EIGF: An Explainable Graph Neural Network Fusion Approach for Fraud Risk Detection in Insurance Claims

Tiejiang Sun[1], Xu Han[2], Jiaying Chen[3,*] and Yao Ge[4]

[1]Chang'an University, Xi'an, China

[2]California State Polytechnic University, Pomona, United States

[3]Cornell University, Ithaca, United States

[4]University of Miami, Coral Gables, United States

[*]Corresponding author: Jiaying Chen. jc2744@cornell.edu

Abstract

Screening insurance claims for fraud poses three difficulties at once. Positives are scarce, the entities under review are linked to one another, and every alert consumes investigator time, so a workable system has to rank well and show its evidence. We describe EIGF, an Explainable Insurance Graph Fusion framework in which a tabular risk prior is corrected by relational signal from a GraphSAGE branch. The data are public Medicare provider records released under CC BY 4.0. We collapse 558,211 inpatient and outpatient claims into 60 leakage-controlled provider features covering 5,410 providers, 506 of whom carry a potential-fraud label. Shared beneficiaries and shared physicians give rise to two provider graphs, each built with inverse-entity-degree weighting followed by top-k pruning. Class-weighted focal loss trains the graph branch; a balanced histogram gradient-boosting branch captures nonlinear tabular risk. A convex weight, tuned on validation precision-recall area under the curve (PR-AUC) alone, mixes the two probabilities and lets the model back away from noisy graph propagation when that helps. Over five stratified 60/20/20 splits EIGF reaches a test PR-AUC of 0.7211±0.0358, ROC-AUC of 0.9464±0.0114, F1 of 0.6554±0.0287, and recall of 0.7663±0.0429. Mean PR-AUC gains are 0.0102 over histogram gradient boosting and 0.0103 over GraphSAGE. Against the graph branch the gain holds on every split (one-sided exact Wilcoxon p=0.03125); against the tabular branch the direction is consistent, but five splits cannot settle it. Our explanation package draws on SHAP, integrated gradients, relation ablation, neighborhood evidence, and a masking-based fidelity test. Zeroing the five highest-ranked features cuts the mean risk probability of high-confidence fraudulent providers by 0.3839, against 0.0549±0.0727 for random five-feature masks. Two further findings matter for practice: beneficiary-sharing edges carry more information than physician-sharing edges, and on some splits the edges hurt. Unconditional message passing is therefore the wrong default, and reliability-aware fusion the safer one. Every value reported here comes from the accompanying executable code and its saved experiment outputs.

Keywords

Insurance fraud, Medicare claims, graph neural networks, GraphSAGE, explainable artificial intelligence, class imbalance, evidence fusion.

1. Introduction

Claims arrive as rows, but they do not behave as independent transactions. One provider may bill the same beneficiaries over and over, share physicians with neighboring providers, or take

part in coordinated behavior that leaves no trace inside any single claim record. Rules engines and tabular classifiers still earn their place in production, yet neither has a way to represent these links. Graph neural networks (GNNs) do: they let node attributes be combined with evidence drawn from connected entities, and the fraud literature reports gains on device-sharing, transaction, and healthcare graphs [1], [6], [8], [9]. Swapping an established tabular model for a GNN is nonetheless harder than it looks, for three practical reasons.

Labels come first. They are rare and they are imperfect: in this dataset 9.35% of providers carry a potential-fraud mark, which makes accuracy meaningless and lets ROC-AUC look healthy while precision at any useful recall stays poor. PR-AUC is what we optimize and report as the primary ranking metric, with ROC-AUC, F1, precision, recall, and specificity kept alongside it for operational reading [18], [19]. The second issue is that building a graph is itself a modeling assumption. A shared beneficiary can mean a referral pattern, geographic overlap, continuity of care — or coordinated abuse. A shared physician is just as likely to be legitimate. Passing messages across such edges risks smoothing fraudulent and legitimate providers into each other, a failure mode documented for fraud graphs under camouflage and heterophily [14]–[16]. The last issue is the evidence trail an investigator needs. A bare risk score, with no account of which features, relations, and neighbors produced it, resists validation and invites automated action that nobody has checked.

EIGF is our response to all three. We make no claim that a pure GNN wins in general; in EIGF graph learning supplies relational evidence that complements a strong tabular prior. Two models estimate risk in parallel. A balanced histogram gradient-boosting model works directly on provider-level aggregates. A two-layer GraphSAGE model works on the same attributes after propagation over provider graphs derived from shared beneficiaries and shared physicians. A convex fusion, its weight chosen on validation data, decides per split how far the graph evidence should be trusted. The design is deliberately conservative. Informative relations get to improve the ranking; noisy ones leave the tabular branch as an anchor.

We contribute four things. The first is a reproducible provider-graph construction that discounts high-degree shared entities and caps neighborhood size, using no labels at any point. The second is a transparent graph-tabular fusion, its weight selected on validation data, for ranking provider risk under heavy class imbalance. Third comes a multi-view explanation protocol that lines up SHAP feature evidence, integrated-gradient graph evidence, relation ablation, and labeled training-neighbor context. Fourth, we test explanation fidelity directly, intervening on the top-ranked features and measuring the probability drop against random masks. Each claim below is bounded by an experiment we ran. Where the diagnostics are negative we say so: on one split, self-only graph inference beats relational inference, and we report that instead of reading every graph relation as a benefit.

One framing note. The task here is risk screening; legal adjudication is someone else's job. The target field in the public data carries the name PotentialFraud, and it means a provider-level flag that warrants review — never proof that anyone intended to defraud. We position the model accordingly, as a tool that prioritizes work for human investigators.

2. Related Work

A. Insurance and Healthcare Fraud Detection

Domain rules, statistical learning, and data mining of both supervised and unsupervised kinds have been combined in fraud detection for decades. Ngai et al. sorted financial-fraud analytics into classification, clustering, regression, visualization, and outlier-detection families [2]. Joudaki et al. surveyed data-mining work on healthcare fraud and drew attention to how hard reliable labels are to come by [3]. On the Medicare side, researchers have drawn on public payment data, exclusion lists, random undersampling, neural networks, and feature selection

[4], [5], [32]. Two findings recur across this body of work. Class imbalance has to be handled on purpose, and provider-level aggregates — service volume, reimbursements, beneficiary counts — are strong signals.

