International Journal of Data Science and Analysis
2026; 12(5): 97-113
http://www.sciencepublishinggroup.com/j/ijdsa
doi: 10.11648/j.ijdsa.20261205.11

Coverage and Protocol-Aware Evaluation of Risk Enrichment and Graph-Tabular Learning for Anti-Money Laundering and Fraud Detection

Karim Bettaieb[1, *], Afef Kacem Echi[1, 2], Houcem Hammami[1]

[1]Department of Applied Mathematics and Modeling Engineering, University of Tunis, Tunis, Tunisia
[2]Laboratory for Information and Communication Technologies and Electrical Engineering, University of Tunis, Tunis, Tunisia

Email address:

bettaiebkariim@gmail.com (Karim Bettaieb), ff.kacem@gmail.com (Afef Kacem Echi), houcem.hammami@ept.ucar.tn (Houcem Hammami)

To cite this article:

Karim Bettaieb, Afef Kacem Echi, Houcem Hammami. (2026). Coverage and Protocol-Aware Evaluation of Risk Enrichment and Graph-Tabular Learning for Anti-Money Laundering and Fraud Detection. *International Journal of Data Science and Analysis, 12*(5), 97-113. https://doi.org/10.11648/j.ijdsa.20261205.11

Received: 13 July 2026; **Accepted:** 29 July 2026 **Published:** 8 September 2026

Abstract: Reported gains from graph-based machine learning in anti-money laundering and fraud detection can vary substantially with the historical evidence available to a model and with the evaluation protocol used. This study introduces a coverage-aware and protocol-aware evaluation framework for assessing account-risk enrichment, graph neural networks, tabular gradient boosting, and graph-tabular stacking across synthetic banking anti-money-laundering data, a real Bitcoin transaction graph, and a complementary real credit-card fraud dataset. The analysis compares account-risk enrichment, four topology-only graph neural networks, XGBoost, LightGBM, and embedding-based stacking under warm-start, account-grouped, cold-start, coverage-sensitivity, and temporal evaluation settings. On the synthetic HI-Small benchmark, strict account grouping removes source-account historical coverage by construction and substantially limits the opportunity for enrichment to contribute useful signal. Under the warm-start protocol, enrichment produces only a small change in discrimination, while the coverage-sensitivity analysis shows no reliable improvement in ROC-AUC as historical coverage increases. PR-AUC results instead indicate a small but consistent performance cost at most tested coverage levels after correction for multiple comparisons. On the lower-coverage LI-Small benchmark, enrichment again shows a small negative effect that does not remain significant after manuscript-wide correction. External validation on the Elliptic Bitcoin graph shows that topology-only graph neural networks do not automatically outperform strong tabular models. GraphSAGE is the strongest graph neural network tested, but XGBoost and LightGBM achieve clearly higher ROC-AUC and PR-AUC. Graph-tabular stacking is also dataset dependent: it yields small positive gains on Elliptic, while tabular features alone are competitive with or superior to stacking on HI-Small. These results show that graph-based improvements should not be interpreted independently of historical coverage, split construction, and feature redundancy. Practical evaluation should report coverage and protocol alongside graph-derived claims, use strong tabular baselines, and complement ROC-AUC with PR-AUC and calibration-oriented metrics before concluding that a graph-based method provides a robust advantage.

Keywords: Anti-money Laundering, Graph Neural Networks, Protocol-aware Evaluation, Coverage-aware Evaluation, Account-risk Enrichment, Multi-dataset Validation, Stacking

1. Introduction

Reported gains in AI-based financial crime detection are difficult to trust in isolation: the same technique can appear beneficial, neutral, or harmful depending on the historical evidence available to it and the evaluation protocol under which it is measured. Financial institutions operate under AML regulatory frameworks (FATF recommendations, the US Bank Secrecy Act/FinCEN, EU AML Directives) that mandate detection and reporting of suspicious activity. The core

detection challenge is structural: money laundering exploits the relational nature of financial networks through typologies spanning multiple accounts – *fan-out*, *fan-in*, *hub*, and *cycle* patterns – that are not detectable from individual transactions in isolation, which motivates graph-relational learning over transaction-level models alone.

Graph neural networks (GNNs) have produced a growing literature applying graph convolution, attention, and message-passing to fraud and AML detection [1–4, 7, 8, 11]. However, graph topology is not automatically superior to tabular representations: gradient boosting (XGBoost, LightGBM [5, 6]) remains highly competitive on transaction-level features, and a recent critical re-evaluation of GNNs for Bitcoin fraud detection found that, once transductive leakage is removed under a genuinely inductive protocol, real topology can underperform edge-shuffled random topology, with a Random Forest outperforming every GNN operator tested [20] – a finding on a structurally adjacent real dataset that this paper investigates independently rather than assumes. Whether a GNN, an enrichment technique, or a stacked combination improves detection depends on factors recent methodological critiques of this domain identify as recurring and unresolved [21, 22]: the historical *coverage* of any account-level statistic used as a node feature, whether the evaluation protocol allows entities to leak between training and validation (warm-start) or strictly isolates them (cold-start), the graph construction used, and whether apparent gains reflect genuine signal or evaluation artifacts.

The account-risk and cold-start-fraud-detection literatures separately establish that historical-evidence sufficiency affects graph-based fraud detection [16, 17] – including in the domain-adjacent setting of cold-start fraud detection on online review platforms [24], which we cite for its general cold-start mechanism, not as AML or financial-transaction evidence – while the leakage and evaluation-protocol literature separately establishes that split construction affects measured performance [15, 21, 22]; what is not yet established, for account-risk enrichment or graph–tabular stacking specifically, is a shared vocabulary connecting how much a technique's benefit should be trusted to the coverage and protocol conditions – jointly, not in isolation – that produced it. A finding that looks like a stable improvement under one protocol can collapse under a stricter one, and a finding on one dataset family may not transfer to another.

This paper does not propose a new GNN architecture. Instead, it studies *when* graph learning, account-risk enrichment, and embedding-based stacking help or fail, across a coverage-aware and protocol-aware design spanning three data sources: (i) the synthetic IBM AMLSim benchmark (HI-Small, LI-Small) [10], isolating the effect of protocol and coverage on a graph-node account-risk enrichment mechanism; (ii) Elliptic [9], a real Bitcoin transaction graph with genuine illicit/licit labels, used as external validation of whether conclusions on synthetic banking data transfer to real crypto-AML data – related to, but distinct from, banking AML, since Elliptic's entities are transactions, not persistent accounts; and (iii) Credit Card Fraud (Université Libre de Bruxelles, ULB) [18], a real, extreme-imbalance card-fraud benchmark used only as a complementary non-AML robustness check on the evaluation framework, not as AML evidence.

This paper makes five contributions. *(C1)* A protocol-aware and coverage-aware evaluation framework – coverage, fallback rate, effective non-zero coverage, and protocol sensitivity – that makes explicit how much historical evidence supports a reported enrichment or graph-derived value, and under which split construction it was measured (Section 3.2); this is the organizing contribution the remaining four apply. *(C2)* A systematic analysis of account-risk enrichment under warm-start, cold-start, and account-grouped protocols on synthetic banking AML data, including a five-level coverage sensitivity sweep isolating coverage from protocol. *(C3)* External validation on a real crypto-AML graph (Elliptic): a topology-only GNN sweep (GCN, GraphSAGE, GAT, GIN) showing graph topology alone is not automatically superior to tabular gradient boosting. *(C4)* A cross-dataset stacking analysis examining a candidate boundary condition for graph–tabular complementarity: consistently positive but small gains on Elliptic (whose 166 features contain no graph-derived quantity), versus no measured advantage on HI-Small (whose tabular features already include $target - risk$, $source - risk$, fan-in, and fan-out). This contrast is consistent with, but does not causally establish, the hypothesis that graph embeddings are more useful when they contribute information not already encoded by the tabular features. *(C5)* A complementary real, non-AML card-fraud stress test of the operational-metrics and calibration components under extreme imbalance. Synthesized, C2–C5 yield evidence-based guidance on why small near-zero ΔAUC values should be interpreted cautiously, and why ROC-AUC should be complemented with PR-AUC, Precision/Recall@k, Brier score, and Expected Calibration Error.

2. Related Work

Graph learning for fraud and AML. The Graph Convolutional Network (GCN) [1], GraphSAGE [2], the Graph Attention Network (GAT) [3], and the Graph Isomorphism Network (GIN) [4] established the message-passing architectures used throughout this study. Fraud-specific variants include PC-GNN [7] and CARE-GNN [8]; for AML specifically, Johannessen and Jullum [11] applied heterogeneous GNNs to bank transaction networks, and Akoglu et al. [12] survey graph-based anomaly detection broadly. This paper introduces no new GNN operator; it isolates the conditions – coverage, protocol, dataset family – under which these existing architectures and account-risk enrichment succeed or fail.

Tabular gradient boosting. XGBoost [5] and LightGBM [6] remain highly competitive on tabular financial features. Van Vlasselaer et al. [13] showed that augmenting tabular features with network-derived features improves credit card fraud detection (APATE), directly informing this paper's

stacking design; Pourhabibi et al. [14] survey graph-based fraud detection methods. Tabular gradient boosting is this study's consistent reference point, not an assumed weaker baseline.

Account-risk and target encoding. Micci-Barreca [16] introduced smoothed target encoding for high-cardinality categorical attributes; Prokhorenkova et al. [17] formalized ordered target statistics in CatBoost to prevent training-set leakage. The $target - risk/source - risk$ features used here (Section 3.2) are structurally equivalent to per-account target encoding applied as graph-node enrichment.

Evaluation protocols and leakage. Kaufman et al. [15] formally defined data leakage; we adopt their discipline of treating near-perfect results (AUC/F1 $\approx$ 1.000) as contamination diagnostics, never as reportable findings. Strict account-grouped cross-validation (GroupKFold/GroupShuffleSplit) is the appropriate stress test for deployment settings with new accounts; a stratified split without grouping is more optimistic (warm-start); a temporal split, as used for Elliptic following its original authors [9], evaluates strictly later time steps. None is universally correct, and results under different protocols are never combined into one leaderboard.

Operational metrics and calibration. Under the severe class imbalance typical of fraud/AML data, PR-AUC and Precision/Recall@k are more informative than ROC-AUC alone [13, 14], and calibration (Brier score, Expected Calibration Error [19]) addresses the separate question of whether predicted probabilities are individually trustworthy. Section 7 reports these metrics across the available experiments, with explicit exceptions documented in Section 3.3.

