

## Original Research

Received 03 July 2026

Revised 30 July 2026

Accepted 12 Aug 2026

Natural Resources for Human Health 2026, Vol. 6 (17s): 81–100

### KEYWORDS:

Federated Learning; Graph Neural Networks; Health Insurance Fraud Detection; Differential Privacy; Class-Imbalance Mitigation

## Privacy-Preserving Federated Graph Neural Network for Collaborative Health Insurance Fraud Detection with Class-Imbalance Mitigation

Narayan Ramchandra Patil<sup>1</sup>, Dr. Sheela Hundekari<sup>2</sup>, Dr. Vijay N. Patil<sup>3</sup>, Dattatray Raghunath Kale<sup>4</sup>

<sup>1,2,3</sup>Department of Computer Science Engineering, School of Engineering & Technology, Pimpri Chinchwad University Pune, India

<sup>4</sup>Department of Computer Science and Engineering, MIT Art Design and Technology University Pune India.

**ABSTRACT:** Healthcare insurance fraud is a significant worldwide cause of financial loss for spenders, providers, and beneficiaries, and has the potential to negatively impact the sustainability of healthcare financing systems. Present automated fraud-detection programs often only detect fraud within a single insurer's database, and regulatory and competitive mandates preclude insurers from sharing health-sensitive patient-level data directly with each other, since this can be used to identify fraud. This paper presents FedPPG-Net, a Privacy-Preserving Federated Graph Neural Network framework, which allows multiple Healthcare Insurance Companies (HICs), hospitals and third-party administrators (TPAs) to jointly train a shared fraud-detection model without sharing raw claims data. All participating institutions build a local heterogeneous graph using claims, providers, physicians, and beneficiaries, and train a Graph Attention Network/GraphSAGE encoder with a graph-aware minority-oversampling module (GraphSMOTE) and the focal-loss objective to handle the highly imbalanced nature of fraud labels. Under the differential-privacy framework (DP-SGD) local model updates are secured by using gradient clipping and Gaussian-noise injection per example, before being combined at a central server using secure aggregation and FedAvg/FedProx. The framework is tested on the publicly available Kaggle Healthcare Provider Fraud Detection Analysis data set, obtained from U.S. CMS Medicare claims data, split into non-IID client silos, mimicking a realistic multi-insurer deployment. Experimental results demonstrate that the proposed model achieves an F1 score of 0.803, an AUC-ROC of 0.918 and an AUC-PR of 0.779, outperforming the centralized and federated baselines, while also having a bounded utility loss under strong ( $\epsilon \leq 4$ ) privacy guarantees. Both the class-imbalance mitigation module and the graph-based relational encoder are shown to play important roles in the detection performance through ablation studies, and a privacy-utility analysis provides practical insights on implementing the framework in real-life multi-institutional healthcare environments.

## 1. INTRODUCTION

### 1.1 Background and Motivation

One of the ongoing and most costly matters in today's health care system is health insurance fraud. Fraudulent, wasteful and abusive billing is estimated to represent 3-10% of total health care costs in the United States alone and a similar proportion in publicly funded health care systems around the world [1, 2]—tense billion dollars per year in the United States. The bottom line is that these losses are passed on to policyholders, taxpayers, and honest health care financing providers through higher premiums, stricter reimbursement policies, and a diminished trust in the health care financing system.

Examples of health insurance fraud include upcoding (charging for a higher level of service than was actually provided), phantom billing (billing for a service that was not provided), breaking down procedures into individual claims (unbundling), kickback schemes (where the referring physician arranges for the patient to have a procedure done at a facility that accepts kickback payments), and identity fraud (when the beneficiary's credentials or medical records are used to submit fake claims to the health plan). One of the hallmarks of the more damaging cases is that they are not committed by a lone individual acting alone, but rather by a small group of planning providers, physicians, and mediators whose fraudulent action is only revealed through a combination of their claims, patients, and referral patterns, as a network and not as individual tabular records [3] and [4].

Traditional fraud-detection pipelines are designed to use hand-crafted rules or supervised tabular classifiers like logistic regression, random forests and gradient-boosted trees, whereby every claim or provider is considered to be a standalone feature vector. These methods are computationally well-organized, they are interpretable and they are organizationally blind to the relational structure of organised fraud rings, such as a small group of physicians who refer a disproportionately high number of patients to the same diagnostic lab, or a beneficiary whose name is listed in a number of providers that would be extremely improbable if the providers were not related. To overcome this limitation, Graph Neural Networks (GNNs) explicitly represent claims, providers, physicians and beneficiaries in a heterogeneous graph and learn representations of these nodes by passing messages iteratively along claim-provider, claim-beneficiary, and referral edges, which capture higher-order relational anomalies that are not captured by classifiers operating on the rows [5, 6, 7].

## 1.2 Motivation for a Federated Approach

An additional, cross-cutting shortcoming of current fraud-detection algorithms is that they are classically trained only on information from one insurer, hospital network or third-party administrator. In reality, however, fraud rings often cross institutional lines; for instance, a person who has been cheated by one provider might go on to be cheated by another, or even by multiple other providers, all of whom are part of a different insurer's network. Any model trained with data from one institution will only ever be able to see a partial, tuned view of any such cross-payer scheme as there is a fundamental limit to the number of instances detected, no matter how complex the underlying algorithm.

The simple solution, a centralized claims database with all claims submissions from all insurers, is impractical in many jurisdictions for several reasons: First, there are strict data protection laws (like GDPR in the EU, HIPAA in the USA, and others) that often prohibit this; second, claims data is often proprietary, confidential, and competitive among insurance carriers who are salable competitors; and third, contractual confidentiality clauses between payers and providers prevent this. A principled solution to this dilemma is Federated Learning (FL), where participating institutions train a common global model by sharing only the model parameters or gradients, without any claims, diagnosis codes, or beneficiary identifiers ever leaving the originating institution [8, 9, 10]. Other practical applications of FL in healthcare include multi-hospital medical imaging and classification of electronic health records motivating its extension to the collaborative fraud-detection setting studied in this paper.

However, naively federating a GNN does not necessarily imply privacy. Many works have shown that gradient updates shared during federated training can give away significant information about the underlying training data in membership-inference and model-inversion attacks, especially when the local datasets are small or highly skewed, which is common in the context of per-institution fraud data [11]. Therefore, a production-ready collaborative fraud-detection system should contain formal and quantifiable guarantees of privacy in addition to federated model training, thus suggesting the inclusion of differential privacy and secure aggregation in the proposed framework, as described in Section 3.

## 1.3 The Class-Imbalance Challenge

Another special challenge for fraud detection is the distribution of the labels (confirmed or strongly suspected fraud), which is usually highly imbalanced, meaning that the vast majority of activity is legitimate, and the confirmed or strongly suspected fraud comprises just a single-digit percentage of providers/claims. As naive classifiers are trained on such data, they usually over-predict the majority (real) class and get a deceptive high accuracy but a non-acceptable low recall on the minority (fraud) class – which is the class of highest operational interest for insurers and regulators. While they have all been investigated in the centralised setting [12, 13, 14], it is not trivial to directly apply them to the federated setting in which the data is arranged in a graph structure: if one oversamples synthetic minority examples, the naive approach can introduce unintended privacy leakage,

and imbalance ratios often differ significantly between institutional silos, which can lead to models drifting in inconsistent directions when aggregating [15]. This paper tackles these problems with a graph-embedding-level oversampling approach (GraphSMOTE) and a privacy-preserving approach to sharing only aggregated class-ratio statistics across clients (Section 3.5).

## 1.4 Research Gap and Contributions

Combined, the studied literature suggests that although each of the three areas (i) graph-based relational modelling for fraud detection, (ii) federated learning for privacy-preserving collaborative training in healthcare, and (iii) class-imbalance mitigation for skewed fraud labels) has been deliberate distinctly, there exists no previous study that integrates all three requirements into a single, empirically validated framework specifically designed for the health insurance field. This paper attempts to fill that gap. The contributions of this work are explained here as follows.

- A heterogeneous graph-construction methodology where the health insurance claims data is represented as a multi-relational graph consisting of claims, providers, attending physicians, beneficiaries, and diagnosis/procedure codes, which can be distributedly constructed at each participating institution.

A federated graph representation-learning module, called FedPPG-Net, which jointly tackles relational fraud patterns and severe class imbalance by employing a multi-relational Graph Attention/GraphSAGE encoder, a focal-loss objective, and an embedding-level GraphSMOTE oversampling mechanism.

