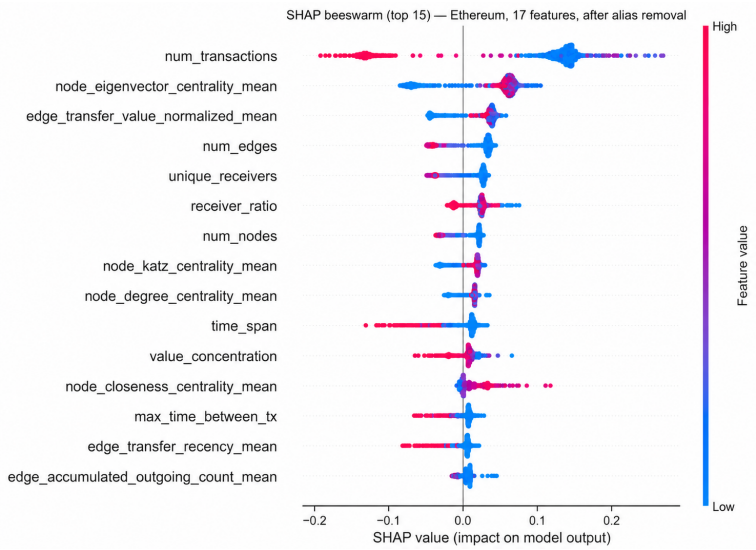

Fig. 2: SHAP beeswarm plot for ERC-20 token classification (17-feature collinearity-aware model), where each dot represents a token. The position reflects the SHAP impact, and the color indicates the feature value, with red representing high values and blue representing low values.

The first block identifies the best-performing data source combinations, namely CA (ChainAbuse) and CMC (CoinMarketCap), and shows that ERC-20 starts from a richer descriptor vocabulary (47 vs. 20), pruned to 17 retained features for the final model. The second block reports classification performance: both architectures achieve strong results, though ERC-20 benefits from cleaner class separation (1% vs. 7% false positive rate). The third block compares explainability outputs: Top SHAP Feature indicates the most globally important feature, Top CIU Feature shows which feature has highest contextual influence at the instance level, and DEXiRE Rules reports the number of symbolic rules (77 on Bitcoin, 44 of which fraud; 18 on ERC-20). Confident Learning, applied to the Bitcoin training set, improves macro-F_1 by 0.8pp.

Despite fundamentally different data structures, similar behavioral motifs emerge. Dispersion ratios dominate on Bitcoin, while transactional-activity signals (`num_transactions`) lead on ERC-20, complemented by eigenvector- and Katz-centrality features. Table 11 shows that these features capture a shared underlying concept, namely how funds spread across the network. We argue that this convergence reflects the economics of financial crime rather than blockchain architecture: money launderers *must* fragment value to obscure trails, scam operators *must* concentrate control to extract funds, and both operate under time pressure, producing temporal compression. These constraints are imposed by the crime, not the ledger and, hence the forensic vocabulary should now include terms borrowed from architectures such as DAG-based systems and sharded chains.

Table 10: Cross-architecture synthesis: best-scenario metrics, top explanatory features, and label noise analysis for both blockchain paradigms. Bitcoin BA, ROC-AUC and FPR are computed on the same held-out test partition used in Table 3 (Scenario 2, RF).

Metric	**Bitcoin**	**ERC-20**
Best Scenario	CA + Elliptic++	CA + CMC
Dataset Size	31,389 wallets	13,639 tokens
Feature Dimensions	20	17
Macro-F_1	0.92	0.985
Balanced Accuracy	92.5%	98.1%
ROC-AUC	0.95	0.992
False Positive Rate	7.0%	1.0%
Top SHAP Feature	unique_out_ratio	num_transactions
Top CIU Feature	total_unique_in	num_transactions
DEXiRE Rules	77 (44 fraud)	18

Table 11: Architecture-invariant fraud signatures: semantic equivalence of top discriminative features across Bitcoin (UTXO) and Ethereum (account-based) ledgers.

Semantic Category	**Bitcoin**	**ERC-20**
Dispersion	unique_out_ratio	unique_receivers
Centrality	clustering_coef	node_eigenvector_centrality
Temporal burst	time_interval_ratio	max_time_between_tx
Value fragment.	weighted_avg_tx	edge_transfer_value_normalized
Lifetime	wallet_lifetime	time_span

Transferability still requires implementation work. Bitcoin required address-level pruning and wallet aggregation for UTXO semantics, while Ethereum demanded log parsing and a higher-dimensional feature set (47 vs. 20). Nevertheless, semantic categories (structural, monetary, temporal) remain unchanged, suggesting that extending to a new ledger mainly requires a dedicated adapter rather than redesigning the framework. We emphasise that portability is claimed at the methodological level only: separate Bitcoin and Ethereum models are trained, with no parameter transfer evaluated across ledgers. A direct cross-chain test (training on one ledger, evaluating on the other) is identified as future work.