Synthesis and gap. Each of the four threads above – graph architectures, tabular gradient boosting, account-risk/target encoding, and evaluation-protocol/leakage discipline – has largely been developed and validated on its own terms, with operational-metrics work (the fifth thread) typically layered on afterward rather than integrated into the design of the comparison itself. Graph-architecture papers [1–4, 7, 8, 11] report accuracy gains without systematically varying the historical coverage available to the graph signal; the target-encoding and account-risk literature [16, 17] establishes leakage-safe estimation but not how the resulting statistic's reliability depends on split construction; the leakage/evaluation-protocol literature [15, 21, 22] establishes that split construction changes measured performance in general, without connecting this specifically to account-risk enrichment or graph–tabular stacking; and the cold-start literature [24] establishes that historical-evidence sparsity degrades detection without a shared vocabulary for quantifying how much evidence a given account-level statistic actually had. Existing studies generally evaluate graph architectures, account-risk enrichment, leakage controls, or graph–tabular combinations separately. Few studies jointly examine how historical account coverage, split protocol, dataset family, and redundancy between graph-derived and tabular information affect the measured benefit of graph learning. This study addresses this gap through a protocol-aware and coverage-aware evaluation across synthetic banking AML, real crypto-AML, and complementary card-fraud data (Section 3.2).

3. Datasets, Coverage-Aware Framework, and Protocols

3.1. Datasets

Table 1 summarizes the four datasets and the protocols applicable to each. They are not interchangeable – each belongs to a distinct tier (synthetic banking AML, real crypto AML, real non-AML fraud) – and results are never combined across tiers into one leaderboard.

HI-Small/LI-Small are two scale/prevalence variants of the synthetic IBM AMLSim multi-bank wire-transfer benchmark [10], sharing the same schema and laundering-typology labels (fan-out, fan-in, hub, cycle). HI-Small_Trans.csv contains 5,078,345 raw transactions; LI-Small_Trans.csv (lower illicit rate) contains 6,924,049. Under the same 10:1 negative-subsampling protocol used throughout (approximating 1–10% operational fraud prevalence and keeping experiments tractable), the HI-Small working set used for the heavy graph protocols is 82,947 transactions (5,177 positive, 6.24%), and the LI-Small working set is 74,215 (3,565 positive, 4.80%). Each transaction carries a source account, destination account, amount, currency, payment format, and binary label. HI-Small is the primary protocol/coverage-sensitivity dataset; LI-Small, with substantially lower historical coverage, is used as a cold-start diagnostic. Both are synthetic throughout and never used to support claims about real banking data.

Elliptic [9] is a real, anonymized Bitcoin transaction graph: 203,769 nodes (transactions), 234,355 directed edges (verified BTC flow), 166 features per node (a documented time step in $\{1, \ldots, 49\}$, $\approx$2-week intervals, plus 165 undocumented features). Of all nodes, 4,545 (2.23%) are labeled illicit, 42,019 (20.62%) licit, and 157,205 (77.15%) unlabeled; restricted to labeled nodes, the positive rate is 9.76%. Obtained via its official Kaggle release under CC BY-NC-ND 4.0. *Elliptic is real crypto illicit-transaction/AML graph validation, not banking AML*: its nodes are individual transactions, not persistent accounts, and it has no source-/destination-account identity analogous to HI-Small/LI-Small – this directly determines which parts of Section 3.2 apply to it. Unlabeled nodes are retained as message-passing context but excluded from the training loss and from every reported metric.

Credit Card Fraud (ULB) [18] contains 284,807 real, anonymized card transactions, 492 confirmed fraud (0.173%) – more extreme imbalance than either AML dataset – with 28 PCA components (V1–V28), amount, and elapsed time; no graph structure. We use it strictly as a *complementary, non-AML, rare-event robustness check* on the operational-metrics framework (Section 7), not as AML evidence.

Table 1. *Dataset taxonomy and applicable evaluation protocols (Sections 3.1 and 3.3).*

Dataset	Nature	Domain	Working-set size (positive %)	Applicable protocols
HI-Small	Synthetic	Banking AML	82,947 (6.24%)	P1, P2-warmstart, P2-acctgrp, coverage sweep
LI-Small	Synthetic	Banking AML	74,215 (4.80%)	P3 (cold-start diagnostic)
Elliptic	Real	Crypto AML	203,769 nodes (9.76% of labeled)	Elliptic temporal
Credit Card Fraud	Real	Card fraud, not AML	284,807 (0.173%)	Operational-metrics check only

3.2. Coverage-Aware Evaluation Framework

Given transactions $D = \{(t_i, y_i)\}$ with source account a_i^s and destination account a_i^d, and a training split $D_{\text{tr}} \subset D$, we define two account-risk enrichment statistics, estimated from D_{tr} labels only:

$$target - risk[b] = \Pr[y = 1 \mid a_i^d = b], \quad b \in A^d, \qquad (1)$$

$$source - risk[a] = \Pr[y = 1 \mid a_i^s = a], \quad a \in A^s. \qquad (2)$$

Both are structurally equivalent to per-account target encoding (Section 2); accounts absent from training receive an explicit fallback of 0.0, making the cold-start failure mode visible in the feature itself. This mechanism applies to *HI-Small/LI-Small only* (as explained in the framework-boundary paragraph below, this mechanism does not transfer to Elliptic).

We distinguish three quantities reported alongside every enrichment result: *coverage* (the fraction of evaluation entities whose statistic was estimated from ≥ 1 training example); *fallback rate*, coverage's complement; and *effective non-zero coverage*, a stricter quantity – the fraction of entities whose value is both training-derived *and* non-zero, since a covered account can still map to $target-risk = 0$/$source-risk = 0$ if it had zero observed fraud in training. We report this distinction because account-level coverage figures (e.g., $\approx 45\%$ under warm-start, Section 4) are substantially higher than transaction-level effective non-zero coverage (2–12% under the same protocol); conflating the two overstates the genuine signal available.

Warm-start describes a split where a substantial fraction of validation accounts also appear in training, so enrichment carries meaningful information; *cold-start* is the complement – near-universal fallback. Whether a protocol produces warm- or cold-start is not intrinsic to the dataset but a mechanical consequence of split construction, which motivates *protocol sensitivity*: the measured effect of a fixed technique changing, potentially reversing sign, purely as a function of the evaluation protocol applied to the same data and model. Under GroupKFold/GroupShuffleSplit *by source account*, every source account in a validation fold is, by construction, absent from that fold's training data; consequently $source - risk$ coverage in validation is exactly 0.0% in every fold – a deterministic, mechanical consequence of grouping by the same key $source - risk$ is indexed by, not an empirical finding. $target - risk$ coverage survives partially ($\approx$ 12.6–14.0% under P1) only because destination accounts are not the grouping key. Reporting an AUC without its protocol is, under this framework, an incomplete claim.

Following Kaufman et al. [15], we treat AUC/F1 at or near 1.000 as a contamination diagnostic, never a reportable result; no such degenerate configuration is reported anywhere in this paper.

Boundary: non-transferability to Elliptic. The definitions above presuppose a persistent account entity that can be the source/destination of more than one transaction. This holds for HI-Small/LI-Small but not Elliptic, whose nodes are individual transactions with no persistent wallet/actor identifier. *$target - risk$ and $source - risk$ are therefore not computed for Elliptic anywhere in this paper*; we treat this as an explicit boundary rather than an approximate substitute that could silently introduce leakage of its own. A leakage-safe, inductive account-risk analogue for transaction-graph datasets (e.g., using only strictly earlier time steps) is future work (Section 9); Section 5 evaluates Elliptic using node topology and features only.

3.3. Models, Metrics, and Protocols

Models. Logistic Regression, XGBoost [5], and LightGBM [6] are the tabular reference point throughout. Four standard topology-only GNNs – GCN [1], GraphSAGE [2], GAT [3], GIN [4] – are used as-proposed (2-layer, hidden width 64→32, weighted cross-entropy) on both HI-Small and Elliptic; no architectural novelty is claimed. On HI-Small only, *HybridGAT* is a GAT configured to accept $target - risk$/$source - risk$ alongside nine structural/behavioral node features; we report a *base* variant (without $target - risk$/$source - risk$) and an *enriched* variant to isolate enrichment's own contribution. HybridGAT is not evaluated on Elliptic, where its enrichment features are undefined (Section 3.2); Elliptic's GAT baseline is unenriched only. For *stacking* (contribution C4), XGBoost is trained on tabular features concatenated with a graph-derived embedding – the 64-d penultimate HybridGAT representation on HI-Small, the 32-d penultimate GraphSAGE representation on Elliptic – against matched tabular-only and embeddings-only baselines, on both datasets, across three seeds.

Metrics. Where saved probability outputs were available, a single shared module computed the Receiver Operating Characteristic Area Under the Curve (ROC-AUC), the Precision–Recall Area Under the Curve (PR-AUC) – both referred to generically as AUC where the specific curve is clear from context – F1, precision/recall, Precision@k/Recall@k for $k \in \{1\%, 5\%, 10\%\}$, Brier score, and Expected Calibration Error (ECE) using consistent definitions across every dataset in this paper. Exceptions: no calibration metric was computed for the HI-Small stacking ablation (Section 7); the Elliptic

tabular baselines and the Credit Card Fraud comparison are single-split runs with no saved per-instance predictions, so no bootstrap confidence interval is available for them; and the HI-Small stacking ablation's per-seed differences likewise have no computed confidence interval (Section 6). The F1 threshold is selected, per fold/seed/protocol, by scanning the precision-recall curve for the value maximizing F1 on the same evaluation split being reported – an in-sample-optimal threshold, not fixed in advance or selected on an independent validation split, confirmed by inspection of the evaluation code. F1 is accordingly an optimistic diagnostic operating-point measure and is not used as the primary basis for model comparison in this paper; ROC-AUC and PR-AUC, being threshold-independent, are treated as the primary ranking metrics.

Protocols. Table 2 summarizes the six protocols used; they are not interchangeable and are never combined into one leaderboard. P1 and P2-acctgrp group by source account, structurally eliminating validation $source - risk$ coverage; P2-warmstart does not group, permitting warm-start coverage; the coverage sweep holds P2-warmstart fixed and varies what fraction of naturally coverable transactions retain their non-fallback value, isolating coverage from split construction. *Seeds, precisely stated:* P2-warmstart, P2-acctgrp, the coverage sweep, the Elliptic GNN sweep, and both stacking ablations use three seeds (42, 123, 2024); *P1 (5-fold GroupKFold) and P3 (LI-Small) are single-seed (*42*)*, a computational trade-off discussed in Section 9, not an inconsistency across protocols. The Elliptic temporal split reuses a fixed train/test assignment, and a leakage guard confirms on every run that zero edges connect a train-side to a test-side node. All GNN training used a CPU-only PyTorch Geometric installation (Section 9).