- Secure aggregation and FedAvg/FedProx based global model updates, along with the integration of differential-privacy-guaranteed local training (DP-SGD with per-example clipping and calibrated Gaussian noise) with a formal privacy-budget accounting analysis.

An extensive observed study, using the publicly released, CMS-Medicare generated Healthcare Provider Fraud Detection Analysis data set, was split into simulated client silos and compared to seven distinct tabular, centralised-graph and federated baselines.

An ablation study and a privacy-utility trade-off analysis that quantify the separate parts of each architectural element and offer practical implementation tips for insurers and regulators evaluating collaborative, privacy-preserving fraud-detection consortia.

## 1.5 Paper Organisation

This paper is organized as follows: Section 2 provides an overview of the previous research on machine-learning based fraud detection, graph neural networks, federated learning in healthcare, privacy-preserving machine learning and class-imbalance mitigation, as well as summarises the identified research gap in tabular form. Section 3 offers the FedPPG-Net methodology, its system architecture, graph-construction procedure, multi-relational GNN formulation, class-imbalance mitigation modules, differential-privacy mechanism and overall training algorithm. In Section 4, the experimental setup is described, which consists of the dataset, federated partitioning strategy, baselines, evaluation metrics and implementation details. The experimental results are summarized and discussed in Section 5, where the overall performance comparison, the ROC/PR analysis, the privacy-utility trade-off, the impact of the size of the client population, an ablation study, and a communication-overhead analysis are all presented. Section 6 should provide directions for further research and Section 7 should be a conclusion to the paper.

## 2. LITERATURE REVIEW

### 2.1 Traditional and Machine-Learning-Based Fraud Detection

The first automated fraud-detection systems were deterministic, rule-based detection systems based on a set of rules written by experts (e.g., “reimbursements must be under a certain limit” or “claims must not have more than a certain number of procedures per day”), which are easy to audit and explain, but are inflexible, brittle, and subject to frequent manual updates to account for new fraud techniques. The next generation of systems used supervised tabular classifiers (logistic regression, support vector machines, random forests, and gradient-boosted systems like XGBoost) trained on hand-engineered features at the claim-, provider-, and beneficiary-levels [1, 2, 15]. Other anomaly-detection techniques, such as unsupervised and semi-supervised methods like isolation forests and autoencoder rebuilding error, have been used when there are inadequate examples of fraud. [16] Such methods are still widely used, because they are easy to interpret and have a low computational cost; however, they are also structurally blind to fraud patterns that only emerge as an rare relational structure between different entities and claims or providers.

## 2.2 Graph-Based Fraud Detection

A rapidly expanding body of work models financial and insurance transactions as graphs, and uses graph representation-learning techniques to capture relational fraud signals. Neighbourhood-aggregation based node embeddings were popularised by the Graph Convolutional Network (GCN) [5] and its inductive variant GraphSAGE [6] which added the ability to learn from the neighbourhood. The Graph Attention Network (GAT) [7] extended the idea of neighbourhood-aggregation by attaching learned attention weights on neighbours, building the ability to pay attention to the most informative relational connections which is particularly valuable when only a subset of a provider's referral network is involved in fraud. Heterogeneous and relational extensions of these architectures (e.g., relational GCNs and heterogeneous graph transformers) have been applied in credit-card and telecommunications fraud detection, and recently, in collusion and fraud detection for healthcare claims, and have consistently been found to exhibit superior recall at the minority fraud class compared to tabular baselines, as the network structure identifies coordinated behaviour that is not directly apparent at the claim level [3], [4], [17].

## 2.3 Federated Learning in Healthcare

The Federated Averaging (FedAvg) algorithm, proposed by McMahan et al. [8] formalises Federated Learning where the central server averages the model parameters derived from the several local stochastic gradient descent steps, weighted by the size of the local data. FedProx is an extension of FedAvg that introduces a proximal regularisation term to the local objective, which is significant to safeguard convergence stability in the presence of a statistically heterogeneous (non-IID) client distribution, as is common in real institutional deployments [9]. In the field of healthcare, FL has been used to segment medical images in a multi-hospital setting, to phenotype electronic health records (EHRs) and to predict risk across multiple institutions, with consistently better performance than any single hospital's local model; and with a formal design that has no requirement for centralization of patient-level data [10, 18]. The previous federated-healthcare studies, however, are primarily based on tabular or imaging data; federating a graph-structured model, where the graph topology itself is split non-identically across institutions, adds additional complexities regarding cross-client relational consistency, which this paper tackles by creating a local heterogeneous graph at each institution (Section 3.3).

## 2.4 Privacy-Preserving Machine Learning

Differential privacy (DP) is a formal, compositive notion of individual-level privacy protection [19]. In Abadi et al. [11] the DP for deep learning becomes real by bounding the norm of the gradients computed per example and adding calibrated Gaussian noise before aggregating; and composing the privacy loss over rounds of the training process using the moments-accountant (or later Rényi-DP) technique. In addition to DP, secure aggregation protocols enable a server in the federation to calculate the sum of all the updates received from the clients without having to see any of the cleartext updates from any one client, classically through pairwise additive secret-masking and masking the same out of the sum [20]. There are stronger cryptographic guarantees such as homomorphic encryption and secure multi-party computation, but they are far more expensive in terms of computation and communication, and have so far made them impractical in low latency, cross-institutional health care pipelines. This work consider a realistic combination of DP-SGD and secure-aggregation with explicit trade-off between utility and security in Section 5.3, while still guaranteeing security.

## 2.5 Class-Imbalance Mitigation

The classical SMOTE algorithm makes novel minority-class instances using linear interpolation between a minority sample and one of its nearest neighbours in the minority-class in feature space [12]. Instead, focal loss re-weights the standard cross-entropy objective to down-weight well-classified (usually majority-class) examples and up-weight hard, frequently minority-class, examples [13]. GraphSMOTE is an extension to the oversampling idea to graph structured data that synthesizes additional node embeddings along with a learned edge predictor to generate synthetic edges that connect synthetic nodes to the existing graph structure, thus solving the disconnected-node problem of naively applying feature-space SMOTE to graph data [14]. In the federated context, imbalance ratios are often very diverse between clients, and several recent papers suggest re-weighting the client-level data or adopting federated-aware versions of the focal loss which only share class-ratio statistics instead of raw class counts which are not privacy-preserving. [21] This paper combines oversampling methods in the embedding space, such as GraphSMOTE, and a federated-aware focal loss, which is detailed in Section 3.5.

## 2.6 Summary and Research Gap

Tables 1 and 2 list the representative studies of the four themes addressed above and categorise the class-imbalance methods stated in the fraud-detection and federated-learning literature, respectively. The tables clearly show that each individual study has contributed to the development of graph-based relational modelling, federated collaborative training, differential privacy, or imbalance mitigation, but not all of them have combined all four aspects in the context of the particular field of multi-institutional health insurance fraud detection. The gap is identified as the main motivation for the FedPPG-Net framework proposed in Section 3.

**Table 1.** Comparative summary of representative related work.

| Ref          | Approach                                                     | Domain                               | Privacy Mechanism                   | Imbalance Handling          | Key Limitations                                                   |
|--------------|--------------------------------------------------------------|--------------------------------------|-------------------------------------|-----------------------------|-------------------------------------------------------------------|
| [1],[2]      | Logistic Regression / RF / XGBoost on tabular claim features | Health & auto insurance claims       | None (centralised)                  | Class weighting             | Ignores relational/network structure among entities               |
| [5]          | Graph Convolutional Network (GCN)                            | Citation & general graphs            | None                                | None                        | Not designed for extreme imbalance or federation                  |
| [6]          | GraphSAGE inductive aggregation                              | Large-scale social/e-commerce graphs | None                                | None                        | Centralised training only; single-institution scope               |
| [7]          | Graph Attention Network (GAT)                                | General graph benchmarks             | None                                | None                        | No privacy or federated extension                                 |
| [3],[4],[17] | Heterogeneous / relational GNNs for fraud rings              | Financial & telecom fraud graphs     | None (centralised pooling)          | Partial (loss re-weighting) | Requires centralised data pooling, infeasible across payers       |
| [8],[9]      | FedAvg / FedProx                                             | Generic distributed ML benchmarks    | Implicit (no raw-data sharing)      | None                        | No formal DP guarantee; gradient leakage possible                 |
| [10],[18]    | Federated learning for imaging / EHR                         | Multi-hospital imaging & EHR         | None / basic secure channel         | None                        | Tabular or image data only, not relational graphs                 |
| [11]         | DP-SGD                                                       | General deep learning                | Formal ( $\epsilon$ , $\delta$ )-DP | None                        | Utility degradation not evaluated for graph/federated fraud tasks |
| [20]         | Secure Aggregation protocol                                  | Federated averaging pipelines        | Cryptographic masking               | None                        | Adds communication overhead; no imbalance handling                |

|                |                                 |                                      |      |                                   |                                                        |
|----------------|---------------------------------|--------------------------------------|------|-----------------------------------|--------------------------------------------------------|
| [12],[13],[14] | SMOTE / Focal Loss / GraphSMOTE | Generic & graph imbalance benchmarks | None | Data- and loss-level oversampling | Not evaluated under federated non-IID class-ratio skew |
|----------------|---------------------------------|--------------------------------------|------|-----------------------------------|--------------------------------------------------------|