Relationships entered the healthcare literature more recently. Yoo et al. ran GraphSAGE over provider, physician, and beneficiary data [6]. Chen et al. added spatiotemporal constraints to medical-insurance graphs [28]; Lu et al. built an attributed heterogeneous information network with hierarchical attention [29]; Zhang et al. proposed hierarchical multimodal fusion on dynamic heterogeneous graphs [30]. Hong et al. brought together topology, feature, and semantic channels [27], and Muhammad et al. paired heterogeneous and relation-embedded GNNs with graph explainers on activity-level medical claims [7]. Explainable Medicare fraud models built on tabular learners have been studied separately [31]. Our goal and our experimental posture both differ from these. We stay with public provider-level screening, we hold the splits identical while comparing graph learning against strong tabular baselines, and we let the graph contribution shrink whenever the relations prove unreliable.

B. Graph-Based Fraud Learning

Organized fraud suits graph methods well, since coordinated actors tend to reuse the same devices, accounts, addresses, beneficiaries, or intermediaries. Liang et al. built a device-sharing network for e-commerce return-freight insurance and showed what network learning is worth operationally [8]. InfDetect took the idea further, into a large-scale platform covering device, transaction, friendship, and buyer-seller graphs [9]. Our inductive neighborhood-aggregation backbone is GraphSAGE [11]; GCN and GAT remain the foundational alternatives [10], [12]. Surveys track how quickly graph representation learning and its industrial applications have developed [13].

Ordinary neighborhood averaging works because of homophily, and fraud graphs break that assumption. CARE-GNN answered with relation-aware aggregation designed to resist camouflaged fraudsters [14]. PC-GNN attacked severe class imbalance through label-balanced sampling and neighborhood selection [15]. AUC-oriented GNN optimized ranking objectives directly for imbalanced fraud detection [16]. All three argue for careful relation handling. None of them removes the possibility that a badly constructed graph is worth less than the raw attributes. EIGF keeps a standard, auditable GraphSAGE branch for that reason and puts robustness at the fusion layer, in place of an opaque stack of neighbor selectors. Recent work in a related direction applies dynamic heterogeneous graph contrastive learning to preserve structural semantics when fraud labels are scarce [33].

C. Explainable Graph Learning

How much does an input variable move a prediction? LIME, SHAP, and integrated gradients all estimate this [20], [21], [25]. GNNs have their own tools: GNNExplainer isolates influential subgraphs and node features, while PGExplainer learns a parameterized explanation generator [22], [23]. Surveys note that a graph explanation may target features, edges, nodes, paths, or prototypes, and that evaluation should measure fidelity — visual plausibility on its own proves little [24]. Insurance screening adds one requirement. An explanation has to separate evidence intrinsic to a provider from evidence that arrived through its relations.

We are not proposing a new universal explainer. What EIGF assembles is a reproducible evidence package for a hybrid risk model: TreeSHAP on the tabular branch, integrated gradients on the graph branch, a normalized consensus attribution, relation-specific ablation, and a list of influential labeled training neighbors. On top of these sits a feature-mask intervention, which asks whether the globally top-ranked features really do support the high-confidence predictions. Auditability drove the design. Attention scores and edge weights are not treated as explanations until something validates them.

3. Data and Graph Formulation

A. Public Dataset and Prediction Unit

Our experiments run on version 1 of the Healthcare Provider Fraud Detection Analysis dataset, distributed through Mendeley Data under CC BY 4.0 (DOI: 10.17632/552r52kmgb.1) [26]. Its four training files hold a provider-level PotentialFraud label, beneficiary attributes, 40,474 inpatient claims, and 517,737 outpatient claims. Merging the claim tables yields a corpus of 558,211 claims tied to 138,556 unique beneficiaries and 5,410 providers, among which 506 are positive (9.353%) and 4,904 negative. Claim dates run from late 2008 through December 2009. Some fall before 2009 because an episode can begin before the claim end date on record.

We predict at the provider level, matching an investigation workflow where a risk engine ranks billing entities for audit. No label ever enters feature construction or edge construction. Raw identifiers — Provider, BeneID, ClaimID, and the physician fields — serve only for grouping and for graph topology, and none is encoded as a predictive categorical feature. The main experiment reported here contains no synthetic observations.

Table I. Dataset and Graph Summary

Item	Value	Role / note
Claims	558,211	Inpatient + outpatient
Providers	5,410	Prediction nodes
Potential-fraud providers	506 (9.353%)	Positive class
Beneficiaries	138,556	Shared-entity relation
Provider features	60	No label-derived features
Beneficiary graph	68,309 edges	29 isolated nodes
Physician graph	6,243 edges	1,482 isolated nodes

B. Provider-Level Feature Engineering

Sixty numeric features come out of training-independent transformations, in four groups. Claim-volume variables cover total, inpatient, and outpatient claim counts, unique beneficiaries, claims per beneficiary, active months, and monthly entropy. Financial variables summarize reimbursed and deductible amounts by sum, mean, median, standard deviation, and maximum. Clinical-utilization variables record claim duration, inpatient stay duration, diagnosis and procedure counts, admission-diagnosis frequency, and how many distinct attending, operating, and other physicians appear. Beneficiary-composition variables aggregate age, sex, race, renal-disease indicators, chronic-condition prevalence, mortality status, coverage months, and annual reimbursement and deductible amounts.

Aggregation leaves some numeric summaries missing; we fill those with zero. Within each split a StandardScaler is fitted on the training providers alone, then applied to validation and test providers, which keeps holdout distributions from leaking in. The feature table shipped with the code package carries human-readable names and raw provider aggregates, so any input can be audited.