Table 2. *Evaluation protocols. HI-Small/LI-Small protocols follow the account-grouping conventions of Section 3.2; the Elliptic protocol follows the temporal convention of [9].*

Protocol	Dataset	Split
P1	HI-Small	5-fold GroupKFold by source account (seed 42 only)
P2-warmstart	HI-Small	Stratified label split, no account grouping (3 seeds)
P2-acctgrp	HI-Small	GroupShuffleSplit by source account (3 seeds)
Coverage sweep	HI-Small	P2-warmstart, retained coverage varied 0–100% (3 seeds)
P3	LI-Small	GroupShuffleSplit by source account, cold-start diagnostic (seed 42 only)
Elliptic temporal	Elliptic	Train: time step ≤ 34; test: > 34 (3 seeds)

3.4. Implementation and Reproducibility Details

This subsection consolidates, in a single reference point, implementation details reported elsewhere in this paper; no value here is new relative to the sections it is drawn from. All GNN training used a CPU-only PyTorch Geometric installation (Section 9); tabular baselines used XGBoost and LightGBM [5, 6] and Logistic Regression. The four topology-only GNN architectures (GCN, GraphSAGE, GAT, GIN) were used as-proposed: 2-layer, hidden width $64 \rightarrow 32$, weighted cross-entropy loss, trained for 50 epochs on the Elliptic sweep (Section 3.3); HybridGAT (HI-Small only) extends this GAT configuration with nine structural/behavioral node features plus, in its enriched variant, $target - risk$/$source - risk$. Stacking used, as the graph-derived input, the 64-d penultimate HybridGAT representation on HI-Small and the 32-d penultimate GraphSAGE representation on Elliptic (Section 3.3). Graphs were constructed from shared source/destination accounts for HI-Small/LI-Small and used as-is from Elliptic's official edge list (Section 9); Elliptic's 157,205 unlabeled nodes are retained as message-passing context but excluded from the training loss and from every reported metric (Section 3.1). P2-warmstart, P2-acctgrp, the coverage sweep, the Elliptic GNN sweep, and both stacking ablations used three seeds (42, 123, 2024); P1 and P3 are single-seed (42) (Section 3.3). Splits were constructed with GroupKFold/GroupShuffleSplit by source account for P1/P2-acctgrp/P3, a stratified label split without grouping for P2-warmstart, and a fixed temporal train (time step $\leq$ 34)/test ($>$ 34) assignment for Elliptic; a leakage guard confirms on every Elliptic run that zero of 234,355 edges connect a train-side to a test-side node (Section 5), and account-grouping by construction prevents source-account leakage on HI-Small/LI-Small (Section 3.2). The F1 operating threshold is selected, per fold/seed/protocol, by scanning the precision–recall curve for the value maximizing F1 on the same evaluation split being reported – an in-sample-optimal threshold, not a fixed or independently validated one (Section 3.3); ROC-AUC and PR-AUC, being threshold-independent, are the primary ranking metrics. Confidence intervals use a two-level (seed+instance) bootstrap with 10,000 resamples where per-instance predictions were saved (Section 4.3); hypothesis tests include Wilcoxon signed-rank, McNemar (at a 0.5 threshold), and a Friedman test with Nemenyi post-hoc for the four-architecture Elliptic comparison (Section 5.1). All 23 formal hypothesis tests computed in this paper were corrected jointly using the Benjamini–Hochberg procedure at FDR$=0.05$, implemented with a reproducible script and independently cross-checked against the `statsmodels` reference implementation [25] (Section 6.3), at a False Discovery Rate (FDR) of 0.05. *Not independently verifiable from this paper's saved experiment logs, and therefore not reported here rather than approximated:* exact pinned software version numbers

(Python, PyTorch, PyTorch Geometric, XGBoost, LightGBM, `statsmodels`) and an explicit early-stopping configuration; an early-stopping interaction is discussed only as a plausible, untested mechanism for the stacking probability-compression pattern in Section 8, not as a logged hyperparameter.

4. Results on Synthetic Banking AML Data

Table 3 summarizes the headline HI-Small and LI-Small result for every protocol and model; results are grouped by protocol and must be read protocol-by-protocol, not as a single ranked list. Every interpretive claim below is paired with the statistical evidence supporting it – a bootstrap CI, a signed-rank test, a multi-seed mean/std, or an explicit statement that no test is available.

Table 3. *Graph experiment results, HI-Small/LI-Small (synthetic banking AML), CPU-only PyTorch Geometric. Grouped by protocol; not a single ranked list. HI-Small stacking rows report the three-seed mean ± std for configuration C (tabular + embedding); the full three-configuration breakdown is Table 5.*

Protocol	Model	AUC	AUC std	F1	F1 std
P1: 5-fold GroupKFold (seed 42)	HybridGAT+enrichment	0.6871	0.0536	0.2052	0.0599
P2-warmstart (3 seeds)	HybridGAT base	0.7039	0.0074	0.2210	0.0037
P2-warmstart (3 seeds)	HybridGAT enriched	0.7075	0.0006	0.2194	0.0061
P2-warmstart (3 seeds)	XGBoost + GNN Stacking (config. C)	0.8093	0.0386	0.5417	0.0295
P2-acctgrp (3 seeds)	HybridGAT base	0.6821	0.0350	0.2025	0.0346
P2-acctgrp (3 seeds)	HybridGAT enriched	0.6823	0.0350	0.2018	0.0346
P2-acctgrp (3 seeds)	XGBoost + GNN Stacking (config. C)	0.6927	0.0599	0.3494	0.0378
LI-Small, P3 (seed 42)	XGBoost tabular	0.7386	–	0.1834	–
LI-Small, P3 (seed 42)	HybridGAT base	0.6417	–	0.1351	–
LI-Small, P3 (seed 42)	HybridGAT enriched	0.6404	–	0.1341	–
LI-Small, P3 (seed 42)	XGBoost GNN Stacking	0.7447	–	0.2078	–

4.1. P1: A Strict Cold-Start Stress Test

Under P1 (5-fold GroupKFold by source account, seed 42), HybridGAT with enrichment reaches AUC$= 0.6871 \pm 0.0536$, F1$= 0.2052 \pm 0.0599$ – variance across the 5 folds of one run, not across seeds (Section 3.3). As established structurally in Section 3.2, $source - risk$ coverage in validation is exactly 0.0% in every fold under this protocol – a deterministic consequence of grouping by the same key $source - risk$ is indexed by – while $target - risk$ coverage reaches only $\approx$ 12.6–14.0%. This structural result is sufficient on its own to establish P1 as the strictest cold-start stress test in this study, under which enrichment has, by construction, the least opportunity of any protocol here to contribute signal. *No unenriched HybridGAT model was evaluated under P1's exact 5-fold protocol*, so we make no claim that the enriched model is statistically indistinguishable from an unenriched baseline: no comparison number or test exists for that specific claim, and we do not assert one.

4.2. P2: Warm-Start and Account-Grouped Results

Under P2-warmstart (stratified label split, no account grouping), HybridGAT base reaches AUC$= 0.7039 \pm 0.0074$ and enriched AUC$= 0.7075 \pm 0.0006$ across three seeds, ΔAUC $= +0.0036$ (Wilcoxon signed-rank, $W = 2.0$, $p = 0.75$); under P2-acctgrp (GroupShuffleSplit by source account, 80/20), base AUC$= 0.6821 \pm 0.0350$ vs. enriched AUC$= 0.6823 \pm 0.0350$ ($W = 0.0$, $p = 0.50$). Neither is significant, but with only three paired seeds the smallest possible two-sided Wilcoxon p-value is 0.25: this test has essentially no power at this sample size, so non-significance here is a failure to reject at very low power, not proof of no effect. The more informative evidence for P2-warmstart is the coverage-sensitivity sweep below, which reuses this protocol's full per-instance predictions for a substantially more powerful bootstrap test. Together, these results support the reading that account grouping specifically, not the warm-start/cold-start label alone, determines whether enrichment coverage exists at all, while any residual warm-start benefit is small and not distinguishable from zero on the available test.

4.3. Coverage Sensitivity

Table 4 and Figure 1 report a controlled sweep holding the P2-warmstart split fixed while varying, from 0% to 100%, the fraction of naturally coverable validation transactions that retain their non-fallback value, with a two-level (seed + instance) bootstrap (10,000 resamples) applied to the full per-instance predictions at every seed and level.

Table 4. *Coverage sensitivity sweep on HI-Small (P2-warmstart held fixed). dest%/src%/fallback% are the shares of validation transactions with non-zero* `dest_risk`*, non-zero* `src_risk`*, and neither. Δ columns are two-level (seed+instance) bootstrap results,* 10,000 *resamples, two-sided raw p;* p_{BH} *is the Benjamini–Hochberg-adjusted p-value from this paper's complete, manuscript-wide 23-test family (Section 6.3); * marks* $p_{\mathrm{BH}} < 0.05$*. All five ΔAUC tests remain non-significant after correction (*$p_{\mathrm{BH}} = 0.986$ *for every level).*

Coverage	Base AUC	Enrich AUC	ΔAUC [95% CI] (raw p)	ΔPR-AUC [95% CI]	raw p / p_{BH}	dest%/src%/fallback%
0%	0.7039	0.7067	$+0.0027$ $[-0.0080, 0.0170]$ (0.72)	-0.0018 $[-0.0041, -0.0002]$	0.0044 / *0.0253**	0.0/0.0/100.0
25%	0.7039	0.7067	$+0.0028$ $[-0.0089, 0.0174]$ (0.77)	-0.0022 $[-0.0043, -0.0008]$	0.0026 / *0.0253**	0.7/2.9/96.6
50%	0.7039	0.7066	$+0.0027$ $[-0.0102, 0.0171]$ (0.68)	-0.0012 $[-0.0055, 0.0013]$	0.6376 / 0.9859	1.4/5.8/93.2
75%	0.7039	0.7078	$+0.0038$ $[-0.0090, 0.0195]$ (0.68)	-0.0021 $[-0.0043, -0.0007]$	0.0036 / *0.0253**	2.0/8.8/89.8
100%	0.7039	0.7075	$+0.0036$ $[-0.0077, 0.0180]$ (0.66)	-0.0023 $[-0.0048, -0.0006]$	0.0108 / *0.0497**	2.8/11.7/86.4

Every 95% AUC-delta CI spans zero (raw $p = 0.66$–0.77; BH-adjusted $p = 0.986$ at every level, Section 6.3): coverage alone does not produce a statistically reliable AUC gain at any level tested, confirming the near-flat curve in Figure 1. A McNemar test at a standard 0.5 threshold (seed 42) finds zero discordant predictions at every level – enrichment changes no hard classification at this threshold – reinforcing this paper's choice of ROC-AUC/PR-AUC over fixed-threshold metrics as primary. The PR-AUC delta tells a sharper, formally confirmed story: *four of the five coverage levels (0%, 25%, 75%, 100%) show a small negative PR-AUC delta that remains statistically distinguishable from zero after the manuscript-wide 23-test Benjamini–Hochberg correction* (adjusted $p = 0.025$–0.050, all < 0.05; Table 4); only at 50% does the interval include zero (adjusted $p = 0.986$). This is, together with the LI-Small McNemar result (Section 4.4), one of only two findings in this study that survive the complete, manuscript-wide correction, and it directly supports this paper's own argument that PR-AUC is the more diagnostic metric under this domain's severe imbalance (Section 7): account-risk enrichment's effect on the metric most relevant to this domain is a small, consistently-signed, and now formally confirmed *cost*, not undetectable noise. Even at 100% retained coverage, effective non-zero transaction-level coverage remains limited (2.8% for $target - risk$, 11.7% for $source - risk$) despite far higher account-level coverage, directly explaining why so little of the available coverage translates into measurable benefit.