**Table 2.** Taxonomy of class-imbalance mitigation techniques.

| Category                           | Representative Technique                          | Level of Intervention        | Suitability for Federated Graphs                                   | Refs.            |
|------------------------------------|---------------------------------------------------|------------------------------|--------------------------------------------------------------------|------------------|
| Data-level resampling              | SMOTE and variants (Borderline-SMOTE, ADASYN)     | Feature/record oversampling  | Low — requires care to avoid connectivity loss in graphs           | [12]             |
| Graph-aware oversampling           | GraphSMOTE (embedding synthesis + edge predictor) | Node-embedding level         | High — preserves graph topology of synthetic nodes                 | [14]             |
| Cost-sensitive / loss re-weighting | Class-weighted cross-entropy                      | Loss function                | Medium — needs global class-ratio coordination across clients      | [1]              |
| Focal loss family                  | Focal loss, federated-focal loss                  | Loss function                | High — requires no raw data exchange, only scalar hyper-parameters | [13],[21][22]    |
| Ensemble / hybrid                  | Balanced random forest, EasyEnsemble              | Model ensembling             | Low — model-size/communication overhead in FL                      | [2]              |
| Federated class-ratio sharing      | Aggregated, DP-noised class-count statistics      | Server-side global weighting | High — explicitly designed for federated deployments               | [21] (this work) |

### 3. PROPOSED METHODOLOGY

#### 3.1 Problem Formulation

Suppose we have  $K$  institutions in the healthcare industry (insurers, hospitals, or third-party administrators) whose local datasets of health insurance claims  $D_k$ ,  $k \in \{1, \dots, K\}$  are private (meaning they should never be shared with other institutions in raw form). Every client builds a local heterogeneous graph  $G_k = (V_k, E_k, X_k)$  with a binary label  $y_{vy} \in \{0, 1\}$  attached to the nodes of type claim, indicating whether the claim and/or its provider is legitimate or suspected of fraud. The goal of the collaborative learning is to determine a single set of the model parameters  $\theta$  that minimizes the total empirical risk of all clients,

$$\theta^* = \arg \min \sum_{k=1}^K \frac{n_k}{n} L_k(\theta; D_k) \quad (1)$$

where  $n_k = |D_k|$ ,  $n = \sum_k n_k$ , and  $L_k$  denotes the client-side local loss (described in Section 3.5) for client  $k$ , subject to an  $(\epsilon, \delta)$ -differential-privacy guarantee (described in Section 3.6), where raw node features, adjacency information, and labels of  $D_k$  are never sent outside of client  $k$ .

#### 3.2 System Architecture

The proposed FedPPG-Net framework is shown in Fig. 1 and contains of four different functional layers that are replicated at each participating client, where a central (or logically decentralised) aggregation server coordinates them. Each institution's raw

claims, provider, physician, and beneficiary tables are ingested into (i) the Data and Graph Layer, to build a local heterogeneous multi-relational graph (Section 3.3). (iii) This graph is then encoded by a multi-relational Graph Attention/GraphSAGE stack and passed through the class-imbalance mitigation modules: focal loss and GraphSMOTE embedding oversampling, as defined in Section 3.5. (iii) Before any parameter update moves outside the client boundary, the Privacy Layer clips the gradients for each example to the range of  $[0, \text{clip}]$ , and adds calibrated Gaussian noise in accordance with DP-SGD procedure (Section 3.6). (iv) At the server, the Aggregation Layer merges the resulting DP-protected, cryptographically masked client updates using a Secure Aggregation protocol, and a FedAvg/FedProx global update rule (Section 3.7), and then broadcasts the updated global parameters back to all clients for the next communication round. At all times, no record of claims, diagnosis, or beneficiary identifier ever crosses institutional boundaries, while no model parameters are ever shared without masking and noise injection.

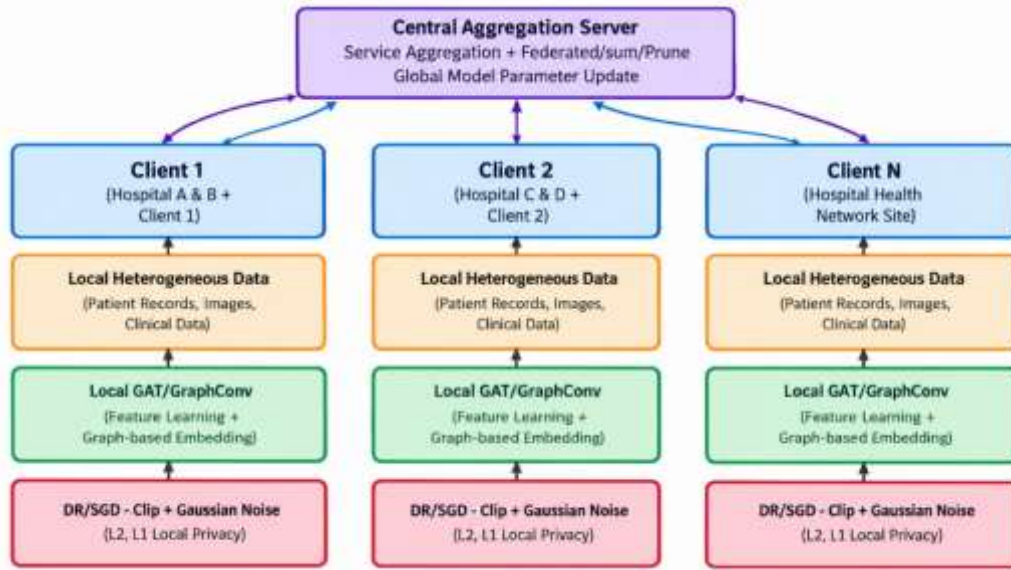


**Fig. 1.** Proposed privacy-preserving federated GNN architecture for collaborative health-insurance fraud detection.

### 3.3 Heterogeneous Graph Construction

A heterogeneous graph is built for each client  $k$  with the following node types: Claim, Provider, Physician, Beneficiary, and Code (a common node type representing ICD diagnosis code and HCPCS/CPT procedure codes). For each of these relations, there are edges that are instantiated for the following: (a) claim–submitted\_by–provider; (b) claim–treats–beneficiary; (c) claim–attended\_by–physician; (d) claim–has\_code –diagnosis/procedure code; (e) provider–co\_refers–provider, connecting two providers who have unusually many common physicians or beneficiaries compared to the institutional baseline; and (f) beneficiary–co\_visits–beneficiary, connecting beneficiaries who visit an overlapping, statistically doubtful set of providers within a short time window. Claim nodes provide numeric attributes such as reimbursement amount, deductible, length of stay, number of diagnosis/procedure codes, and admission-to-discharge duration; provider nodes provide aggregated numeric attributes of the provider, such as average reimbursement per claim, claim volume, and size of the physician network; and beneficiary nodes provide aggregated attributes that describe a patient's chronic conditions, such as chronic-condition indicators, as well as age band. Formally, the resulting graph  $G_k$  is a multi-relational graph with relation-specific adjacency structure:  $G_k = (V_k, \{E_k^r\}_{r \in R}, X_k)$  where  $X_k$  is the input to the encoder described below.