Finally, curation beats volume. The 12× smaller ChainAbuse corpus outperforms TokenScout by 2.4 percentage points in F_1. Confident Learning identifies 2.2% noisy labels in the Bitcoin dataset; removing them yields a 0.8 percentage point gain.

6 Conclusion

In this paper we address a regulatory trilemma: how to build fraud detectors that are accurate, interpretable, and portable across blockchain architectures. The framework we presented, combining Confident Learning for data curation, graph-temporal feature engineering, Balanced Random Forests, and a three-layer XAI stack, offers an answer at the methodological level: the pipeline template and the feature taxonomy are reusable across heterogeneous ledgers via chain-specific extractors, while the learned parameters remain ledger-specific.

Our empirical evaluation demonstrates strong performance: we achieve a macro-F_1 of 0.92 on 31,000 Bitcoin wallets and an F_1 of 0.985 with 98.1% Balanced Accuracy on 157,000 ERC-20 tokens.

An additional noteworthy outcome is the emergence of consistent discriminative features across different architectures. Dispersion, centrality, and temporal burstiness appear on both Bitcoin and Ethereum, despite radically different data models. We interpret this as an evidence that fraud signatures arise from the economics of crime, not from ledger design and this should broaden the scope of investigation to include these emerging architectures, such as DAG-based systems and sharded chains.

The framework supports a multi-tiered deployment pipeline: real-time RF scoring with SHAP for early warning, CIU for case-level investigation, and DEXiRE rules for audit-ready documentation. Our full implementation is openly accessible on an anonymous link [2].

We acknowledge several limitations. Our labels, though curated, ultimately derive from community reports, hence are noisy by nature. Adversarially-aware actors might adapt once they learn which heuristics we rely on. (i) Confident Learning handles i.i.d. label noise, not coordinated adversarial reporting; (ii) temporal features use complete entity histories, so metrics are valid ex-post and bound streaming use from above; (iii) splits are random stratified, leaving chronological re-validation as future work; (iv) the ERC-20 scope covers deployer-/contract-level scams, not intra-token manipulation; (v) collinearity pruning ($|\rho| < 0.7$) preserves headline performance, but residual dependence in a small centrality cluster suggests reading SHAP at the feature-family level; (vi) portability is methodological only, with no cross-chain parameter transfer evaluated.

References

1. Anjomshoae, S., Kampik, T., Främling, K.: Py-ciu: a python library for explaining machine learning predictions using contextual importance and utility. In: IJCAI-PRICAI 2020 Workshop on Explainable Artificial Intelligence (XAI), january 8, 2020 (2020)
2. Anonymous Authors: Code and Data for Cross-Architecture Blockchain Forensics. Anonymous Link: `https://figshare.com/s/0f4033826458a0686d7a` (2026), accessed: 2026-01-20
3. Bartoletti, M., Carta, S., Cimoli, T., Saia, R.: Dissecting ponzi schemes on ethereum: identification, analysis, and impact. Future Generation Computer Systems **100**, 915–953 (2020)
4. Chainalysis: The 2025 crypto crime report. Tech. rep., Chainalysis (2025), `https://go.chainalysis.com/2025-crypto-crime-report.html`
5. Chang, T.H., Svetinovic, D.: Improving bitcoin ownership identification using transaction patterns analysis. IEEE Transactions on Systems, Man, and Cybernetics: Systems **50**(1), 9–20 (2018)
6. Chen, W., Zheng, Z., Ngai, E.C.H., Zheng, P., Zhou, Y.: Exploiting blockchain data to detect smart ponzi schemes on ethereum. IEEE Access **7**, 37575–37586 (2019)