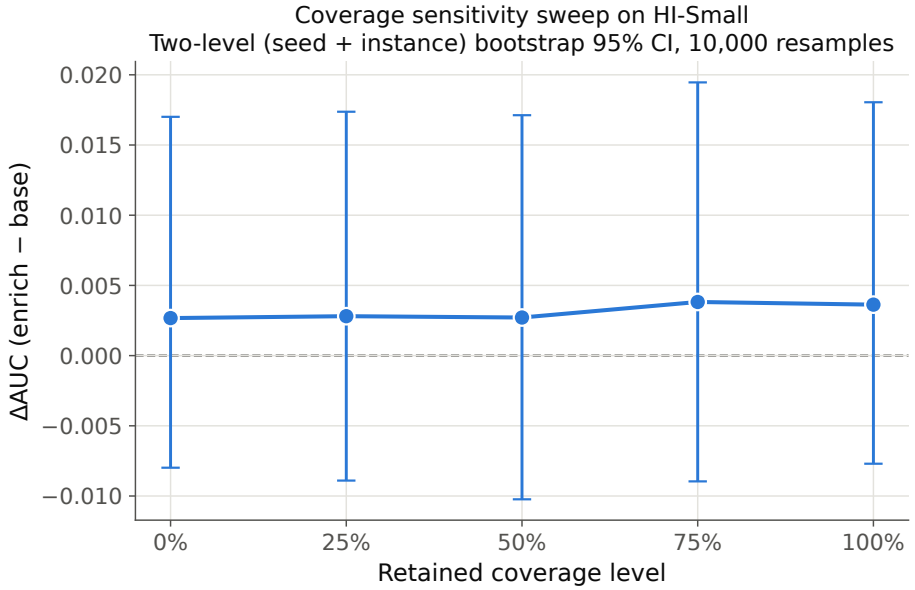

Figure 1. *Coverage sensitivity sweep on HI-Small: two-level bootstrap 95% CIs for ΔAUC (enrich − base) by retained-coverage level. Every interval spans zero (Table 4).*

4.4. LI-Small: A Cold-Start Diagnostic

LI-Small has substantially lower historical account coverage than HI-Small (Table 3). HybridGAT base (AUC= 0.6417, instance-bootstrap 95% CI $[0.6248, 0.6588]$) and enriched (AUC= 0.6404, CI $[0.6230, 0.6573]$) are close, but a paired bootstrap on the same 14,077 instances gives mean $\Delta\mathrm{AUC} = -0.0013$, unadjusted 95% CI $[-0.0025, -0.0001]$, two-sided unadjusted $p = 0.0388$ – *this unadjusted interval excludes zero*, though the result does not survive this paper's manuscript-wide, complete 23-test Benjamini–Hochberg correction (corrected $p = 0.121$, Section 6.3). Rather than "effectively identical," the data support a more precise reading: under LI-Small's low-coverage cold-start

conditions, enrichment produces a small negative AUC delta whose unadjusted evidence points toward a real, not merely null, effect, though this is not established at the corrected significance level. XGBoost on tabular features alone reaches AUC$=0.7386$ (CI $[0.7209, 0.7562]$), consistent with structural features remaining discriminative while enrichment's unadjusted evidence points toward a measurable, if slight, harm to discrimination. A McNemar test comparing XGBoost-tabular against XGBoost+GNN-stacking at a 0.5 threshold gives statistic$=12.19$, $p = 0.00048$ (corrected $p = 0.011$ – *one of only two findings in this study that survive this paper's complete, manuscript-wide 23-test correction*, together with the coverage-sweep PR-AUC result of Section 4.3; Section 6.3): stacking's aggregate AUC (0.7447, CI $[0.7263, 0.7625]$) exceeds tabular's, and this is significant, but the change is not uniformly favorable at the instance level (669 instances gained by stacking versus 804 lost relative to tabular-only) – ranking-metric improvement and instance-level decision consistency are not the same property.

4.5. HI-Small Stacking: A Three-Configuration, Three-Seed Ablation

This paper's earlier draft reported one number for HI-Small stacking: XGBoost on tabular features concatenated with the 64-d HybridGAT embedding (configuration C) reached AUC$=0.8471$, F1$=0.5023$ under P2-warmstart at seed 42 only, with no tabular-only or embeddings-only comparison. That comparison now exists across all three seeds and both P2 protocols, for three configurations: (A) tabular only (11-d); (B) HybridGAT embedding only (64-d); (C) tabular + embedding (75-d, the previously reported configuration).

Table 5. *HI-Small stacking ablation: three-seed (42, 123, 2024) mean ± std ROC-AUC. Configuration A ranks first in 5 of 6 (protocol, seed) cells; per-seed values for configuration C at P2-warmstart are 0.8471/0.7562/0.8245 (seeds 42/123/2024) – seed 42, the value reported in the earlier manuscript draft, is the* highest *of the three, not representative.*

Protocol	Configuration	ROC-AUC (mean ± std)	F1 (mean ± std)
P2-warmstart	A: tabular only	0.8530 ± 0.0019	0.5582 ± 0.0192
P2-warmstart	B: embedding only	0.7912 ± 0.0148	0.3572 ± 0.0169
P2-warmstart	C: tabular + embedding	0.8093 ± 0.0386	0.5417 ± 0.0295
P2-acctgrp	A: tabular only	0.7593 ± 0.0312	0.3506 ± 0.0334
P2-acctgrp	B: embedding only	0.7437 ± 0.0335	0.2917 ± 0.0308
P2-acctgrp	C: tabular + embedding	0.6927 ± 0.0599	0.3494 ± 0.0378

Configuration A reaches a higher mean ROC-AUC than C under both P2-warmstart (0.8530 vs. 0.8093) and P2-acctgrp (0.7593 vs. 0.6927), and ranks first, by ROC-AUC, in five of six (protocol, seed) cells; in the sixth (P2-acctgrp, seed 2024), C exceeds A by 0.0030 AUC. Configuration B ranks last in five of six cells. Configuration C's cross-seed std (0.0386/0.0599) is substantially larger than A's (0.0019/0.0312) or B's (0.0148/0.0335). A distinct pattern accompanies this: configuration C's predicted-probability distribution is markedly compressed relative to A and B at P2-warmstart seed 123 (probability range 0.097 vs. 0.847–0.981 at the other two seeds) and at all three P2-acctgrp seeds – reported here as a directly measured property; its likely mechanism (an early-stopping interaction with the higher-dimensional concatenated input) is discussed in Section 8 but not directly verifiable, since per-iteration boosting curves were not logged. No bootstrap CI or signed-rank test has yet been computed for the A-vs-C difference; the mean, std, and per-seed ranking above are reported at the level of statistical resolution three seeds support. We accordingly withdraw the earlier characterization of HI-Small's stacked configuration as this study's most promising result: the evidence, now that it exists, does not support it.

5. External Validation on Real Crypto AML Data

We restate explicitly: *Elliptic is real crypto illicit-transaction/AML graph validation, not banking AML*, and no result here supports a banking deployment claim. All models use the temporal protocol of Table 2 (train: time step ≤ 34; test: > 34), with loss and metrics restricted to labeled nodes, no account-risk enrichment applied (Section 3.2), and a leakage guard confirming zero of $234{,}355$ edges connect a train-side to a test-side node.

5.1. Tabular Baselines and Topology-Only GNN Sweep

Logistic Regression, XGBoost, and LightGBM, trained once each (single, unseeded run) on Elliptic's 166 node features alone, reach ROC-AUC 0.8813/0.9276/0.9364 respectively (PR-AUC 0.2921/0.7990/0.8038; Table 6). Four topology-only GNNs, three seeds each (42, 123, 2024), 50 epochs, are compared against these: *GraphSAGE is the strongest GNN* (ROC-AUC$=0.8777 \pm 0.0029$, PR-AUC$=0.4014 \pm 0.0420$), ahead of GIN (0.7952 ± 0.0025), GAT (0.7926 ± 0.0103), and GCN (0.7689 ± 0.0129). A Friedman test across the four architectures (seeds as blocks) gives $\chi^2 = 8.20$, raw $p = 0.0421$ (this test does not survive this paper's complete, manuscript-wide 23-test Benjamini–

Hochberg correction, corrected $p = 0.121$, Section 6.3); a Nemenyi post-hoc comparison (critical difference 2.708), which uses its own family-wise critical-difference correction at the pairwise level – a separate procedure from, and not part of, the 23-test BH family – finds only *GraphSAGE vs. GCN* survives *that* correction (rank difference 3.00): under the Nemenyi procedure specifically, GraphSAGE's advantage over the weakest architecture is significant, but its advantage over GAT and GIN, while directionally consistent across all three seeds, does not clear this threshold. Given the omnibus Friedman test itself does not survive the paper-wide correction, we report this Nemenyi finding as a descriptive pairwise comparison under its own, narrower-scope correction, not as independent confirmation beyond what the omnibus test supports.