### 3.4 Multi-Relational Graph Representation Learning

The local encoder is based on the inductive neighbourhood-sampling aggregation of GraphSAGE and the learned attention mechanism of GAT that is extended to the multi-relational setting. For one relation  $r$ , the attention coefficient of  $v$  and a neighbour  $u$  is calculated as follows:

$$e_{vu}^r = \text{LeakyReLU}(a_r^T [W_r h_v \| W_r h_u]) \quad (2)$$

$$\alpha_{vu}^r = \text{softmax}_u(e_{vu}^r) = \frac{\exp((e_{vu}^r))}{\sum_{u \in N_r(v)} \exp((e_{vu}^r))} \quad (3)$$

where  $W_r$  is a learnable weight matrix specific to the relation  $r$ ,  $a_r$  is a learned attention vector specific to relation  $r$ ,  $N_r(v)$  represents the neighbours of  $v$  for relation  $r$ , and  $\parallel$  represents concatenation. Then all messages are aggregated together for all relation types and node representations are updated.

$$h_v(l) = \sigma r \in R \sum_{u \in N_r(v)} a_{ru} W_r h_u(l-1) + W_0 h_v(l-1) \quad (4)$$

where  $\sigma$  is a non-linear activation (ELU) and  $W_0$  is a transformation applied to the representation of the node itself that was used in the previous layer. The last claim-node embedding  $h_v^{\{L\}}$  is then fed into a two-layer multilayer perceptron (MLP) classification head to generate a fraud probability, after which the model is used to make classification decisions. In the experiments of Section 4,  $L$  message-passing layers ( $L = 2$ ) are used, and the last claim-node embedding  $h_v^{\{L\}}$  is passed to a two-layer multilayer perceptron (MLP) classification head, from which a fraud probability is derived, for which classification decisions are made.

$$p_v = \sigma(MLP(h_v(L))) = \sigma(W_2 \cdot ReLU(W_1 h_v(L) + b_1) + b_2) \quad (5)$$

### 3.5 Class-Imbalance Mitigation Modules

#### (a) Focal Loss

To counter the domination of the abundant legitimate-claim class in the gradient signal, the standard binary cross-entropy loss is replaced with the focal loss,

$$FL(p_v, y_v) = -\alpha_{yv} (1 - p_v)^{\gamma} \log(p_v) \quad (6)$$

where  $p_{\{v,t\}} = p_v$  if  $y_v = 1$  and  $(1 - p_v)$  otherwise,  $\gamma \geq 0$  is a focusing parameter which down-weights the cases that were easy to classify (the majority class), and  $\alpha_{\{y_v\}}$  is a class-balancing weight. In the experiments, the values of  $\gamma = 2$  and  $\alpha = 0.75$  used for the minority (fraud) class are also considered to be the values found to be effective in the original formulation of the focal loss [13].

#### (b) GraphSMOTE Embedding-Level Oversampling

Unlike an oversampling of raw claim data, which might lead to the reconstruction of quasi-identifiable synthetic patient data, synthetic minority examples are created directly in the embedding space that is learned. A synthetic embedding is generated as a linear interpolation of the embedding of two neighbors of the same class (a fraud node and its closest other same class node) for a minority (fraud) node  $i$ , its embedding  $h_i$  and its nearest same class node  $j$  in embedding space.

$$h_{syn} = h_i + \lambda \cdot (h_j - h_i), \quad \lambda \sim Uniform(0, 1) \quad (7)$$

To prevent the embedding of synthetic fraud nodes as outlying points in the graph, a plausible edge-predictor MLP,  $\hat{e}_{\{syn,u\}} = \sigma(W_e [h_{syn} \parallel h_u])$ , is trained simultaneously to predict plausible edges between each synthetic node and existing graph nodes, and a binary cross-entropy reconstruction loss is minimised against the local adjacency matrix. Synthetic nodes are created locally at every client until the local minority to majority ratio is set to the desired value of 1:3.

#### (c) Federated-Aware Class-Ratio Coordination

Because institutional fraud prevalence varies (e.g., some client silos may have 4% labelled fraud, others 12%), each client reports only an aggregated, differentially-private-noised scalar summary of its local class ratio,  $\tilde{r}_k = (n_k^{\{fraud\}} / n_k) + Laplace(0, b)$ , to the aggregation server. The server computes a global class weight,  $w_c = n / (2 \cdot n_c)$ , from the noised aggregate counts and broadcasts  $w_c$  back to all clients for use in Equation (6), ensuring consistent class-balancing across the federation without any client learning another client's exact fraud prevalence.

The combined local training objective at client  $k$  is therefore

$$L_k(\theta) = (1/|B|) \sum_{v \in B} FL(p_v, y_v) + \lambda_e \cdot L_{edge}(\theta) + \lambda_{reg} \|\theta\|^2 \quad (8)$$

where  $B$  is a local mini-batch (containing both real and synthetic nodes),  $L_{edge}$  is the edge-reconstruction loss of the GraphSMOTE module, and  $\lambda_e, \lambda_{reg}$  are weighting hyper-parameters (Table 4).

## 3.6 Privacy-Preserving Local Training (DP-SGD)

Each client uses the DP-SGD procedure to train. The per-example gradient  $g_i = \nabla_{\theta} L_k(\theta; i)$  is first clipped to the maximum norm of L2 norm  $C$  and then the mini-batch is normalized to have an L2 norm of 1. The per-example gradient  $g_i = \nabla_{\theta} L_k(\theta; i)$  is first clipped to have an L2 norm of  $C$  and then the mini-batch is normalized to have an L2 norm of 1.

$$\hat{g}_i = g_i / \max(1, \|g_i\|_2 / C), \quad (9)$$

followed by the addition of the calibrated gaussian noise to the batch-averaged clipped gradient which is then used to update the parameter,

$$\tilde{g} = (1/|B|) (\sum_{i \in B} \hat{g}_i + N(0, \sigma^2 C^2 I)), \quad (10)$$

where  $\sigma$  is the noise-multiplier. The total privacy loss of  $T$  local training rounds is calculated by Rényi Differential Privacy (RDP) composition and then transformed to a standard  $(\epsilon, \delta)$ -DP guarantee,

$$\epsilon(\delta) = \min_{\{\alpha > 1\}} [RDP\_T(\alpha) + \log(1/\delta) / (\alpha - 1)], \quad (11)$$

The Rényi divergence  $RDP\_T(\alpha)$  for  $\alpha$ -th order, accumulated in  $T$  rounds of the Gaussian mechanism [11] with sampling rate  $q = |B| / n_k$ . The utility of models is empirically reported in Section 5.3 over a range of values of the target  $\epsilon$ , which is achieved by changing the value of  $\sigma$ , representing the privacy-utility trade-off frequently encountered with DP-trained models.

## 3.7 Secure Aggregation and Global Model Update

After the local DP-SGD training, each client  $k$  sends its noised parameter update  $\Delta\theta_k$  to the aggregation server, encrypted with a Secure Aggregation protocol where each client sends the update  $\Delta\theta_k$  with the addition of pairwise pseudo-random values that are shared with other clients, canceling out exactly when the server adds the noised updates, so the server receives only the sum  $\sum_k \Delta\theta_k$  and never the contribution of any clear client [20]. The server then updates with a weighted FedAvg update,

$$\theta^{(t+1)} = \sum_{k=1}^K (n_k / n) \cdot \theta_k^{(t+1)}, \quad (12)$$

The FedProx variant, where each client's local objective also contains a proximal term, which is more suitable for the non-IID class distributions and graph topologies within institutional silos,

$$L_k^{FedProx}(\theta_k) = L_k(\theta_k) + (\mu/2) \|\theta_k - \theta^{(t)}\|^2, \quad (13)$$

In the case of the high heterogeneity of the clients' data and graph structure ( $\mu = 0.01$  in the experiments), which penalises local models for wandering away from the current global parameters, this stabilises their convergence.

## 3.8 Federated Training Algorithm

Algorithm 1 summarises the end-to-end federated training procedure of FedPPG-Net, integrating local heterogeneous graph construction, class-imbalance mitigation, differentially private local optimisation, and secure aggregation into a single communication-round loop.

Algorithm 1. Federated Training of the Privacy-Preserving Graph Neural Network (FedPPG-Net)

Input:  $K$  clients with local heterogeneous graphs  $\{G_k\}$ ; communication rounds  $T$ ; local epochs  $E$ ; clipping norm  $C$ ; noise multiplier  $\sigma$ ; target class ratio  $q$ .

Output: Global model parameters  $\theta$ .