7. Contreras, V., Marini, N., Fanda, L., Manzo, G., Mualla, Y., Calbimonte, J.P., Schumacher, M., Calvaresi, D.: A dexire for extracting propositional rules from neural networks via binarization. Electronics **11**(24), 4171 (2022)
8. Du, H., Che, Z., Shen, M., Zhu, L., Hu, J.: Breaking the anonymity of ethereum mixing services using graph feature learning. IEEE Transactions on Information Forensics and Security **18**, 1–14 (2023)
9. Financial Action Task Force: International standards on combating money laundering and the financing of terrorism & proliferation: The FATF recommendations. Tech. rep., FATF/OECD (2012), updated 2023
10. He, B., Chen, Y., Chen, Z., Hu, X., Hu, Y., Wu, L., Chang, R., Wang, H., Zhou, Y.: Txphishscope: Towards detecting and understanding transaction-based phishing on ethereum. In: Proceedings of the 2023 ACM SIGSAC Conference on Computer and Communications Security. pp. 120–134 (2023)
11. Hong, Y., Kwon, H., Lee, J., Hur, J.: A practical de-mixing algorithm for bitcoin mixing services. In: Proceedings of the 2nd ACM Workshop on Blockchains, Cryptocurrencies, and Contracts. pp. 15–20 (2018)
12. Hu, Y., Seneviratne, S., Thilakarathna, K., Fukuda, K., Seneviratne, A.: Characterizing and detecting money laundering activities on the bitcoin network. arXiv preprint arXiv:1912.12060 (2019)
13. Hu, Y., Sun, Y., Chen, Y., Chen, Z., He, B., Wu, L., Zhou, Y., Chang, R.: Mfgscope: A lightweight framework for efficient graph-based analysis on blockchain. IEEE Transactions on Dependable and Secure Computing (2024)
14. Huo, Y., Hu, Y., Zhou, Y., Yu, T., Wu, L., Wang, C.: Shedding light on shadows: Automatically tracing illicit money flows on evm-compatible blockchains. Proceedings of the ACM on Measurement and Analysis of Computing Systems **9**(3), 1–35 (2025)
15. Iscan, C., Kumas, O., Akbulut, F.P., Akbulut, A.: Wallet-based transaction fraud prevention through lightgbm with the focus on minimizing false alarms. IEEE Access **11**, 131465–131474 (2023)
16. Krupp, J., Rossow, C.: teEther: Gnawing at ethereum to automatically exploit smart contracts. In: 27th USENIX security symposium (USENIX Security 18). pp. 1317–1333 (2018)
17. Li, X., Liu, S., Li, Z., Han, X., Shi, C., Hooi, B., Huang, H., Cheng, X.: Flowscope: Spotting money laundering based on graphs. In: Proceedings of the AAAI conference on artificial intelligence. vol. 34, pp. 4731–4738 (2020)
18. Li, X., Yepuri, A., Nikiforakis, N.: Double and nothing: Understanding and detecting cryptocurrency giveaway scams. In: Proceedings of the Network and Distributed System Security Symposium (NDSS) (2023)
19. Lin, D., Wu, J., Yu, Y., Fu, Q., Zheng, Z., Yang, C.: Denseflow: Spotting cryptocurrency money laundering in ethereum transaction graphs. In: Proceedings of the ACM on Web Conference 2024. pp. 4429–4438 (2024)
20. Liu, S., Hooi, B., Faloutsos, C.: Holoscope: Topology-and-spike aware fraud detection. In: Proceedings of the 2017 ACM on Conference on Information and Knowledge Management. pp. 1539–1548 (2017)
21. Lorenz, J., Silva, M.I., Aparício, D., Ascensão, J.T., Bizarro, P.: Machine learning methods to detect money laundering in the bitcoin blockchain in the presence of label scarcity. In: Proceedings of the first ACM international conference on AI in finance. pp. 1–8 (2020)
22. Lundberg, S.M., Lee, S.I.: A unified approach to interpreting model predictions. Advances in neural information processing systems **30** (2017)