Table 6. *Elliptic (real crypto illicit/AML graph, not banking AML): topology-only GNN sweep (3 seeds, mean ± std) vs. tabular baselines (single split, node features only). GraphSAGE is the strongest GNN but does not close the gap to XGBoost/LightGBM. Nemenyi post-hoc (critical difference 2.708, $\alpha = 0.05$): only GraphSAGE vs. GCN (rank diff. 3.00) is significant; GraphSAGE vs. GAT/GIN and GCN vs. GAT/GIN are not.*

Family	Model	ROC-AUC	PR-AUC	F1	Seeds
Tabular (baseline)	LogisticRegression	0.8813	0.2921	0.4389	1
Tabular (baseline)	XGBoost	0.9276	0.7990	0.8255	1
Tabular (baseline)	LightGBM	0.9364	0.8038	0.8268	1
Topology-only GNN	GCN	0.7689 ± 0.0129	0.1537 ± 0.0072	0.2626 ± 0.0077	3
Topology-only GNN	GraphSAGE	0.8777 ± 0.0029	0.4014 ± 0.0420	0.4587 ± 0.0104	3
Topology-only GNN	GAT	0.7926 ± 0.0103	0.2068 ± 0.0196	0.3065 ± 0.0138	3
Topology-only GNN	GIN	0.7952 ± 0.0025	0.2155 ± 0.0194	0.3400 ± 0.0118	3

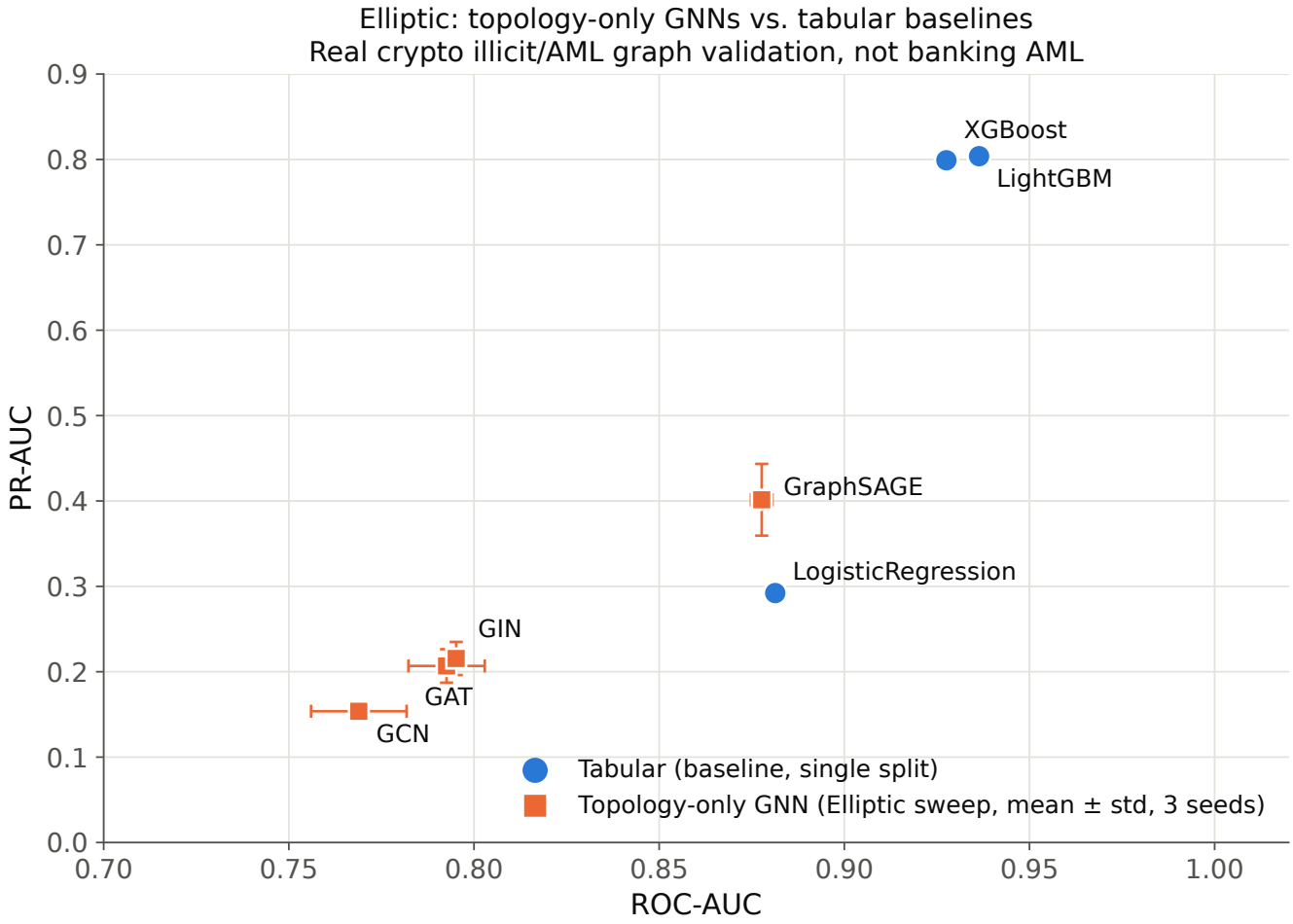

Figure 2. *Elliptic: topology-only GNNs (mean±std, 3 seeds) versus tabular baselines (single split). The best GNN (GraphSAGE) approaches Logistic Regression on ROC-AUC but no GNN closes the gap to XGBoost or LightGBM.*

Critically, even GraphSAGE does not close the gap to tabular gradient boosting: its ROC-AUC (0.878) sits below even Logistic Regression (0.881), the *weakest* tabular baseline, and its PR-AUC (0.401) remains roughly half that of XGBoost/LightGBM ($\approx$ 0.80). The tabular baselines are single, unseeded runs, so this comparison is numerical, not formally tested; the gap's magnitude is large relative to the GNN sweep's own seed-to-seed variability (PR-AUC std$=$ 0.042 for GraphSAGE), making it unlikely to be attributable to tabular-model seed variance alone, though this remains an inference, not a tested claim. This supports the interpretation that on real data, a purely topology-driven GNN does not automatically outperform tabular gradient boosting – the central empirical basis for this paper's framing of graph learning as a technique requiring per-protocol, per-dataset verification.

5.2. Computational Cost and Practical Interpretation

Computational cost. GAT is the most expensive architecture to train (mean 51.0s across seeds), nearly double GCN's cost (27.0s), while GraphSAGE – the best-performing architecture – is the second-cheapest (34.2s); the best-performing architecture is not the most expensive one.

Practical interpretation. Read together with the preceding result – that even the strongest GNN (GraphSAGE) does not close the gap to tabular gradient boosting on Elliptic – the cost comparison reinforces this paper's practitioner-facing recommendation (Section 8): a topology-only GNN is not automatically the right default on real transaction-graph data, on either accuracy or compute grounds, and a strong tabular baseline should be established first before a more expensive graph architecture is adopted. Architecture choice among GNNs, when one is used, should not default to the most expensive option: the cheapest architecture tested here (GCN) was also the weakest, but the strongest (GraphSAGE) was not the most expensive either, so cost and accuracy do not trade off monotonically in this comparison.

6. Stacking and Complementary Graph Signal

Direct account-risk enrichment (HI-Small) and topology-only GNN embeddings alone (Elliptic) are each, on their own, unreliable or insufficient routes to graph-derived signal (Sections 4–5). This section asks whether *stacking* a graph-derived embedding with tabular features, as an input to a gradient-boosted classifier, recovers complementary signal that neither alone does. On Elliptic, a matched, multi-seed ablation shows a small, consistently positive gain across all three evaluated seeds for both classifiers, though $n = 3$ seeds limits formal statistical certainty; on HI-Small, the equivalent ablation (Section 4.5) does not: tabular features alone are competitive with, or superior to, the stacked configuration in five of six evaluated cells. We report both at the level of confidence each independently supports, not as proxies for one another.

6.1. Elliptic

Table 7 and Figure 3 compare, for XGBoost and LightGBM over three seeds, three configurations: (A) tabular only (166-d); (B) GraphSAGE embeddings only (32-d); (C) tabular + embedding (198-d).

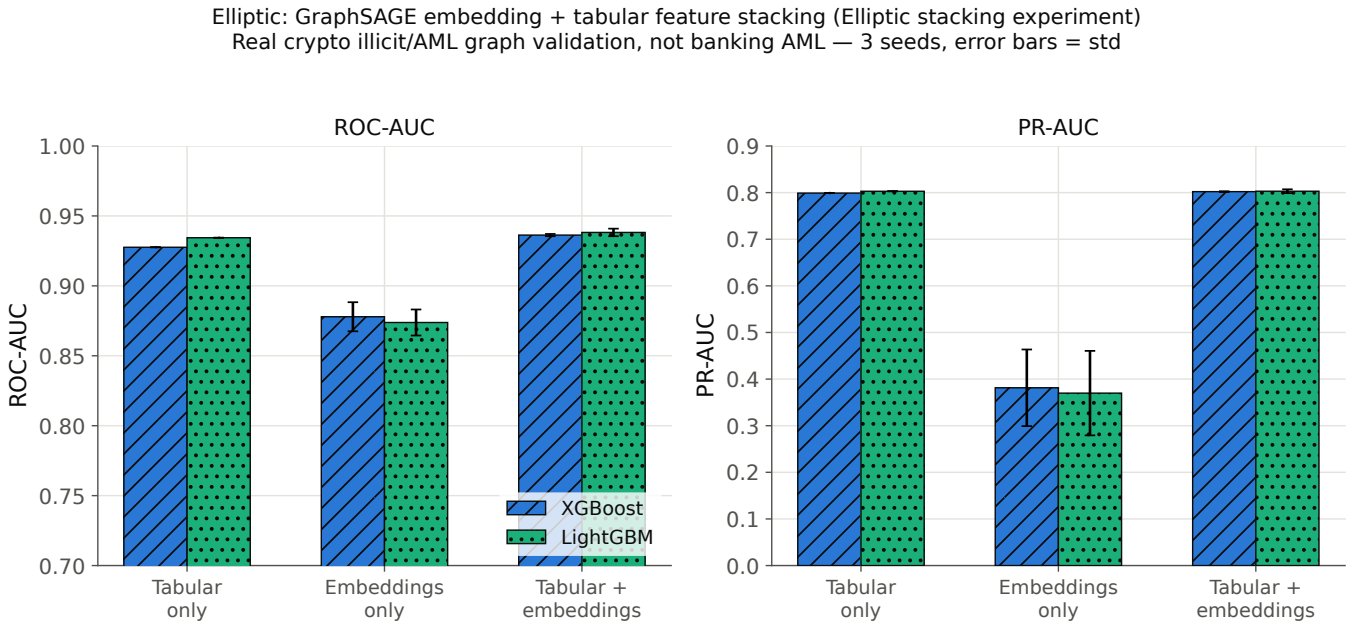

Figure 3. *Elliptic GraphSAGE + tabular stacking, 3 seeds, error bars = std. Embeddings alone (B) underperform tabular-only (A); tabular + embeddings (C) is at or above A.*

Table 7. *Elliptic: GraphSAGE + tabular stacking, 3 seeds, mean ± std (tabular-only row is seed-invariant by construction). Stacking (C) gives a small but consistent gain over tabular-only (A); embeddings alone (B) underperform A.*

Classifier	Feature configuration	ROC-AUC	PR-AUC	F1
XGBoost	A: tabular only (166-d)	0.9276 ± 0.0000	0.7990 ± 0.0000	0.8255 ± 0.0000
XGBoost	B: GraphSAGE embeddings only (32-d)	0.8779 ± 0.0104	0.3813 ± 0.0822	0.4552 ± 0.0476
XGBoost	C: tabular + embeddings (198-d)	0.9363 ± 0.0008	0.8021 ± 0.0008	0.8121 ± 0.0009
LightGBM	A: tabular only (166-d)	0.9344 ± 0.0000	0.8029 ± 0.0000	0.8274 ± 0.0000
LightGBM	B: GraphSAGE embeddings only (32-d)	0.8738 ± 0.0093	0.3699 ± 0.0906	0.4510 ± 0.0496
LightGBM	C: tabular + embeddings (198-d)	0.9382 ± 0.0027	0.8030 ± 0.0041	0.8130 ± 0.0037