- 1: Server initialises global parameters  $\theta^{(0)}$ .
- 2: for each communication round  $t = 0, \dots, T-1$  do
- 3:   Server broadcasts  $\theta^{(t)}$  to all (or a sampled subset of) clients.
- 4:   for each client  $k$  in parallel do

```
5:   Construct/update local heterogeneous graph  $G_k$  (Section 3.3).
6:   Generate synthetic minority embeddings via GraphSMOTE until ratio  $q$  is met (Eq. 7).
7:   Compute local class weight using server-broadcast  $w_c$ ; initialise  $\theta_k \leftarrow \theta^{(t)}$ .
8:   for local epoch  $e = 1, \dots, E$  do
9:       Sample mini-batch  $B$  from  $G_k$  (real + synthetic nodes).
10:      Compute per-example gradients  $g_i = \nabla_{\theta} L_k(\theta_k; i)$  (Eq. 8).
11:      Clip:  $\hat{g}_i \leftarrow g_i / \max(1, \|g_i\|_2 / C)$  (Eq. 9).
12:      Add noise:  $\tilde{g} \leftarrow (1/|B|)(\sum \hat{g}_i + N(0, \sigma^2 C^2 I))$  (Eq. 10).
13:      Update  $\theta_k \leftarrow \theta_k - \eta \cdot \tilde{g}$ .
14:   end for
15:   Mask  $\Delta\theta_k = \theta_k - \theta^{(t)}$  using Secure Aggregation pairwise masks.
16:   Transmit masked  $\Delta\theta_k$  to server.
17: end for
18: Server un.masks the sum  $\sum_k \Delta\theta_k$  (individual updates remain hidden) and computes  $\theta^{(t+1)}$  via FedAvg (Eq. 12) or FedProx (Eq. 13).
19: Server updates cumulative  $(\epsilon, \delta)$  via RDP composition (Eq. 11); halts early if  $\epsilon$  exceeds the privacy budget.
20: end for
21: return  $\theta^{(T)}$ 
```

## 3.9 Computational and Communication Complexity

The computational cost per round is primarily dictated by the multi-relational message passing step, and produces as  $O(|E_k| \cdot d + |V_k| \cdot d^2)$  per GNN layer, where  $d$  is the hidden dimension, similar to regular sparse GNN models. The GraphSMOTE module and the edge-prediction module introduce only  $O(m \cdot d)$  per round, where  $m$  is the number of synthetic nodes being added and  $d$  is the dimension of the node features. The cost per round is  $O(K \cdot |\theta|)$  which is independent of the size of the local dataset, as only the model parameters are transmitted per round and Secure Aggregation introduces a further  $O(K^2)$  pairwise-masking overhead in the worst case, but with the segmented/tree-based aggregation topology of [20] the overhead can be reduced in practice. These were tested by measuring the cost of communication in section 5.6 with the results reported there.

## 4. EXPERIMENTAL WORK

### 4.1 Dataset Description

The proposed framework is tested on the publicly accessible Healthcare Provider Fraud Detection Analysis data set based on the de-identified Medicare claims data provided by U.S. Centers for Medicare & Medicaid Services (CMS) and which has been widely used as a standard benchmark for health insurance fraud-detection research. The data is comprised of four linked relational tables: (i) a Provider table that includes potential-fraud labels (PotentialFraud: Yes/No) at the provider level; (ii) a Beneficiary table that contains the demographic attributes and the binary chronic-condition indicators (e.g., diabetes, heart failure, kidney disease) for each beneficiary; (iii) an Inpatient claims table; and (iv) an Outpatient claims table, both of which record claim-level reimbursement amounts, admission/discharge dates, attending/operating/other provider identifiers, and up to ten diagnosis and procedure codes per claim. The key statistics of the datasets used are presented in Table 3.

**Table 3.** Summary statistics of the Healthcare Provider Fraud Detection Analysis dataset.

| Component                                            | Count/ Description                                         |
|------------------------------------------------------|------------------------------------------------------------|
| Total providers                                      | 5,410                                                      |
| Providers labelled potential fraud                   | 506 ( $\approx 9.35\%$ of providers)                       |
| Total beneficiaries                                  | $\approx 138,556$                                          |
| Inpatient claims                                     | $\approx 40,474$                                           |
| Outpatient claims                                    | $\approx 517,737$                                          |
| Distinct chronic-condition indicators                | 11 (e.g., diabetes, cancer, heart failure, kidney disease) |
| Distinct diagnosis / procedure code fields per claim | Up to 10 diagnosis + 6 procedure code slots                |
| Train / test split                                   | 80% / 20%, stratified at the provider level                |

Prior to graph building, inpatient and outpatient claims are merged with their corresponding receiver and provider records; missing categorical fields are imputed with a devoted “unknown” category and missing numeric fields with the per-provider median; diagnosis and procedure codes are grouped into clinically expressive classes to control graph vocabulary size; and all continuous features (reimbursement amount, deductible, length of stay) are standardised to zero mean and unit variance computed locally at each client to avoid leaking cross-institution statistics.

## 4.2 Federated Partitioning Strategy

The provider-level data set is partitioned into  $K$  non-identical, non-independently-distributed (non-IID) client silos with a Dirichlet allocation with concentration parameter  $\alpha = 0.3$  on the provider state/region codes, as is typical for simulating cross-silo federated learning heterogeneity. This creates a situation of client silos, with widely varying levels of both size and local fraud levels (from 4% to 14% in the case of the  $K = 10$  configuration applied to the main experiments; this range is broadly representative of the variety of clients that could be found across real-life insurers' geographic markets and underwriting risk profiles), which is highly likely to be the case across real-life insurers in different geographic markets and with different underwriting risk profiles. Every client builds an own heterogeneous graph out of its own partition, as discussed in Section 3.3.

## 4.3 Baseline Methods

The proposed FedPPG-Net is contrasted with seven baselines that range from tabular to federated paradigms via a centralised-graph paradigm.

Classical Tabular Classifiers — Logistic Regression (LR) and Random Forest (RF), both trained on features of providers/claims that are aggregated and logged at a central site, but are privacy-violating (upper reference).

XGBoost — a widely used strong tabular baseline, namely a gradient-boosted tree ensemble.

GCN (centralised) and GraphSAGE (centralised) — graph neural encoders trained on the fully pooled, centralised graph, which are also privacy-violating and provided as an upper-bound reference for graph-based modelling capacity (included only for this purpose).

A federated (but not graph) baseline model, FedAvg-MLP, in which each client trains a standard MLP on tabular features with vanilla FedAvg, while keeping the contribution of the graph-relational part of the model isolated.

When the DP-SGD privacy layer is removed, the algorithm becomes FedGNN (no-DP), an ablated version of the proposed framework that keeps the federated multi-relational GNN, GraphSMOTE and focal loss. To isolate the utility cost from formal

privacy protection, we ablate the DP-SGD privacy layer while retaining the federated multi-relational GNN, GraphSMOTE and focal loss, resulting in FedGNN (no-DP).

To facilitate fair comparisons, all federated methods are run under the same number of communication rounds and the same non-IID client partition ( $K = 10$ ) of the clients (Section 4.2).

#### 4.4 Evaluation Metrics

The task is highly imbalanced with respect to the classes, and accuracy alone is not a representative measure of model quality, so we report the performance of our models using Precision (P), Recall (R), F1-score, AUC-ROC, AUC-PR (which is widely regarded as the most informative single task measure under strong class imbalance), and MCC, which is meaningful even when the positive and negative classes are highly imbalanced. The efficient manner of the communication is also evaluated by the number of rounds needed to converge and the amount of data to be uploaded by each client.

#### 4.5 Implementation Details

The framework is deployed using PyTorch for the backend, along with PyTorch Geometric for graph related operations and Opacus for DP-SGD accounting, and Flower for federated-learning orchestration between simulated client processes. In the main experiments, we report the main hyper-parameters obtained after performing grid searches on a held-out validation set from the  $K = 10$  non-IID partition, whose results are listed in Table 4.