C. Relation Construction

Bipartite incidence data yields two provider-provider relations. Under the beneficiary view, providers i and j connect when at least one beneficiary shows up in claims from both. Under the physician view, they connect when they share an attending, operating, or other physician. Projected naively, such graphs end up dominated by the most common entities. We therefore

let an entity e that connects d_e providers contribute an inverse-degree weight to each provider pair:

$$w_{ij}^{(r)} = \Sigma_e I(i,e)\, I(j,e) / \log2(1 + d_e), \quad r \in \{bene, phys\}. \tag{1}$$

A rare shared entity therefore counts for more than a ubiquitous one. Before symmetrization we keep only the top 20 weighted neighbors per provider, which helps both scalability and oversmoothing. The beneficiary graph that results has 68,309 undirected edges and just 29 isolated nodes. Its physician counterpart is far sparser: 6,243 edges, 1,482 isolated nodes. For message passing we row-normalize the union adjacency A=A_bene+A_phys. Edges are built once, from identifiers, with no target labels involved.

Our setting is transductive. Representation learning sees every provider node and every unlabeled relation, while the loss reads only training labels and model selection reads only validation labels. What this measures is semi-supervised risk ranking over a known provider network. Inductive performance on providers that appear later, and temporal generalization, fall outside it; we leave both to future work.

4. Proposed Eigf Method

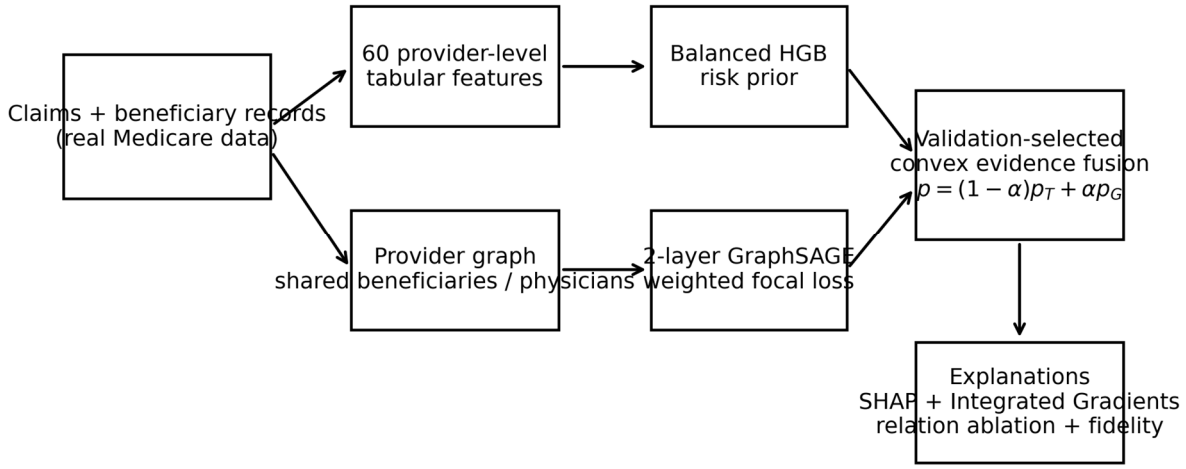

Figure 1. EIGF architecture. A tabular risk prior and a GraphSAGE relational branch are fused using a validation-selected convex weight; explanations cover feature, relation, and neighborhood evidence.

A. Tabular Risk Prior

A histogram gradient-boosting classifier (HGB) forms the tabular branch: 300 boosting iterations, learning rate 0.05, at most 31 leaves per tree, L2 regularization 1.0, balanced class weights. Tree ensembles make awkward baselines for GNN studies, because they pick up nonlinear interactions and thresholds in aggregated claims with no representation learning at all. We kept this strong baseline on purpose; a linear comparator would have been easier to beat and less informative. Write p_T(i) for the fraud probability predicted for provider i.

Balanced logistic regression, random forest, XGBoost, and a feed-forward multilayer perceptron serve as additional baselines. Every one of them gets the same standardized features, the same provider splits, and a decision threshold selected the same way on validation data.

B. GraphSAGE Relational Branch

For the graph branch we use a two-layer mean-aggregation GraphSAGE network. At layer l the row-normalized adjacency yields a neighborhood message $m_i^{(l)}=\Sigma_j A_{ij} h_j^{(l)}$, and separate linear transformations handle the self and neighbor messages:

$$h_i^{(l+1)} = \mathrm{ReLU}(W_s^{(l)} h_i^{(l)} + W_n^{(l)} m_i^{(l)}). \quad (2)$$

Hidden layers hold 64 and 32 units, each followed by dropout 0.25, and a final linear layer emits the logit z_i with $p_G(i)=\mathrm{sigmoid}(z_i)$. Two layers buy a two-hop receptive field while keeping oversmoothing in check. We optimize with AdamW at learning rate 0.004, weight decay 4×10^{-4}, gradients clipped at 5, and early stopping on validation PR-AUC.

Imbalance needs handling, and we prefer not to invent synthetic graph nodes, so the graph model carries a positive-class-weighted focal loss. Given label y_i, predicted probability p_i, and $p_t=p_i$ for positives or $1-p_i$ for negatives:

$$L = -(1 - p_t)^{\gamma} [\, y \log(p)\cdot\omega_+ + (1-y) \log(1-p) \,], \quad (3)$$

with $\gamma=1.5$ and ω_+ set to the negative-to-positive ratio of the training set. Focal modulation damps the easy examples; the positive weight offsets class prevalence [17], [18].

C. Validation-Selected Evidence Fusion

The two pure models fail differently. Relational dependence is invisible to the tabular model, while the GNN happily propagates noise, legitimate referral patterns, and dominant majority-class signals alike. EIGF mixes their probabilities convexly:

$$p_{EIGF}(i) = (1-\alpha)\, p_T(i) + \alpha\, p_G(i), \quad 0 \le \alpha \le 1. \quad (4)$$

We search α on a grid from 0 to 1 in steps of 0.01, scoring each candidate by validation PR-AUC and breaking ties first on validation F1, then on proximity to $\alpha=0.5$. Whatever weight wins is frozen before any test evaluation. Averaged over five splits α comes to 0.514 ± 0.154 — neither branch dominates consistently. Note that this is a grid search, not a learned stacking model: it adds no trainable parameters, which keeps the risk score easy to reproduce and easy to decompose.