Embeddings alone underperform tabular-only for both classifiers (XGBoost 0.8779 ± 0.0104 vs. 0.9276; LightGBM 0.8738 ± 0.0093 vs. 0.9344), consistent with Elliptic's tabular features already carrying most of the signal. Stacking gives a small, consistent gain across all three seeds: XGBoost $0.9276 \rightarrow 0.9363 \pm 0.0008$ ($+0.0087$ AUC, $+0.0031$ PR-AUC); LightGBM $0.9344 \rightarrow 0.9382 \pm 0.0027$ ($+0.0038$ AUC, PR-AUC essentially flat). A Wilcoxon signed-rank test (C vs. A, $n = 3$ seeds) gives $W = 0.0$, $p = 0.25$ for both classifiers – the smallest possible two-sided p-value at this sample size; after this paper's complete, manuscript-wide 23-test Benjamini–Hochberg correction this test does not survive (corrected $p = 0.575$, Section 6.3), and is uninformative at this sample size regardless. A seed-level bootstrap of the mean delta (10,000 resamples, not itself assigned a formal p-value and so not part of the 23-test corrected family) gives a 95% CI of $[0.0078, 0.0095]$ for XGBoost and $[0.0007, 0.0057]$ for LightGBM, both excluding zero on an unadjusted basis: stacking improved ROC-AUC in all three evaluated seeds for both classifiers, providing consistent positive evidence, although the small number of seeds limits formal statistical certainty.

6.2. HI-Small

Table 5 (Section 4.5) reports the matched three-configuration, three-seed HI-Small ablation. Configuration A (tabular only) exceeds configuration C (tabular + embedding) on mean ROC-AUC under both protocols and ranks first in five of six (protocol, seed) cells; no CI, signed-rank test, or effect size has yet been computed for this difference, so it is reported at the descriptive, ranking level three seeds support, without a formal significance claim.

6.3. Cross-Dataset Synthesis

This is a cross-dataset stacking analysis examining a candidate boundary condition for graph–tabular complementarity (contribution C4): stacking is evidenced at two distinct outcomes, not one uniform conclusion reported at two confidence levels – consistently positive but small gains across all three evaluated seeds on Elliptic (formal certainty limited by $n = 3$), versus no measured advantage on HI-Small. This divergence is consistent with a structural difference between the two tabular baselines – HI-Small's already includes engineered graph-derived statistics ($target - risk$, $source - risk$, fan-in, fan-out), whereas Elliptic's 166 features contain none – so a learned embedding has room to supply genuinely new information on Elliptic but not on HI-Small. This contrast is consistent with, but does not causally establish, the hypothesis that graph embeddings are more useful when they contribute information not already encoded by the tabular features; feature redundancy was not independently manipulated in this study, so this remains a supported interpretation, not a proven causal law.

Multiple-comparison correction. Every formal hypothesis test with a computed p-value was included in a single manuscript-wide family of 23 tests and corrected using the Benjamini–Hochberg procedure at FDR $= 0.05$. The correction was implemented using a reproducible script and independently cross-checked against the `statsmodels` reference implementation. Adjusted p-values for the principal comparisons are reported in the corresponding results and in Table 4. The family comprises: the coverage-sweep ΔAUC bootstrap tests (5), the coverage-sweep McNemar tests (5), the coverage-sweep ΔPR-AUC bootstrap tests (5), the P2-warmstart/P2-acctgrp Wilcoxon tests (2), the Elliptic Friedman test (2, computed separately for ROC-AUC and PR-AUC), the Elliptic stacking Wilcoxon tests (2), and the LI-Small paired-bootstrap and McNemar tests (2). *Five of the 23 tests survive*: the LI-Small McNemar test (raw $p = 0.00048$, adjusted $p = 0.011$) and *four of the five coverage-sweep ΔPR-AUC bootstrap tests* – 0%, 25%, 75%, *and* 100% *retained coverage (adjusted* $p = 0.025$–0.050*, all* < 0.05*; only* 50% *does not, adjusted* $p = 0.986$*).* Eighteen of the 23 tests do not survive, including the Elliptic Friedman test (raw $p = 0.0421$, adjusted $p = 0.121$), the LI-Small paired-bootstrap AUC delta (raw $p = 0.0388$, adjusted $p = 0.121$), both Elliptic stacking Wilcoxon tests (raw $p = 0.25$, adjusted $p = 0.575$), all five coverage-sweep ΔAUC bootstrap tests (adjusted $p = 0.986$), all five coverage-sweep McNemar tests (adjusted $p = 1.0$), and both P2 Wilcoxon tests (adjusted $p \geq 0.986$). The Elliptic and HI-Small stacking seed-level bootstrap confidence intervals (Sections 6.1, 6.2) were not assigned a formal p-value in this study and are accordingly not part of this 23-test family; we report them at their unadjusted level and do not describe them as confirmed or formally significant. Completing this correction strengthens, rather than weakens, this paper's central statistical narrative: a second, independent line of evidence (the coverage sweep's effect on PR-AUC, not only its already-reported AUC-based non-effect) now also

survives formal correction, directly supporting this paper's own argument that PR-AUC is the more diagnostic metric under this domain's severe imbalance (Section 7).

7. Operational Metrics and Calibration

Where saved probability outputs were available, models were evaluated using the shared metrics module (Section 3.3): ROC-AUC, PR-AUC, Precision/Recall@k, Brier score, and ECE, with the documented exceptions of Section 3.3 (no calibration metrics for the HI-Small stacking ablation; no bootstrap confidence intervals for single-split experiments lacking saved per-instance predictions; no formal confidence interval for the HI-Small stacking configuration-A-vs-C difference). ROC-AUC averages ranking quality over the entire score distribution, including the overwhelming majority of true negatives an analyst would never review; under the severe imbalance present throughout this study (positive rates 9.76% on Elliptic's labeled nodes down to 0.173% on Credit Card Fraud), a model can reach high ROC-AUC while ranking poorly in the small top fraction an analyst can actually review. PR-AUC is more sensitive to this regime – directly illustrated by the coverage sweep (Section 4.3), where the PR-AUC delta remains statistically distinguishable from zero at four of five coverage levels after this paper's complete, manuscript-wide 23-test Benjamini–Hochberg correction (Section 6.3), while the AUC delta does not survive that same correction at any of them.

The clearest illustration occurred on Credit Card Fraud (*a real, non-AML benchmark used only as a robustness check*): LightGBM reached a superficially respectable ROC-AUC$= 0.9117$, yet its PR-AUC was only 0.0371, with Precision@$1\% = 0.047$ (Recall@$1\% = 0.276$) versus Precision@$1\% = 0.154$ (Recall@$1\% = 0.898$) for both Logistic Regression and XGBoost on the same split. We verified this reflects LightGBM's default class-imbalance handling (`is_unbalance=True`), not a metrics artifact; no per-instance data exists to formally test this specific comparison, so it is reported as a single-split diagnostic finding about metric choice, not a criticism of LightGBM generally.

Calibration. No model underwent post-hoc calibration (Platt scaling, isotonic regression); Brier/ECE describe native, uncalibrated output and are a diagnostic of *relative* calibration, not a claim of absolute calibration. Tabular gradient boosting is consistently the best-calibrated family: on Elliptic, XGBoost/LightGBM reach Brier ≈ 0.024 versus 0.135–0.292 for the four GNNs (GraphSAGE, the strongest and best-calibrated GNN, still reaches only Brier$= 0.135$). Stacking on Elliptic also produced the *best* calibration of the three configurations for both classifiers (Brier$= 0.0226$ under configuration C vs. 0.0241 tabular-only) – a small point in the same direction as its ROC-AUC/PR-AUC gain. No calibration metric was computed for the HI-Small stacking ablation.

8. Discussion

Read together, Sections 4–7 support one central claim: the measured benefit of graph-derived signal – whether as direct account-risk enrichment, a topology-only embedding, or a stacked combination with tabular features – depends on the evaluation protocol, the historical coverage available, and, for stacking, on how much of the graph-derived information a learned embedding could supply is already present in the tabular features it is combined with. We organize the interpretation below around four synthesis questions.

Why does enrichment's benefit depend on coverage and protocol (C1–C2)? The mechanism is structural, not statistical. Under account-grouped splits, $source - risk$ coverage in validation is exactly 0.0% in every fold – a set-theoretic consequence of partitioning accounts into disjoint train/validation groups while indexing a statistic by that same key, true by construction for any dataset with this account structure. $target - risk$'s partial survival ($\approx$ 12.6–14.0% under P1) is the informative asymmetry: destination accounts are not the grouping key, so fan-in/hub typologies (Section 1) let a training-seen destination account still appear in validation. This is why the same technique produces such different apparent effects across P1 (structurally zero opportunity), P2-warmstart (small, non-significant gain), P2-acctgrp (near-zero), and LI-Small (a small negative *delta*, unadjusted 95% CI $[-0.0025, -0.0001]$, that does not survive this paper's complete, manuscript-wide 23-test correction, Section 6.3): coverage and protocol interact multiplicatively, since protocol determines how many observations are even possible before reliability can be asked at all. The coverage sweep's PR-AUC delta is the sharper instrument here: it *does* survive this same correction at four of five coverage levels (Section 4.3), so the cost of low coverage is formally confirmed on the metric this domain's imbalance makes most diagnostic, even where the AUC delta is not. A plausible mechanism for LI-Small's negative delta specifically is that feeding two near-constant, near-fallback channels into HybridGAT under very low coverage may dilute the contribution of the genuinely informative structural features the model would otherwise rely on more heavily – a cost, not merely an absence of benefit; this is offered as a plausible, not independently tested, account. This framework operationalizes, for account-risk enrichment specifically, the same concern the leakage-safe and inductive-evaluation literature raises generally [21, 22]: a reported gain is not interpretable without the conditions that produced it.