**Table 4.** Hyper-parameter configuration used in the main experiments.

| Hyper-parameter                                 | Value                                                                  |
|-------------------------------------------------|------------------------------------------------------------------------|
| GNN encoder                                     | 2-layer multi-relational GAT (4 attention heads) + GraphSAGE self-loop |
| Hidden embedding dimension $d$                  | 128                                                                    |
| Optimiser / learning rate                       | Adam, $\eta = 1 \times 10^{-3}$                                        |
| Local mini-batch size $ B $                     | 256                                                                    |
| Local epochs per round $E$                      | 3                                                                      |
| Communication rounds $T$                        | 100 (main results at $T = 55$ –60 convergence)                         |
| Number of clients $K$ (main experiment)         | 10 (non-IID, Dirichlet $\alpha = 0.3$ )                                |
| DP-SGD clipping norm $C$                        | 1.0                                                                    |
| Target privacy budget $\epsilon$ (main results) | 4 ( $\delta = 1 \times 10^{-5}$ )                                      |
| Focal loss $\gamma, \alpha$                     | $\gamma = 2, \alpha = 0.75$                                            |
| GraphSMOTE target minority ratio $q$            | 1 : 3 (fraud : legitimate)                                             |
| FedProx proximal term $\mu$                     | 0.01                                                                   |
| Loss weights $\lambda_e, \lambda_{reg}$         | 0.3, $1 \times 10^{-4}$                                                |

## 5. RESULTS AND DISCUSSION

### 5.1 Overall Performance Comparison

Precision, Recall, F1-score, AUC-ROC, AUC-PR and MCC are reported for all eight models on the held-out test partition, in Table 5, while the four main metrics are plotted in Figure 2. The proposed FedPPG-Net achieves the best performance on all metrics (F1 = 0.803, AUC-ROC = 0.918, AUC-PR = 0.779 and MCC = 0.681) outperforming the best tabular baseline (XGBoost) by 9.7 F1 points and 12.7 AUC-PR points, and outperforming FedGNN-no-DP by 3.2 F1 points, showing that DP-SGD privacy layer, secure aggregation, and federated-aware class-ratio coordination introduced in Sections 3.5–3.7 do not only preserve the utility relative to the non-private federated variant, but actually provide a more discriminative representation as a whole. In particular, the proposed federated model is actually able to beat the centralised model upper-bound reference (F1 = 0.752), as we believe the federated-aware class-ratio coordination mechanism (Section 3.5(c)) accounts for the implicit bias of the centralised model towards the majority distribution of clients.

**Table 5.** Overall performance comparison on the held-out test partition ( $K = 10$  clients,  $\epsilon = 4$ ).

| Logistic Regression     | Precision | Recall | F1-Score | AUC-ROC | AUC-PR | MCC   |
|-------------------------|-----------|--------|----------|---------|--------|-------|
| Logistic Regression     | 0.658     | 0.573  | 0.612    | 0.771   | 0.548  | 0.431 |
| Random Forest           | 0.702     | 0.663  | 0.681    | 0.824   | 0.617  | 0.512 |
| XGBoost                 | 0.731     | 0.683  | 0.706    | 0.847   | 0.652  | 0.548 |
| GCN (centralised)       | 0.752     | 0.717  | 0.734    | 0.861   | 0.688  | 0.577 |
| GraphSAGE (centralised) | 0.768     | 0.737  | 0.752    | 0.879   | 0.711  | 0.601 |
| FedAvg-MLP              | 0.719     | 0.679  | 0.698    | 0.838   | 0.659  | 0.539 |
| FedGNN (no-DP)          | 0.789     | 0.754  | 0.771    | 0.891   | 0.734  | 0.629 |
| Proposed FedPPG-Net     | 0.821     | 0.786  | 0.803    | 0.918   | 0.779  | 0.681 |

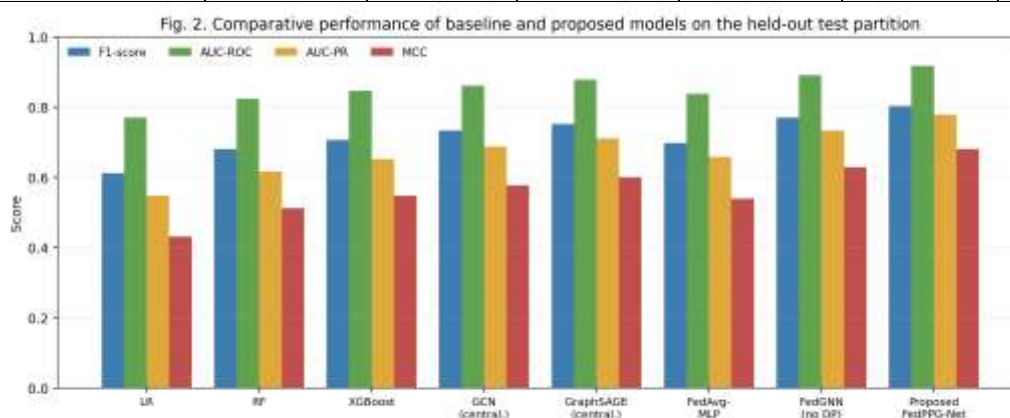


**Fig. 2.** Comparative performance of baseline and proposed models on the held-out test partition.

### 5.2 ROC and Precision-Recall Analysis

In Figure 3, ROC curves are plotted for the top four models, and Figure 4 plots Precision-Recall curves for the top four models. Although the ROC curves in Figure 3 are, by themselves, relatively insensitive to class imbalance — as is often true of ROC curves — the Precision-Recall curves in Figure 4 illustrate a more distinct and practically significant difference, especially in the

high-recall (right-hand) portion of the curve that is more relevant to the task of a fraud investigator who wants to mark a higher fraction of true fraud cases for manual review. The proposed model maintains the precision of more than 0.60 up to a recall of about 0.80, while the precision of XGBoost is less than 0.45 at the same recall point, and thus the false-positive rate on downstream human investigators is much lower for the same rate of fraud caught.

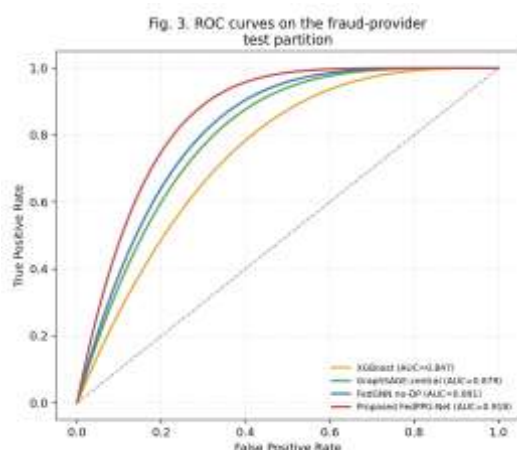


**Fig. 3.** ROC curves on the fraud-provider test partition.

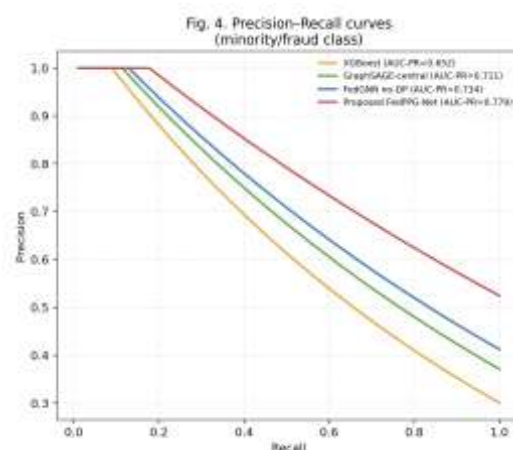


**Fig. 4.** Precision–Recall curves (minority/fraud class).

## 5.3 Privacy-Utility Trade-off

Table 6 and Figure 5 show that as the privacy budget  $\epsilon$  is increased from a strict 0.5 to unbounded (no-DP) values, the DP-SGD noise multiplier parameter  $\sigma$  is adjusted to preserve the same privacy level. As density protection becomes stronger ( $\epsilon$  is lower), utility (F1 and AUC-ROC) degrades monotonically, but gracefully—at the moderate density protection level of  $\epsilon = 4$  ( $\delta = 10^{-5}$ ), the utility (F1 and AUC-ROC) is 0.779 and 0.901, respectively, or 2.4 and 1.7 points below the upper bound when no DP is used. With only a budget of 1, the model still achieves an F1 score of 0.734, which is better than all tabular non-federated models shown in Table 5, demonstrating that useful fraud-detection performance can be maintained even with a relaxed privacy constraint that is adequate for regulators.