The classification threshold likewise comes from validation alone, maximizing F1 over probability quantiles. PR-AUC and ROC-AUC need no threshold; precision, recall, specificity, and F1 all use the frozen validation value.

D. Multi-View Explanation

Explanations are computed once model selection is done. TreeSHAP supplies exact additive attributions for the HGB branch [20]. Integrated gradients attributes the GraphSAGE logit to each standardized provider feature along a zero baseline, which after standardization is the training-feature mean [21]. Globally, we normalize absolute SHAP and integrated-gradient magnitudes separately and then combine them using the same fusion weight α. A uniform average would flatten the two branches together; weighting by α preserves what each one actually contributes operationally. Organizing heterogeneous anomaly signals into traceable, human-readable audit trails has proven valuable elsewhere in the fraud literature [35].

Relation ablation comes next: we run the trained graph model over four adjacency views — union, beneficiary only, physician only, self only. Nothing is retrained here. The intervention is diagnostic, and it measures how the learned model reacts when a relation disappears. Last is a

local case report, listing a provider's top feature attributions together with its strongest training-set neighbors, relation-specific edge weights, and known training labels. Neighbor evidence never exposes test or validation labels. Comparable work on knowledge-enhanced, multi-source explanation supports the general argument that fraud detection need not collapse into a single opaque score [36].

To gauge fidelity we mask the globally top five consensus features to zero for the 80 highest-confidence positive test providers on the best validation split, then compare the mean drop in fused probability against 100 random five-feature masks. If an explanation is worth anything, the features it nominates should move the model further than randomly chosen ones. Causal fraud mechanisms lie beyond what such an intervention can establish; faithfulness to the fitted predictor is what it tests.

5. Experimental Design

A. Splits, Baselines, and Implementation

Five stratified random seeds — 13, 37, 73, 101, 131 — generate nonoverlapping 60% training, 20% validation, and 20% test provider sets per run, each test set holding roughly 1,082 providers and 101 positives. All models see the same split. Hyperparameters stay fixed across seeds, and only three quantities are read off validation data: early-stopping epoch, fusion weight, classification threshold.

The tabular baselines are balanced logistic regression, a 500-tree random forest capped at depth 12, histogram gradient boosting, XGBoost with 600 estimators and scale-positive weight, and a two-hidden-layer MLP trained under the same focal loss family. On the graph side we run GraphSAGE over the union graph plus an explicitly relation-weighted variant. That variant never beat the simpler union model during development. We release its results anyway, for transparency.

Everything ran in Python, on pandas, NumPy, SciPy sparse matrices, scikit-learn, XGBoost, PyTorch, and SHAP. Our graph implementation relies on PyTorch sparse matrix multiplication, so PyTorch Geometric is not needed. Saved CSV, JSON, NPZ, and model checkpoint files let any reported value be reconstructed.

B. Metrics and Statistical Comparison

At 9.35% positive prevalence, PR-AUC reflects the precision-recall trade-off most directly, which is why we make it primary [19]. ROC-AUC scores ranking over positive-negative pairs, though it can understate false-positive burden. F1 summarizes the operating point chosen on validation. Recall gives the share of flagged potential-fraud providers recovered; precision, the fraction of alerts positive under the dataset label; specificity, the fraction of negatives left unalerted.

We report mean±sample standard deviation over five splits. Five observations carry limited inferential power, so we read the statistics with care. A one-sided exact paired Wilcoxon test evaluates our pre-specified directional hypothesis, that fusion improves PR-AUC over either individual branch. Five splits cannot support a claim of broad population-level superiority, and we make none.

6. Results

A. Main Predictive Performance

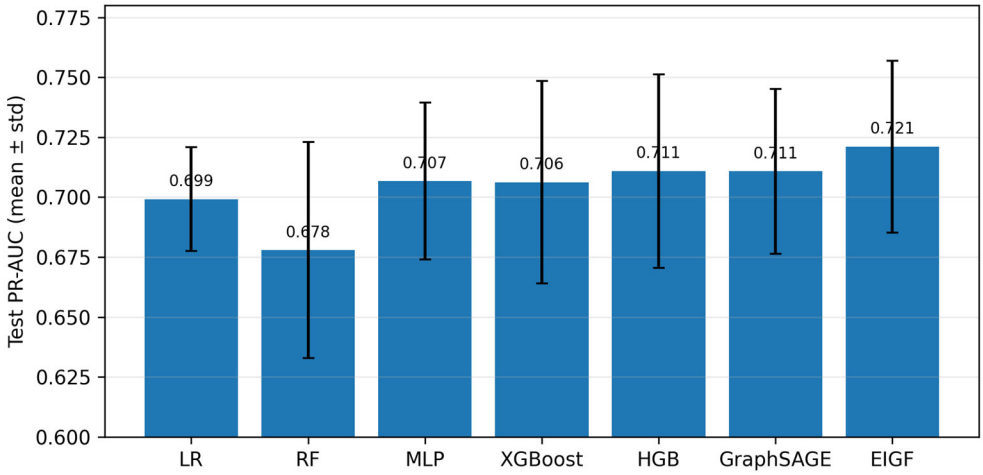

Figure 2. Mean test PR-AUC across five stratified splits. Error bars show one sample standard deviation.