Why can topology-only GNNs trail strong tabular baselines (C3)? That GraphSAGE (ROC-AUC$= 0.878$) does not exceed even Logistic Regression (0.881) on Elliptic is this study's most consequential single finding. Two non-exclusive, Elliptic-specific mechanisms are plausible. First, Elliptic's 166 undocumented features already carry substantial signal on their own (XGBoost/LightGBM exceed ROC-AUC$= 0.93$ from features alone), so the marginal information a message-passing step can add is smaller and may be dominated by the noise that aggregation also introduces. Second, and

independently convergent with a recent critical re-evaluation of GNNs on structurally adjacent Bitcoin fraud data, where real topology underperformed edge-shuffled random topology and a Random Forest outperformed every GNN tested [20], real transaction-graph topology under adversarial/evasive behavior does not necessarily carry a reliable homophily signal. The Friedman test itself does not survive this paper's manuscript-wide correction, and the Nemenyi post-hoc result establishes a narrower claim than a point-estimate ranking alone would suggest: under Nemenyi's own, separate pairwise correction, only GraphSAGE's advantage over GCN survives; its advantage over GAT/GIN, though directionally consistent across all three seeds, does not. A plausible architectural account is that GraphSAGE's sampled-neighbor aggregation is less sensitive to GCN's uniform degree-normalization, which can dilute signal in the heterogeneous-degree neighborhoods transaction graphs plausibly exhibit – stated as a plausible mechanism, not a tested one. Either way, architecture choice among GNNs matters less here than the more basic choice of whether to use topology as the primary signal at all.

When do graph embeddings add non-redundant information (C4)? A gradient-boosted meta-learner receiving an embedding as one input among many can, via its own tree-splitting, learn a data-dependent weighting – leaning on the embedding where informative, ignoring it otherwise – which is why stacking can outperform both direct enrichment (no equivalent down-weighting mechanism) and an embedding used alone (no tabular fallback). This mechanism has a precondition it cannot supply itself: it can only extract value from information the tabular block does not already contain. Elliptic's 166 features contain no graph-derived quantity, so a topology-derived embedding has room to add new information (seed-bootstrap CI excluding zero for both classifiers, Section 6.1); HI-Small's tabular configuration already includes $target - risk$, $source - risk$, fan-in, and fan-out, bounding an embedding's marginal contribution, and the evidence does not show this margin translating into a reliable gain (configuration A ranks first in five of six cells). This pattern examines a candidate boundary condition for graph–tabular complementarity: it is consistent with, but does not causally establish, the redundancy hypothesis, since this study did not manipulate feature redundancy as a controlled variable. This divergence extends the same caution the graph-liability literature raises [20] to embedding-based stacking specifically: even a combination method designed to be robust to an uninformative graph signal is not automatically beneficial once tested against a matched, multi-seed baseline – the HI-Small number reported in the earlier manuscript draft (AUC$=$ 0.8471, seed 42) was the highest of three seeds, not a representative one, a fact invisible in a single-seed evaluation. Separately, LI-Small's McNemar test ($p = 0.00048$) shows stacking's aggregate AUC gain there is not uniformly favorable at the instance level (804 instances lost vs. 669 gained relative to tabular-only) – aggregate ranking improvement and instance-level decision consistency are not the same property.

What does this mean for practical AML/fraud evaluation (C5)? LightGBM's Credit Card Fraud result (ROC-AUC$=$ 0.9117, Precision@1% $=$ 0.047, versus 0.154 for Logistic Regression/XGBoost) mechanistically illustrates why ROC-AUC alone is insufficient under severe imbalance: it averages over the majority of true negatives an analyst never reviews, while PR-AUC and Precision@k isolate exactly the top-ranked regime that matters operationally – the same divergence appears, and is formally confirmed by this paper's complete 23-test correction, in the coverage sweep's PR-AUC-vs-AUC result (Section 4.3). Tabular gradient boosting is also consistently the best-calibrated family (Brier $\approx$ 0.024 on Elliptic vs. 0.135–0.292 for GNNs), and Elliptic stacking improves calibration alongside discrimination. This adds a concrete data point to the cold-start-fraud literature [23, 24], which has so far established cold-start's severity primarily through absence-of-benefit findings; this study's LI-Small result adds an unadjusted-bootstrap-based negative signal to that picture, though it does not reach significance after this paper's complete, manuscript-wide 23-test correction (Section 6.3). For practitioners, the evidence supports a bounded recommendation: establish a strong tabular baseline first, report coverage and protocol alongside any enrichment claim, and verify stacking with a multi-seed, matched ablation – not a single run – checking specifically whether the existing tabular feature set already encodes the relational information an embedding would supply. This paper introduces no new GNN architecture, encoding scheme, or leakage-detection algorithm; its contribution is the coverage-aware, protocol-aware discipline itself (C1), shown to matter empirically across three dataset families, including when applied to this paper's own previously most prominent and least-tested result.

9. Limitations and Future Work

Synthetic and boundary-condition data. HI-Small/LI-Small share one IBM AMLSim generative process [10]; whether the specific coverage thresholds and delta magnitudes reported (including the LI-Small enrichment delta, significant only on an unadjusted basis, Section 6.3) transfer quantitatively to real banking data is not tested here. Elliptic provides genuine external validation that the topology-ceiling and stacking-benefit findings are not IBM AMLSim artifacts, but its transaction-level nodes have no persistent account entity, so it does not validate banking-AML deployment (license: CC BY-NC-ND 4.0). Credit Card Fraud is used only as a non-AML operational-metrics check; no per-instance artifact enables a formal test of its central comparison, so that result is reported at single-split confidence. No proprietary banking AML dataset was available.

Seed count and non-determinism. Three seeds (42, 123, 2024) were used for P2-warmstart, P2-acctgrp, the coverage sweep, the Elliptic GNN sweep, and both stacking ablations; *P1 and P3 (LI-Small) are single-seed (*42*)*, a deliberate computational trade-off – extending every protocol to three seeds was infeasible within budget, which was prioritized for the multi-seed cross-dataset analyses – not an inconsistency

or an oversight; the Elliptic tabular baselines are likewise single, unseeded runs. All GNN training used CPU-only PyTorch Geometric, and the underlying logs document training dynamics that are not bit-for-bit reproducible even under an identical seed (an observed discrepancy at seed 2024 between independent runs); some enrichment deltas are small enough that this is a live, not merely theoretical, threat to exact reproducibility, though not to the qualitative pattern of results. HI-Small's graph is constructed from shared source/destination accounts; Elliptic's is used as-is from its official edge list – results should be compared at the level of qualitative findings, not absolute numbers.

Open questions specific to this revision. No unenriched HybridGAT model was evaluated under P1's exact 5-fold protocol, so P1's argument rests on the structural, provable $source - risk$-coverage result alone; a direct empirical base-model comparison remains untested. The HI-Small stacking ablation's probability-compression pattern (Section 6) is an open question: whether it reflects an early-stopping interaction with the higher-dimensional concatenated input cannot be verified without per-iteration boosting curves, which were not logged. Of the 23 formal statistical tests reported in this paper – one complete, manuscript-wide family, computed by a reproducible script and independently cross-checked against `statsmodels` (Section 6.3) – five survive Benjamini–Hochberg correction at FDR $= 0.05$: the LI-Small McNemar test and four of the five coverage-sweep ΔPR-AUC bootstrap tests. We adopt the corrected reading as authoritative throughout. This does not change the reading of this paper's AUC-based claims (coverage sensitivity, P2 base-vs-enriched, LI-Small AUC delta, Elliptic Friedman, Elliptic stacking), none of which were significant, or were significant only before correction; it does, however, add a second, independently corrected-significant finding (the coverage sweep's PR-AUC cost) to this paper's evidentiary base, alongside the LI-Small McNemar result. F1's operating threshold is selected in-sample (confirmed by inspection of the evaluation code, Section 3.3: the threshold maximizing F1 is chosen on the same split being reported, not a held-out validation split), making it an optimistic diagnostic measure; this paper's main conclusions rely on threshold-independent ROC-AUC/PR-AUC, and replacing this procedure with a validation-selected threshold remains a direction for future revisions.

Scope. Nothing in this paper constitutes a claim of production readiness, regulatory certification, or legal compliance; the 10:1 negative-subsampling ratio was fixed for tractability and not swept for sensitivity. This paper does not claim account-risk enrichment or embedding-based stacking is universally beneficial or harmful, or that any of the four GNN architectures is a general-purpose recommendation – all such claims would contradict the protocol-, coverage-, and dataset-dependence this paper establishes.

Responsible use and ethical considerations. This study did not involve human subjects; no ethics approval or consent to participate was required. HI-Small and LI-Small are fully synthetic; Elliptic and Credit Card Fraud (ULB) are already publicly released in anonymized form, and no re-identification of any individual or account was attempted on any dataset; none of the three data families contains directly identifiable personal information as used in this study. Consistent with the Scope paragraph above, this paper's findings do not generalize automatically beyond the specific datasets, protocols, and coverage conditions under which they were measured, and nothing here should be read as a recommendation to deploy account-risk enrichment, a topology-only GNN, or graph–tabular stacking as a fully automated AML or fraud decision without human review; this paper's own bounded practical recommendation (Section 8) is to complement, not replace, analyst judgment. This distinction matters specifically because the three data families are not interchangeable: HI-Small/LI-Small are synthetic banking AML, Elliptic is real crypto-AML, and Credit Card Fraud is real but not AML evidence at all (Section 3.1); no result in this paper should be read across that boundary. AI-assisted tools were used during drafting, editing, and code-review support; the authors reviewed, verified, and are responsible for the final content, claims, results, and conclusions.

Future work. Near-term and requiring no new data collection: instrumenting the HI-Small stacking fit to test the probability-compression hypothesis directly; two additional seeded P1 runs plus a new base-model 5-fold run to enable a direct P1 comparison; two additional seeded Elliptic tabular runs. Medium-term: a leakage-safe, inductive account-risk analogue for transaction-graph datasets such as Elliptic, computed only from strictly earlier time steps; a typology-stratified (fan-out, fan-in, hub, cycle) breakdown, contingent on data availability; a sensitivity sweep of the 10:1 subsampling ratio; a controlled sweep withholding individual relational features to directly test this paper's stacking boundary-condition hypothesis. Long-term: validation against real, proprietary banking AML data, contingent on institutional access; broader adoption, by this sub-field, of the bootstrap/signed-rank/multiple-comparison-correction infrastructure applied here as standard reporting practice; and future work will extend the evaluation to larger subgraph-level resources such as Elliptic2 when adequate computational resources are available.