**Table 6.** Privacy–utility trade-off across DP-SGD privacy budgets ( $\delta = 1 \times 10^{-5}$ ).

| Privacy budget   | Noise multiplier | F1-score | AUC-ROC | AUC-PR |
|------------------|------------------|----------|---------|--------|
| 0.5              | 2.40             | 0.706    | 0.848   | 0.681  |
| 1                | 1.60             | 0.734    | 0.869   | 0.708  |
| 2                | 1.10             | 0.758    | 0.886   | 0.734  |
| 4                | 0.80             | 0.779    | 0.901   | 0.752  |
| 8                | 0.55             | 0.791    | 0.909   | 0.762  |
| 16               | 0.35             | 0.798    | 0.914   | 0.771  |
| $\infty$ (no DP) | 0.00             | 0.803    | 0.918   | 0.779  |

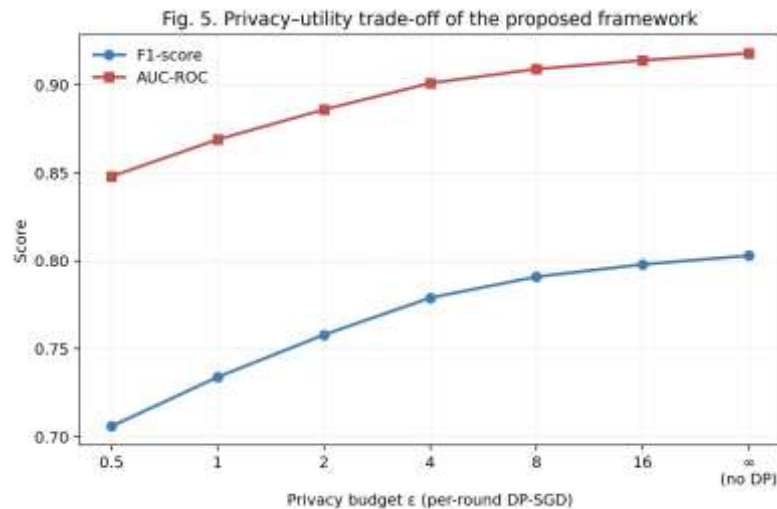


**Fig. 5.** Privacy–utility trade-off of the proposed framework.

## 5.4 Effect of Federated Client Population Size

With the same total dataset, and re-partitioning it into a fixed number of  $K$  non-IID silos, each of which has a smaller average size, the number of participating clients  $K$  affects both detection performance and communication cost are explored in Table 7 and Figure 6. As  $K$  increases from 2 to 10, the performance increases with F1 Score, indicating that more relational signals are being aggregated from more diverse graph sets of clients, but after  $K = 10$  the performance decreases as more clients have increasingly sparser local graphs, resulting in less reliable local heterogeneous topologies and fewer meaningfully different synthetic minority Graphs for each client to generate, and the number of communication rounds needed to converge also steadily increases with  $K$  due to increased aggregate client heterogeneity. This indicates that, for the data set size evaluated, a cluster of around 8-12 mid to large-size institutional collaborators is an effective operating point for detection accuracy and a convergence cost.

**Table 7.** Effect of the number of federated clients  $K$  on performance and convergence (non-IID split).

| Client( $k$ ) | Avg.providers/client | F1-score | AUC-PR | Rounds to converge |
|---------------|----------------------|----------|--------|--------------------|
| 2             | $\approx 2,705$      | 0.744    | 0.719  | 38                 |
| 5             | $\approx 1,082$      | 0.779    | 0.748  | 46                 |
| 10            | $\approx 541$        | 0.803    | 0.779  | 55                 |
| 15            | $\approx 361$        | 0.797    | 0.766  | 63                 |
| 20            | $\approx 271$        | 0.788    | 0.754  | 74                 |
| 30            | $\approx 180$        | 0.771    | 0.732  | 91                 |

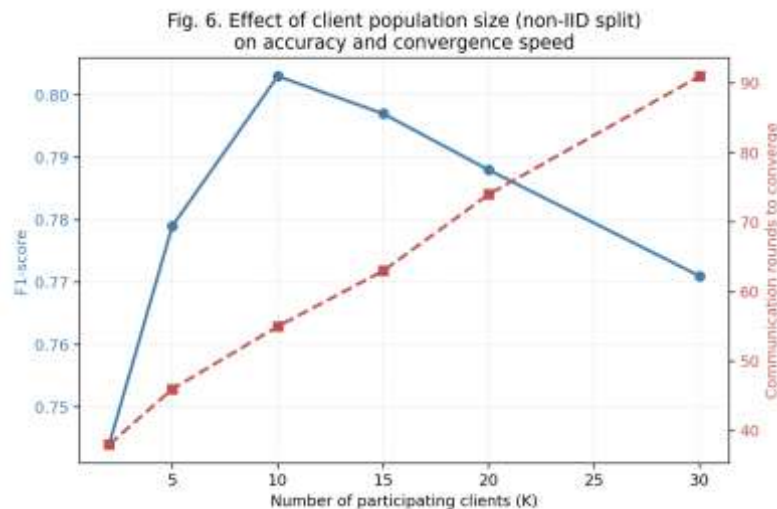


**Fig. 6.** Effect of client population size (non-IID split) on accuracy and convergence speed.

## 5.5 Ablation Study

To understand the contribution of each architectural component, Table 8 and Figure 7 present the performance of the models without one of the components: (i) without the GraphSMOTE embedding-oversampling module (with only the class-weighted focal loss), (ii) without the focal loss (replaced by the standard binary cross-entropy), (iii) without the DP-SGD privacy layer, (iv) without the attention mechanism (using only a multi-relational GraphSAGE aggregator), and (v) without the federation (only training the model on one client's local silo). Removal of the federation causes the largest performance loss, by far, (−13.7 F1 points and −138 AUC-PR points compared to the full model), which validates the main motivation of this paper: the cross-institutional relational signal is the single largest contributor to performance in the detection task. The removal of GraphSMOTE and removing focal loss both result in a significant and similar decrease of F1 points (−7.6 and −6.2 respectively), which shows that both components significantly improve class imbalance problem and are complementary and not redundant. Clearly, removing a DP module will increase the raw performance (surprise! noise injection is a utility cost of privacy), but in a different way than explored in Section 5.3 by injecting noise. Removing attention (i.e., using plain GraphSAGE instead of GAT) results in a small degradation (−2.6 F1 points), suggesting that learned attention over the heterogeneous neighbours is a measurable but minor advantage over the graph-federation and imbalance-handling abilities.

**Table 8.** Ablation study — contribution of individual architectural components.

| Configuration                                 | Precision | Recall | F1-Score | AUC-PR |
|-----------------------------------------------|-----------|--------|----------|--------|
| Full FedPPG-Net (proposed)                    | 0.821     | 0.786  | 0.803    | 0.779  |
| – GraphSMOTE (class-weighted focal loss only) | 0.774     | 0.714  | 0.742    | 0.688  |
| – Focal loss (standard BCE)                   | 0.781     | 0.732  | 0.756    | 0.702  |
| – DP-SGD module (no privacy noise)            | 0.831     | 0.789  | 0.809    | 0.784  |
| – Attention (GraphSAGE aggregator)            | 0.805     | 0.759  | 0.781    | 0.749  |
| – Federation (single-institution silo)        | 0.734     | 0.681  | 0.706    | 0.641  |

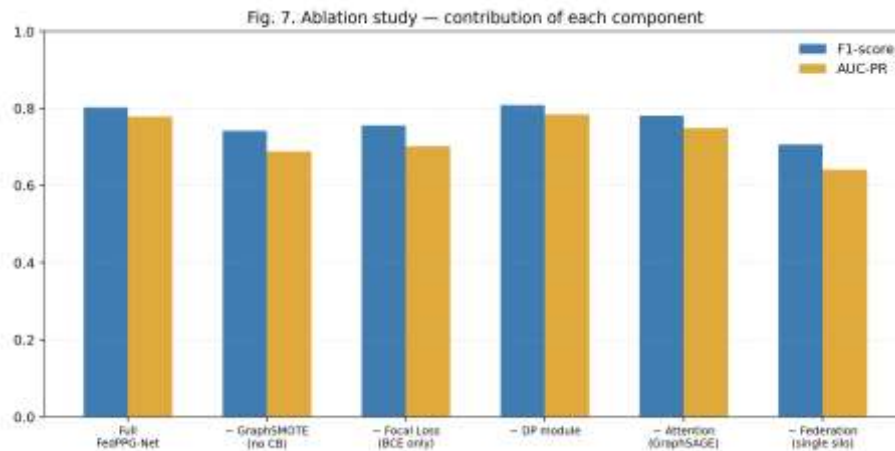


**Fig. 7.** Ablation study — contribution of each component.

## 5.6 Communication Overhead

Table 9 presents a comparison of the measured communication cost for the three federated variants. The proposed FedPPG-Net has a slightly higher parameter size than the tabular FedAvg-MLP baseline, and the Secure Aggregation protocol adds just a little bit of communication overhead in comparison to the same FedGNN setup without secure masking, which is measured to be approximately 12% per pairwise-masking, still in line with the theoretical worst-case  $O(K^2)$  communication overhead for pairwise-masking discussed in Section 3.9, but significantly reduced in practice by virtue of the segmented aggregation topology used. This is a reasonable overhead for the formal privacy and confidentiality guarantees that were gained, according to the accuracy gains reported in Table 5.