Table II. Test Performance Over Five Splits

Model	ROC-AUC	PR-AUC	F1	Recall
Logistic regression	0.9433±0.0113	0.6992±0.0217	0.6399	0.7248
Random forest	0.9437±0.0119	0.6779±0.0451	0.6624	0.7584
MLP	0.9396±0.0125	0.7067±0.0328	0.6409	0.6653
XGBoost	0.9441±0.0103	0.7062±0.0422	0.6469	0.7564
HGB	0.9407±0.0087	0.7109±0.0403	0.6472	0.7208
GraphSAGE	0.9441±0.0116	0.7108±0.0343	0.6461	0.7030
EIGF	0.9464±0.0114	0.7211±0.0358	0.6554	0.7663

Mean PR-AUC is highest for EIGF at 0.7211±0.0358, against 0.7109±0.0403 for HGB and 0.7108±0.0343 for GraphSAGE, absolute gains of 0.0102 and 0.0103. The gain over GraphSAGE holds on all five splits, with a one-sided exact Wilcoxon p-value of 0.03125. Over HGB it holds on four of five, and there the p-value is 0.09375 — directional evidence, short of conventional significance. Compared with HGB, mean recall climbs from 0.7208 to 0.7663 while mean precision slips from 0.5904 to 0.5732. Where a missed suspicious provider costs more than an extra review, that trade is worth making; local audit capacity should decide how far to push it.

Among the single models GraphSAGE takes the highest mean ROC-AUC (0.9441), with HGB almost identical on mean PR-AUC. The highest single-baseline mean F1 belongs to random forest (0.6624), a reminder that model ranking follows whichever operating objective one picks. We built EIGF mainly for ranking and high-recall screening; optimizing F1 alone was never the target.

Split-specific fusion weights range from 0.34 to 0.68. That spread is exactly what our central design assumption predicts: graph evidence has value, yet its reliability shifts with the labeled sample and with neighborhood composition. A fixed 50/50 average, or a mandate to deploy graph-only, would paper over the uncertainty.

B. Review-Budget Utility

Table III. Mean Precision/Recall At Fixed Review Budgets

Model	P@5%	R@5%	P@10%	R@10%	P@20%	R@20%
HGB	0.789	0.430	0.618	0.667	0.418	0.899
GraphSAGE	0.778	0.424	0.622	0.671	0.414	0.889
EIGF	0.796	0.434	0.620	0.669	0.421	0.905

Ranking metrics become audit workload once a review budget is fixed. Taking the top 5% of a test set means opening 55 files out of 1,082. At that budget EIGF reaches precision 0.7964±0.0394 and recall 0.4337±0.0215, an 8.53-fold lift over the 9.35% test prevalence. Widen to 10%, or 109 providers, and precision is 0.6202 with recall 0.6693. At 20%, or 217 providers, recall reaches 0.9050 and precision 0.4212.

All three models stay close across these budgets. GraphSAGE edges ahead on mean precision and recall at 10%; EIGF is strongest on precision at 5% and on recall at 20%. None of this establishes universal dominance, and it is not meant to. Fusion improves the overall precision-recall ranking, while the choice of a local operating budget belongs to validation data and audit capacity.

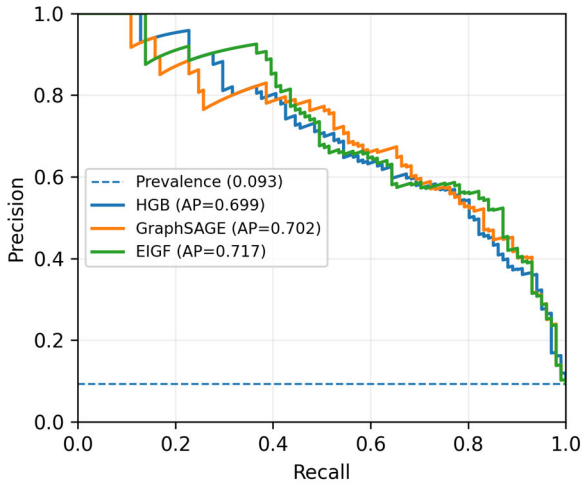

Figure 3. Precision-recall curves on seed 101, the split selected for detailed explanations by validation performance.

C. Relation Diagnostics and Ablation

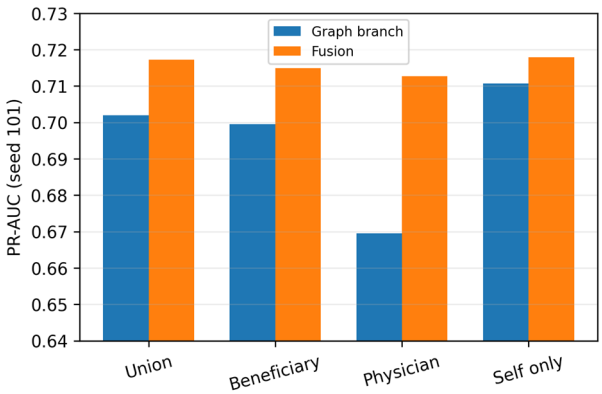

Figure 4. Inference-time relation ablation on seed 101. Self-only removes neighbor messages but retains trained self transformations.

Table IV. Relation Ablation on Seed 101

Graph view	Graph PR-AUC	Fusion PR-AUC	Fusion ROC-AUC
union	0.7020	0.7172	0.9439
beneficiary only	0.6996	0.7149	0.9438
physician only	0.6696	0.7127	0.9377
self only	0.7107	0.7179	0.9448

On the union graph the graph branch reaches PR-AUC 0.7020 and the fusion 0.7172. Strip out physician edges and fused PR-AUC settles at 0.7149. Run physician edges alone and the graph branch falls to 0.6696, the fusion to 0.7127. Within this split, then, the beneficiary relation carries considerably more useful evidence. Part of the explanation is coverage: the physician graph leaves 1,482 providers isolated and its weighted degree is low.

One negative result deserves emphasis. Self-only graph inference attains graph PR-AUC 0.7107 and fused PR-AUC 0.7179, edging past the union view on this split. The five-split fusion result survives this, since the trained graph branch still contributes a distinct nonlinear representation and fusion beats GraphSAGE consistently. What the result does show is that relational propagation can hurt locally, and that our edge semantics are not uniformly homophilous. Provider-specific relation gates, or learned temporal edge reliability, belong in a future version. We have avoided retrofitting any such mechanism after seeing test outcomes.