23. Meiklejohn, S., Pomarole, M., Jordan, G., Levchenko, K., McCoy, D., Voelker, G.M., Savage, S.: A fistful of bitcoins: characterizing payments among men with no names. In: Proceedings of the 2013 conference on Internet measurement conference. pp. 127–140 (2013)
24. Monamo, P., Marivate, V., Twala, B.: Unsupervised learning for robust bitcoin fraud detection. In: 2016 Information Security for South Africa (ISSA). pp. 129–134. IEEE (2016)
25. Monamo, P.M., Marivate, V., Twala, B.: A multifaceted approach to bitcoin fraud detection: Global and local outliers. In: 2016 15th IEEE International Conference on Machine Learning and Applications (ICMLA). pp. 188–194. IEEE (2016)
26. Möser, M., Böhme, R., Breuker, D.: Towards risk scoring of bitcoin transactions. In: Financial Cryptography and Data Security: FC 2014 Workshops, BITCOIN and WAHC 2014. pp. 16–32. Springer (2014)
27. Nerurkar, P., Bhirud, S., Patel, D., Ludinard, R., Busnel, Y., Kumari, S.: Supervised learning model for identifying illegal activities in bitcoin. Applied Intelligence **51**, 3824–3843 (2021)
28. Northcutt, C., Jiang, L., Chuang, I.: Confident learning: Estimating uncertainty in dataset labels. Journal of Artificial Intelligence Research **70**, 1373–1411 (2021)
29. Phetsouvanh, S., Oggier, F., Datta, A.: Egret: Extortion graph exploration techniques in the bitcoin network. In: 2018 IEEE International conference on data mining workshops (ICDMW). pp. 244–251. IEEE (2018)
30. Tang, Y., Xu, C., Zhang, C., Wu, Y., Zhu, L.: Analysis of address linkability in tornado cash on ethereum. In: China Cyber Security Annual Conference. pp. 39–50. Springer (2021)
31. Torres, C.F., Steichen, M., et al.: The art of the scam: Demystifying honeypots in ethereum smart contracts. In: 28th USENIX Security Symposium (USENIX Security 19). pp. 1591–1607 (2019)
32. Tovanich, N., Cazabet, R.: Pattern analysis of money flows in the bitcoin blockchain. In: International Conference on Complex Networks and Their Applications. pp. 443–455. Springer (2022)
33. Toyoda, K., Ohtsuki, T., Mathiopoulos, P.T.: Multi-class bitcoin-enabled service identification based on transaction history summarization. In: 2018 IEEE Int. Conf. Internet of Things (iThings), GreenCom, CPSCom & SmartData. pp. 1153–1160. IEEE (2018)
34. United Nations: Transforming our world: The 2030 agenda for sustainable development (2015), `https://sdgs.un.org/goals/goal16`, a/RES/70/1. Target 16.4: By 2030, significantly reduce illicit financial and arms flows, strengthen the recovery and return of stolen assets and combat all forms of organized crime
35. Victor, F., Weintraud, A.M.: Detecting and quantifying wash trading on decentralized cryptocurrency exchanges. In: Proceedings of the Web Conference 2021. pp. 23–32 (2021)
36. Weber, M., Domeniconi, G., Chen, J., Weidele, D.K.I., Bellei, C., Robinson, T., Leiserson, C.E.: Anti-money laundering in bitcoin: Experimenting with graph convolutional networks for financial forensics. arXiv preprint arXiv:1908.02591 (2019)
37. Wu, C., Chen, J., Zhao, Z., He, K., Xu, G., Wu, Y., Wang, H., Li, H., Liu, Y., Xiang, Y.: Tokenscout: Early detection of ethereum scam tokens via temporal graph learning. In: Proceedings of the 2024 on ACM SIGSAC Conference on Computer and Communications Security. pp. 956–970 (2024)
38. Wu, J., Lin, D., Fu, Q., Yang, S., Chen, T., Zheng, Z., Song, B.: Towards understanding asset flows in crypto money laundering through the lenses of ethereum heists. IEEE Transactions on Information Forensics and Security (2023)

39. Wu, M., McTighe, W., Wang, K., Seres, I.A., Bax, N., Puebla, M., Mendez, M., Carrone, F., De Mattey, T., Demaestri, H.O., et al.: Tutela: An open-source tool for assessing user-privacy on ethereum and tornado cash. arXiv preprint arXiv:2201.06811 (2022)
40. Xia, P., Wang, H., Gao, B., Su, W., Yu, Z., Luo, X., Zhang, C., Xiao, X., Xu, G.: Trade or trick? detecting and characterizing scam tokens on uniswap decentralized exchange. Proceedings of the ACM on Measurement and Analysis of Computing Systems **5**(3), 1–26 (2021)
41. Zhao, C., Guan, Y.: A graph-based investigation of bitcoin transactions. In: Advances in Digital Forensics XI: 11th IFIP WG 11.9 International Conference. pp. 79–95. Springer (2015)
42. Zhou, L., Xiong, X., Ernstberger, J., Chaliasos, S., Wang, Z., Wang, Y., Qin, K., Wattenhofer, R., Song, D., Gervais, A.: Sok: Decentralized finance (defi) attacks. In: 2023 IEEE Symposium on Security and Privacy (SP). pp. 2444–2461. IEEE (2023)