10. Conclusion

This paper presented a protocol-aware, coverage-aware, multi-dataset evaluation of graph learning for AML and fraud detection, spanning a synthetic banking-AML benchmark (HI-Small, LI-Small), a real crypto-AML graph (Elliptic), and a complementary real non-AML benchmark (Credit Card Fraud). The central finding is not that GNNs, enrichment, or any architecture "wins," but that each technique's value depends on the evaluation protocol, the historical coverage available, the dataset, and the representation through which graph information is injected. Account-risk enrichment has no opportunity to contribute signal under strict account-grouped splits ($source-risk$ coverage mechanically 0.0%; no matched baseline exists to test a comparison claim); its effect

on AUC is a small, largely non-significant delta under the most favorable warm-start protocol and under LI-Small's low-coverage cold-start conditions (LI-Small's negative AUC delta does not survive this paper's complete, manuscript-wide 23-test Benjamini–Hochberg correction). Its effect on PR-AUC, however, is formally confirmed: a small, consistently negative cost that survives this same correction at four of the five coverage-sensitivity levels tested, and the LI-Small instance-level McNemar comparison likewise survives – the two findings in this study that hold under the complete corrected family. Topology-only GNNs, across four architectures on real Elliptic data, consistently underperformed tabular gradient boosting. Stacking is evidenced at two distinct outcomes, not one: consistently positive but small gains across all three evaluated seeds on Elliptic (166 features, no graph-derived quantity; formal certainty limited by $n = 3$ seeds), and no measured gain on HI-Small (whose tabular features already include $target - risk$, $source - risk$, fan-in, fan-out) – a candidate boundary condition for when a graph-derived embedding is worth adding via stacking, consistent with but not causally establishing the hypothesis that embeddings help most where they are non-redundant with the tabular features. Operational metrics beyond ROC-AUC (PR-AUC, Precision/Recall@k, calibration) proved equally important on Credit Card Fraud, where a competitively-ROC-AUC model ranked poorly exactly where a fixed analyst review budget matters most.

We draw four practical implications: (i) report coverage and split construction alongside any enrichment claim, since the same technique can appear beneficial, neutral, or harmful depending on these factors alone; (ii) treat a strong tabular gradient-boosting baseline as the default comparison point; (iii) before adopting embedding-based stacking, check whether the tabular feature set already includes engineered relational statistics – where it does not, this study's Elliptic evidence (one real graph dataset, three seeds, small deltas, and incomplete corrected significance) suggests stacking may be considered as a candidate extension, but it should be accepted only after a matched, multi-seed ablation and evaluation with operational and calibration metrics, not assumed low-risk; where the tabular features already include such statistics, this study's HI-Small evidence does not support assuming a benefit at all; and (iv) treat small AUC deltas with caution, since values of a few thousandths are close to this class of experiment's noise floor.

ORCID

0009-0001-7074-7674 (Karim Bettaieb)
0000-0001-9219-5228 (Afef Kacem Echi)
0009-0002-9652-8004 (Houcem Hammami)

Abbreviations

AML	Anti-Money Laundering
AUC	Area Under the Curve
ROC-AUC	Receiver Operating Characteristic Area Under the Curve
PR-AUC	Precision–Recall Area Under the Curve
GNN	Graph Neural Network
GCN	Graph Convolutional Network
GAT	Graph Attention Network
GIN	Graph Isomorphism Network
ECE	Expected Calibration Error
FDR	False Discovery Rate
BH	Benjamini–Hochberg

Author Contributions

Karim Bettaieb: Conceptualization, Data Curation, Formal Analysis, Investigation, Methodology, Software, Validation, Visualization, Writing – original draft, Writing – review & editing

Afef Kacem Echi: Conceptualization, Methodology, Supervision, Validation, Writing – review & editing

Houcem Hammami: Formal Analysis, Methodology, Validation, Writing – review & editing

Data Availability Statement

All experiments use public datasets only; no proprietary banking or institutional data was used. HI-Small/LI-Small are synthetic, from the IBM AMLSim benchmark family [10], subject to its official access conditions. Elliptic [9] was obtained via its official Kaggle release under CC BY-NC-ND 4.0, limited to non-commercial academic research. Credit Card Fraud (ULB) was obtained via its official Kaggle release under the Database Contents License (DbCL) 1.0; it is not an AML dataset. Public datasets must be obtained from their original providers under their respective access conditions and licenses. The shareable research artifact contains preprocessing code, experiment configurations, scripts, and result files; it does not redistribute dataset content where the original license prohibits such redistribution. Code used to produce this paper's results, including preprocessing, experiment, and figure/table-generation scripts, is available upon reasonable request to the corresponding author. An automated test suite accompanies the codebase and verifies temporal-leakage guards, dataset integrity, fixed-seed propagation, and reproducible split construction. It does not guarantee bit-for-bit deterministic GNN training, whose numerical non-determinism is documented in Section 9.

Conflicts of Interest

The authors declare no conflicts of interest.

References

[1] T. N. Kipf and M. Welling, "Semi-Supervised Classification with Graph Convolutional Networks," in Proceedings of the International Conference on Learning Representations (ICLR), 2017.

[2] W. L. Hamilton, Z. Ying, and J. Leskovec, "Inductive Representation Learning on Large Graphs," in Advances in Neural Information Processing Systems (NeurIPS), vol. 30, 2017.

[3] P. Velickovic, G. Cucurull, A. Casanova, A. Romero, P. Liò, and Y. Bengio, "Graph Attention Networks," in Proceedings of the International Conference on Learning Representations (ICLR), 2018.

[4] K. Xu, W. Hu, J. Leskovec, and S. Jegelka, "How Powerful Are Graph Neural Networks?" in Proceedings of the International Conference on Learning Representations (ICLR), 2019.

[5] T. Chen and C. Guestrin, "XGBoost: A Scalable Tree Boosting System," in Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD), 2016, pp. 785–794. https://doi.org/10.1145/2939672.2939785

[6] G. Ke, Q. Meng, T. Finley, T. Wang, W. Chen, W. Ma, Q. Ye, and T.-Y. Liu, "LightGBM: A Highly Efficient Gradient Boosting Decision Tree," in Advances in Neural Information Processing Systems (NeurIPS), vol. 30, 2017.

[7] Y. Liu, X. Ao, Z. Qin, J. Chi, J. Feng, H. Yang, and Q. He, "Pick and Choose: A GNN-Based Imbalanced Learning Approach for Fraud Detection," in Proceedings of the Web Conference (WWW), 2021, pp. 3168–3177. https://doi.org/10.1145/3442381.3449989

[8] Y. Dou, Z. Liu, L. Sun, Y. Deng, H. Peng, and P. S. Yu, "Enhancing Graph Neural Network-based Fraud Detectors against Camouflaged Fraudsters," in Proceedings of the ACM International Conference on Information and Knowledge Management (CIKM), 2020, pp. 315–324. https://doi.org/10.1145/3340531.3411903

[9] M. Weber, G. Domeniconi, J. Chen, D. K. I. Weidele, C. Bellei, T. Robinson, and C. E. Leiserson, "Anti-Money Laundering in Bitcoin: Experimenting with Graph Convolutional Networks for Financial Forensics," in Proceedings of the KDD Workshop on Anomaly Detection in Finance, 2019.

[10] E. Altman, J. Blanua, L. von Niederhäusern, B. Egressy, A. Anghel, and K. Atasu, "Realistic Synthetic Financial Transactions for Anti-Money Laundering Models," in Proceedings of the NeurIPS Datasets and Benchmarks Track, 2023.

[11] B. Johannessen and M. Jullum, "Finding Money Launderers Using Heterogeneous Graph Neural Networks," arXiv preprint, 2023.

[12] L. Akoglu, H. Tong, and D. Koutra, "Graph-Based Anomaly Detection and Description: A Survey," Data Mining and Knowledge Discovery, vol. 29, no. 3, pp. 626–688, 2015. https://doi.org/10.1007/s10618-014-0365-y

[13] V. Van Vlasselaer, C. Bravo, O. Caelen, T. Eliassi-Rad, L. Akoglu, M. Snoeck, and B. Baesens, "APATE: A Novel Approach for Automated Credit Card Transaction Fraud Detection Using Network-Based Extensions," Decision Support Systems, vol. 75, pp. 38–48, 2015. https://doi.org/10.1016/j.dss.2015.04.013

[14] T. Pourhabibi, K.-L. Ong, B. H. Kam, and Y. L. Boo, "Fraud Detection: A Systematic Literature Review of Graph-Based Anomaly Detection Approaches," Decision Support Systems, vol. 133, pp. 113303, 2020. https://doi.org/10.1016/j.dss.2020.113303

[15] S. Kaufman, S. Rosset, C. Perlich, and O. Stitelman, "Leakage in Data Mining: Formulation, Detection, and Avoidance," ACM Transactions on Knowledge Discovery from Data (TKDD), vol. 6, no. 4, pp. 1–21, 2012. https://doi.org/10.1145/2382577.2382579

[16] D. Micci-Barreca, "A Preprocessing Scheme for High-Cardinality Categorical Attributes in Classification and Prediction Problems," ACM SIGKDD Explorations Newsletter, vol. 3, no. 1, pp. 27–32, 2001. https://doi.org/10.1145/507533.507538

[17] L. Prokhorenkova, G. Gusev, A. Vorobev, A. V. Dorogush, and A. Gulin, "CatBoost: Unbiased Boosting with Categorical Features," in Advances in Neural Information Processing Systems (NeurIPS), vol. 31, 2018.

[18] A. Dal Pozzolo, O. Caelen, R. A. Johnson, and G. Bontempi, "Calibrating Probability with Undersampling for Unbalanced Classification," in Proceedings of the 2015 IEEE Symposium Series on Computational Intelligence (SSCI), Cape Town, South Africa, 2015, pp. 159–166. https://doi.org/10.1109/SSCI.2015.33

[19] C. Guo, G. Pleiss, Y. Sun, and K. Q. Weinberger, "On Calibration of Modern Neural Networks," in Proceedings of the 34th International Conference on Machine Learning (ICML), Proceedings of Machine Learning Research, vol. 70, 2017, pp. 1321–1330.

[20] S. Maganti, "When Graph Structure Becomes a Liability: A Critical Re-Evaluation of Graph Neural Networks for Bitcoin Fraud Detection under Temporal Distribution Shift," arXiv preprint, 2026. https://doi.org/10.48550/arXiv.2604.19514

[21] K. Hayat and B. Magnier, "Data Leakage and Deceptive Performance: A Critical Examination of Credit Card Fraud Detection Methodologies," Mathematics, vol. 13, no. 16, 2563, 2025. https://doi.org/10.3390/math13162563.

[22] H. Khaleghpour and B. McKinney, "Leakage Safe Graph Features for Interpretable Fraud Detection in Temporal Transaction Networks," arXiv preprint, 2026. https://doi.org/10.48550/arXiv.2603.06632

[23] B. Deprez, W. Wei, W. Verbeke, B. Baesens, K. Mets, and T. Verdonck, "Advances in Continual Graph Learning for Anti-Money Laundering Systems: A Comprehensive Review," WIREs Computational Statistics, vol. 17, no. 3, 2025. https://doi.org/10.1002/wics.70040

[24] W. Zhang, R. Li, Q. Bai, and S. Wang, "SparseFraudNet: A Graph-based Approach for Cold-start Fraud Detection with Information Aggregation," ACM Transactions on Information Systems, 2025. https://doi.org/10.1145/3748719

[25] S. Seabold and J. Perktold, "statsmodels: Econometric and Statistical Modeling with Python," in Proceedings of the 9th Python in Science Conference (SciPy), 2010. https://doi.org/10.25080/Majora-92bf1922-011