D. Explanation Results and Fidelity

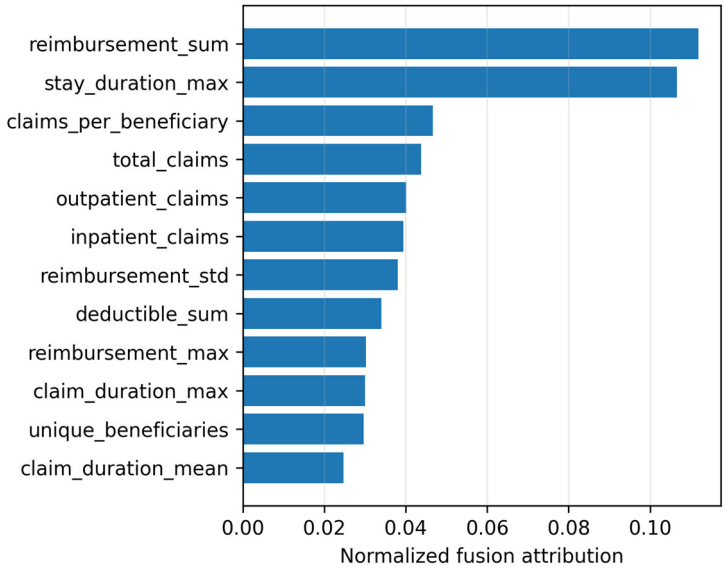

Figure 5. Global EIGF feature evidence on seed 101, combining normalized absolute TreeSHAP and integrated-gradient attributions.

Ten features top the consensus ranking: total reimbursement, maximum inpatient stay duration, claims per beneficiary, total claim count, outpatient claim count, inpatient claim count, reimbursement variability, total deductible, maximum reimbursement, and maximum claim duration. Clinically and operationally this reads as utilization intensity plus payment magnitude. It is emphatically not a causal list of fraudulent behaviors. A large reimbursement total may simply reflect legitimate specialization or unusually severe cases; what the model picks up is a combination in context, never a single forbidden value.

Fidelity gets a quantitative check from the masking intervention. Across the 80 highest-confidence positive test providers, setting the five top consensus features to their training means drops the fused fraud probability by 0.3839 on average. The 100 random five-feature masks produce an average drop of 0.0549, standard deviation 0.0727 — roughly a seventh of the top-feature effect. Reconstructed HGB probabilities match the saved predictions to a maximum absolute error of 0.0, which confirms that the explanation script refits the tabular branch exactly for the saved split.

Take provider PRV56008 as a local example, a positive test provider whose fused probability is 0.8881 (HGB 0.9965, GraphSAGE 0.7880, α=0.52). Leading evidence: 1,370 outpatient claims, 1,370 total claims, reimbursement sum 476,600, 686 unique beneficiaries, 1.997 claims per beneficiary. Its strongest training neighbors, all reached through shared beneficiaries, include providers of both classes. PRV55977 is a positive training neighbor with beneficiary-edge weight 175.23; PRV55969 is a negative one with weight 80.13. Presenting supporting and counter-evidence side by side keeps the explanation from hardening into a one-sided narrative. Provider identifiers in this dataset are pseudonymous codes, and the case is illustrative — no allegation attaches to any real-world named entity.

7. Discussion

A. Why Fusion Helps

What the experiments show is complementarity, not the decisive victory of one representation. HGB does well because provider aggregates already summarize volume, payment, and clinical complexity. GraphSAGE does well because shared entities introduce contextual smoothing and a different nonlinear transformation. Since the two make different errors, validation-weighted probability averaging lifts PR-AUC even on splits where the graph edges by themselves do nothing. The distinction is worth keeping straight. Credit belongs to robust evidence fusion, and not to any claim — which we could not support — that all insurance relations exhibit fraud homophily.

Selected α values double as a coarse reliability indicator. Low α says the validation graph is worth less than tabular risk; high α says relational representation is adding ranking information. In deployment one could track α over time and let a large shift trigger a graph-quality audit. Provider-specific uncertainty estimates would be finer-grained, and future models may well go there, but on 5,410 nodes a global weight audits more easily and overfits less.

B. Operational Use and Governance

We see EIGF supporting a staged investigation workflow. Risk probabilities rank the providers, the selected threshold carves out an alert queue, and the explanation package hands an analyst the features, relations, and known training neighbors to review. From there investigators go to the underlying claims, documentation, policy rules, and clinical context. Denying claims automatically, sanctioning providers, or inferring intent all fall outside what the system should do. A label called PotentialFraud may be noisy or may encode historical investigation practice, and predictions built on it can carry those biases forward.

Governance needs several controls: thresholds calibrated to audit capacity, protected-attribute review, drift monitoring, temporal backtesting, and logging of model output alongside investigator disposition. Our study aggregates beneficiary attributes to provider level, yet any real deployment still owes privacy, access-control, and minimum-necessary obligations. Explanations cut review time, and they can also manufacture false confidence. Fidelity to a model says nothing about whether the model is correct or the decision process fair.

C. Limitations

Evaluation design is our first limitation. Random provider splits are transductive and can drop linked providers into different partitions. For semi-supervised node classification with hidden labels that is appropriate, but it simulates nothing about deployment on future claims; a temporal split and inductive evaluation on newly appearing providers are what would. A second limitation is the graph projection, which collapses a heterogeneous claims network down to provider-provider edges. Computation gets simpler. Event time, diagnosis semantics, edge direction, and multi-entity paths are all lost.

Our target is a potential-fraud flag whose provenance and adjudication standard the public release does not fully document, so performance here measures agreement with a dataset label — not recovered financial loss, and not confirmed legal fraud. We also use only five random splits; they expose variance and enable paired comparisons, yet they cannot carry strong inferential claims. Explanation fidelity is assessed by feature masking, which may well construct inputs outside the observed joint distribution. And the relation ablation, as reported above, finds physician-sharing edges weak and self-only inference competitive. Read the current model, then, as a reproducible benchmark and fusion framework. It is not a finished production architecture.

D. Future Work

Four directions look most pressing to us: temporal, leakage-resistant benchmarks; inductive provider onboarding; learned edge reliability; and uncertainty-calibrated fusion. A heterogeneous temporal GNN could hold on to provider-beneficiary-physician-claim event structure while restricting messages to historical evidence. Relation gates could be regularized toward the tabular prior, then audited by counterfactual edge removal. Evaluation should widen to cost-sensitive precision at fixed review budgets, calibration error, subgroup performance, investigator agreement, and prospective yield. Before anyone generalizes beyond this public dataset, multi-site validation is essential. Continual graph learning under strategic adversarial drift is a further avenue worth pursuing for nonstationary deployment [34].

8. Conclusion

We have presented EIGF, an explainable graph-tabular fusion framework for provider-level insurance fraud risk detection. Working on a real public Medicare claims dataset, it brings together 60 provider features, shared-beneficiary and shared-physician graphs, weighted-focal GraphSAGE, balanced histogram gradient boosting, and a convex fusion whose weight is selected on validation data. Across five splits the framework reaches mean PR-AUC 0.7211 and recall 0.7663, improving average PR-AUC over both of its constituent branches — and we report the graph failures rather than hiding them. The accompanying explanation protocol surfaces high-impact utilization and reimbursement features, exposes relation-specific behavior alongside labeled neighbor context, and shows that masking top-ranked features moves predictions far more than masking random ones.

Our central finding is methodological. Graph information is evidence, and like any evidence its reliability has to be validated before it is trusted; treating it as a guaranteed source of improvement is a mistake. Transparent fusion offers a workable middle route between strong tabular models and relational learning, and it keeps the decomposition of risk auditable throughout.

9. Data and Code Availability

Dataset: Healthcare Provider Fraud Detection Analysis, Mendeley Data, Version 1, DOI 10.17632/552r52kmgb.1, CC BY 4.0 [26]. The accompanying package contains preprocessing,

training, fusion, explanation, figure-generation code, processed data derived from the licensed source, model checkpoints, split indices, and all result tables. The raw source files are not duplicated in the paper package; a download helper and provenance notice are included.

10. Reproducibility Note

Every numerical result in this manuscript is read from saved experiment artifacts generated by the included code. During development we also tried a tabular automobile-insurance dataset and several more complex relation-gating variants, dropping each when its graph formulation or validation performance proved inferior. None of those exploratory runs contributes to the main claims.

References

[1] Motie, S., & Raahemi, B. (2024). Financial fraud detection using graph neural networks: A systematic review. Expert Systems with Applications, 240, 122156. https://doi.org/10.1016/j.eswa.2023.122156.

[2] Ngai, E. W. T., Hu, Y., Wong, Y. H., Chen, Y., & Sun, X. (2011). The application of data mining techniques in financial fraud detection: A classification framework and an academic review of literature. Decision Support Systems, 50(3), 559–569. https://doi.org/10.1016/j.dss.2010.08.006

[3] Joudaki, H., Rashidian, A., Minaei-Bidgoli, B., Mahmoodi, M., Geraili, B., Nasiri, M., & Arab, M. (2015). Using data mining to detect health care fraud and abuse: A review of literature. Global Journal of Health Science, 7(1), 194–202. https://doi.org/10.5539/gjhs.v7n1p194

[4] Bauder, R. A., & Khoshgoftaar, T. M. (2018). The detection of Medicare fraud using machine learning methods with excluded provider labels. In Proceedings of the 31st International Florida Artificial Intelligence Research Society Conference (FLAIRS-31) (pp. 404–409).

[5] Johnson, J. M., & Khoshgoftaar, T. M. (2019). Medicare fraud detection using neural networks. Journal of Big Data, 6, 63. https://doi.org/10.1186/s40537-019-0225-0

[6] Yoo, Y., Shin, D., Han, D., Kyeong, S., & Shin, J. (2022). Medicare fraud detection using graph neural networks. In 2022 International Conference on Electrical, Computer and Energy Technologies (ICECET) (pp. 1–5). https://doi.org/10.1109/ICECET55527.2022.9872963

[7] Muhammad, R., Tbaishat, D., Nazir, A., Yacoub, S., AbdulRazek, M., Abo El-Enen, M. A., & Sahlol, A. T. (2025). Fraud detection and explanation in medical claims using GNN architectures. Scientific Reports, 15, 41734. https://doi.org/10.1038/s41598-025-22910-6

[8] Liang, C., Liu, Z., Liu, B., Zhou, J., Li, X., Yang, S., & Qi, Y. (2019). Uncovering insurance fraud conspiracy with network learning. In Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval (pp. 1181–1184). https://doi.org/10.1145/3331184.3331372

[9] Chen, C., Li, Z., Sun, X., Tian, G., Jiang, J., Zhang, K., & Wang, H. (2019). InfDetect: A large scale graph-based fraud detection system for e-commerce insurance. In 2019 IEEE International Conference on Big Data (Big Data) (pp. 1765–1773). https://doi.org/10.1109/BigData47090.2019.9006115

[10] Kipf, T. N., & Welling, M. (2017). Semi-supervised classification with graph convolutional networks. In International Conference on Learning Representations.

[11] Hamilton, W. L., Ying, R., & Leskovec, J. (2017). Inductive representation learning on large graphs. In Advances in Neural Information Processing Systems 30 (pp. 1024–1034).

[12] Veličković, P., Cucurull, G., Casanova, A., Romero, A., Liò, P., & Bengio, Y. (2018). Graph attention networks. In International Conference on Learning Representations.

[13] Wu, Z., Pan, S., Chen, F., Long, G., Zhang, C., & Yu, P. S. (2021). A comprehensive survey on graph neural networks. IEEE Transactions on Neural Networks and Learning Systems, 32(1), 4–24. https://doi.org/10.1109/TNNLS.2020.2978386

[14] Dou, Y., Liu, Z., Sun, L., Deng, Y., Peng, H., & Yu, P. S. (2020). Enhancing graph neural network-based fraud detectors against camouflaged fraudsters. In Proceedings of the 29th ACM International Conference on Information and Knowledge Management (pp. 315–324). https://doi.org/10.1145/3340531.3411903

[15] Liu, Y., Ao, Z., Qin, Z., Chi, J., Feng, J., Yang, H., & He, Q. (2021). Pick and choose: A GNN-based imbalanced learning approach for fraud detection. In Proceedings of the Web Conference 2021 (pp. 3168–3177). https://doi.org/10.1145/3442381.3449989

[16] Huang, M., Liu, Z., Sun, Y., Zhu, L., Yu, P. S., & He, Q. (2022). AUC-oriented graph neural network for fraud detection. In Proceedings of the ACM Web Conference 2022 (pp. 1311–1321). https://doi.org/10.1145/3485447.3512178

[17] Lin, T.-Y., Goyal, P., Girshick, R., He, K., & Dollár, P. (2017). Focal loss for dense object detection. In Proceedings of the IEEE International Conference on Computer Vision (pp. 2980–2988). https://doi.org/10.1109/ICCV.2017.324

[18] He, H., & Garcia, E. A. (2009). Learning from imbalanced data. IEEE Transactions on Knowledge and Data Engineering, 21(9), 1263–1284. https://doi.org/10.1109/TKDE.2008.239

[19] Davis, J., & Goadrich, M. (2006). The relationship between precision-recall and ROC curves. In Proceedings of the 23rd International Conference on Machine Learning (pp. 233–240). https://doi.org/10.1145/1143844.1143874

[20] Lundberg, S. M., & Lee, S.-I. (2017). A unified approach to interpreting model predictions. In Advances in Neural Information Processing Systems 30 (pp. 4765–4774).

[21] Sundararajan, M., Taly, A., & Yan, Q. (2017). Axiomatic attribution for deep networks. In Proceedings of the 34th International Conference on Machine Learning (pp. 3319–3328).

[22] Ying, Z., Bourgeois, D., You, J., Zitnik, M., & Leskovec, J. (2019). GNNExplainer: Generating explanations for graph neural networks. In Advances in Neural Information Processing Systems 32.

[23] Luo, D., Cheng, W., Xu, D., Yu, W., Zong, B., Chen, H., & Zhang, X. (2020). Parameterized explainer for graph neural network. In Advances in Neural Information Processing Systems 33 (pp. 19620–19631).

[24] Yuan, H., Yu, H., Gui, S., & Ji, S. (2023). Explainability in graph neural networks: A taxonomic survey. IEEE Transactions on Pattern Analysis and Machine Intelligence, 45(5), 5782–5799. https://doi.org/10.1109/TPAMI.2022.3204236

[25] Ribeiro, M. T., Singh, S., & Guestrin, C. (2016). Why should I trust you?: Explaining the predictions of any classifier. In Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (pp. 1135–1144). https://doi.org/10.1145/2939672.2939778

[26] Zhou, T. (2024). Healthcare Provider Fraud Detection Analysis. Mendeley Data (Version 1). https://doi.org/10.17632/552r52kmgb.1

[27] Hong, B., Zhang, Y., Li, Y., Wang, Y., & Zhao, L. (2024). Health insurance fraud detection based on multi-channel heterogeneous graph structure learning. Heliyon, 10, e30045. https://doi.org/10.1016/j.heliyon.2024.e30045

[28] Chen, J.-P., Lu, P., Yang, F., Chen, R., & Lin, K. (2022). Medical insurance fraud detection using graph neural networks with spatio-temporal constraints. Journal of Network Intelligence, 7(2), 480–498.

[29] Lu, J., Wang, Y., Zhang, Y., Li, Y., & Zhao, L. (2023). Health insurance fraud detection by using an attributed heterogeneous information network with a hierarchical attention mechanism. BMC Medical Informatics and Decision Making, 23, 62. https://doi.org/10.1186/s12911-023-02152-0

[30] Zhang, J., Yang, F., Lin, K., & Lai, Y. (2022). Hierarchical multi-modal fusion on dynamic heterogeneous graph for health insurance fraud detection. In 2022 IEEE International Conference on Multimedia and Expo (ICME) (pp. 1–6).

[31] Hancock, J. T., Bauder, R. A., Wang, H., & Khoshgoftaar, T. M. (2023). Explainable machine learning models for Medicare fraud detection. Journal of Big Data, 10, 154. https://doi.org/10.1186/s40537-023-00821-5

[32] Chawla, N. V., Bowyer, K. W., Hall, L. O., & Kegelmeyer, W. P. (2002). SMOTE: Synthetic minority over-sampling technique. Journal of Artificial Intelligence Research, 16, 321–357. https://doi.org/10.1613/jair.953

[33] Jiao, Y., Fan, H., Yue, X., Ping, W., Sun, T., & Wang, J. (2026). Dynamic heterogeneous graph contrastive learning for uncovering collusive financial fraud. Scientific Reports, 16, 12456. https://doi.org/10.1038/s41598-026-58938-5

[34] Fan, H., Jiao, Y., Wang, M., Chen, J., & Yang, S. (2026). Fraud learns too: Continual graph learning under strategic adversarial drift in dynamic networks. Scientific Reports, 16, 14021. https://doi.org/10.1038/s41598-026-60997-7

[35] Ping, W., Jiao, Y., Fan, H., & Zhang, X. (2026). Multimodal fraud detection in financial statements: A trimodal attention network with contrastive evidence chain construction. IEEE Access, 14, 80456–80468. https://doi.org/10.1109/ACCESS.2026.3695466

[36] Chen, J., Liu, J., Liang, Y., & Zhou, M. (2026). KE-MLLM: A knowledge-enhanced multi-sensor learning framework for explainable fake review detection. Applied Sciences, 16, 2909. https://doi.org/10.3390/app16062909