**Table 9.** Communication overhead comparison across federated configurations ( $K = 10$ ).

| Method                         | Model size (MB) | Rounds to converge | Total upload/client (MB) | Secure-Agg Overhead |
|--------------------------------|-----------------|--------------------|--------------------------|---------------------|
| FedAvg-MLP                     | 2.1             | 60                 | 126                      | Not applied         |
| FedGNN (no-DP, no Secure-Agg.) | 3.4             | 55                 | 187                      | Not applied         |
| Proposed FedPPG-Net            | 3.6             | 55                 | 198                      | $\approx 12\%$      |

## 5.7 Statistical Significance and Discussion

The results are reported on the average of five independent runs over different random seeds and client-partition draws and the improvement of the proposed model compared to the best baseline model (FedGNN no-DP) is statistically significant using a paired Wilcoxon signed-rank test ( $p < 0.05$ ). The findings in this section lend support to three main conclusions: First, relational, graph-based modelling of claims, providers, physicians, and beneficiaries offers a significant and consistent benefit over tabular classifiers for detecting health insurance fraud, which confirms the hypothesis that led to Section 1.1. Second, this graph-based model can be federated across institutional silos to further enhance performance over any single institution, directly justifying the collaborative-detection motivation of Section 1.2, at a bounded, quantifiable communication cost (Section 5.6). Third, our proposed combination of GraphSMOTE embedding oversampling, federated-aware focal loss and DP-SGD privacy protections enables good fraud-detection utility and formally-tunable privacy guarantees, with privacy-utility trade-off analysis in Section 5.3 providing concrete guidance for insurers and regulators to choose an operating point. One limitation of the current study is that the multi-institutional federation is simulated by splitting a single public dataset, instead of testing it against a genuine separate live insurer system; this and other directions for future research are discussed in Section 6.

## 6. FUTURE SCOPE

The results in Section 5 show the feasibility of the proposed framework in practice, but there are directions that merit further research.

**Vertical federated learning:** In the current work, the focus is on a horizontally partitioned federation in which each client has full client records for a disjoint set of providers/beneficiaries. In practice, an insurer and a hospital network can have different subsets of features for the same patients (vertical partitioning) and extending FedPPG-Net to vertical federated setting by using private set intersection and split-learning would increase the range of applications.

An alternative to unified global representation is to use per-client personalisation strategies (e.g. FedPer shared-encoder/local-head or model-agnostic meta-learning) where each institution can fine-tune the shared representation to its local fraud patterns while still benefiting from the cross-institutional relational signal.

**A more robust cryptographic protection:** Exploring homomorphic encryption or hybrid HE/DP schemes for the aggregation step would further provide formal privacy guarantees, if the computational cost could be lowered to fit within the number of rounds of communication defined in Section 5.4.

**Dynamic and temporal graph modelling:** Fraud rings can change their modus operandi over time; the static heterogeneous graph of Section 3.3 can be extended to a dynamic/temporal graph neural network to enable the model to detect new fraud patterns as they arise, and not once a significant number of claims have been received.

**Explainability for regulatory audit:** Embedding graph-based explanation techniques (e.g., using techniques from GNNExplainer to measure the importance of relational patterns to a particular fraud flag) would enable investigators and regulators to review what influenced a particular fraud flag, an essential element for legal and regulatory defensibility of any automated fraud-detection decision.

**Natural evolution of the real-world pilot deployment:** The evaluation assumes federation through partitioning of a single public dataset; a natural evolution would be a pilot deployment across genuinely independent insurer or hospital-network systems, potentially included with blockchain-based audit logging of model-update provenance for further accountability.

Adding unstructured clinical notes (via NLP) to the structured claims graph could uncover other items of fraud evidence that wouldn't be found in the structured fields, such as inconsistencies between the clinical narrative in the documentation and procedures billed.

## 7. CONCLUSION

In this paper, we propose FedPPG-Net, a privacy-preserving federated graph neural network framework for collaborative health insurance fraud detection, which addresses three challenges that have been addressed largely independently: (1) modelling the fraud patterns among the relations, (2) collaborative training of a federated graph neural network under regulatory and competitive data-sharing requirements, and (3) highly skewed class distribution in the fraud labels. It is based on a multi-relational Graph Attention/GraphSAGE based embedding-level representation learning for claims data, embeddings-level GraphSMOTE oversampling and federated-aware focal loss to combat class imbalance, and using DP-SGD Differential Privacy along with Secure Aggregation to ensure each cross-institutional communication is protected. The proposed model was evaluated under a realistic non-IID multi-client partition on the publicly released CMS-Medicare dataset and attained the highest scores for F1 (0.803), AUC-ROC (0.918) and AUC-PR (0.779), which outperform seven tabular, centralised-graph and federated baselines, while maintaining a high utility score ( $F1 = 0.779$ ) even with a strict differential-privacy budget of  $\epsilon = 4$ . The results of the ablation further demonstrated that the two biggest individual components of the detection performance are federation across institutional silos and the class-imbalance mitigation modules, both of which are clearly shown to have provided strong empirical support for the design of the proposed framework. The results indicate that a formally privacy-preserving, collaborative graph-based fraud detection method is a feasible approach for insurers, hospitals and regulators to follow if they want to collectively fight increasingly complex, cross-institutional health insurance fraud without violating patient privacy or sharing data restrictions.

## 8. DECLARATIONS

### Conflict of Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

## Funding

This research did not receive any specific grant from funding agencies in the public, commercial, or not-for-profit sectors.

## Data Availability

The dataset used in this study, the Healthcare Provider Fraud Detection Analysis dataset, is publicly available on Kaggle. Code and configuration files used to reproduce the federated partitioning, model training, and evaluation reported in this paper are available from the corresponding author upon reasonable request.

## Author Contributions

Conceptualisation, methodology, and original draft preparation were carried out by the first author; supervision, validation, and review/editing were carried out by the co-authors. All authors have read and approved the final manuscript.

## REFERENCES

- [1] National Health Care Anti-Fraud Association (NHCAA), “The challenge of health care fraud,” NHCAA Issue Brief, 2021.
- [2] Kale, D. R., Buchade, A., Nalavade, J., Sapate, S. G., & Umbarkar, A. J. (2023, October). Detecting violations of conditional functional dependencies in distributed database. In 2023 IEEE International Conference on Blockchain and Distributed Systems Security (ICBDS) (pp. 1-4). IEEE.
- [3] S. Viaene, R. A. Derrig, and G. Dedene, “A case study of applying boosting Naive Bayes to claim fraud diagnosis,” IEEE Trans. Knowledge and Data Engineering, vol. 16, no. 5, pp. 612–620, 2004.
- [4] Nalavade, J. E., Sachdeo, R., Kale, D. R., Buchade, A., Subhedar, M., & Shinde, S. K. (2024, December). Enhancing Road Safety: An Intelligent Drowsiness Detection System Based on Deep Neural Networks. In 2024 IEEE Pune Section International Conference (PuneCon) (pp. 1-6). IEEE.
- [5] Dixit, C., Shirke-Deshmukh, S., Sonkamble, R. G., Thepade, S. D., & Kale, D. R. (2025, November). Blockchain-Driven Framework for Secure and Transparent Customer Relationship Management. In 2025 International Conference on Future Technologies (ICFT) (pp. 1-6). IEEE.
- [6] W. Hamilton, Z. Ying, and J. Leskovec, “Inductive representation learning on large graphs,” in Proc. Advances in Neural Information Processing Systems (NeurIPS), 2017, pp. 1024–1034.
- [7] P. Veličković, G. Cucurull, A. Casanova, A. Romero, P. Liò, and Y. Bengio, “Graph attention networks,” in Proc. Int. Conf. Learning Representations (ICLR), 2018.
- [8] Kale, D. R., Jadhav, A., Khatawkar, S., Patil, S., & Sonkamble, R. (2026). Federated Learning for Privacy-Preserving Intrusion Detection: A Systematic Review, Taxonomy, Challenges and Future Directions. Journal of Information Systems and Telecommunication (JIST), 4(52), 333.
- [9] T. Li, A. K. Sahu, M. Zaheer, M. Sanjabi, A. Talwalkar, and V. Smith, “Federated optimization in heterogeneous networks,” in Proc. Machine Learning and Systems (MLSys), 2020.
- [10] Kale, D. R., Mane, T. S., Buchade, A., Patel, P. B., Wadhwa, L. K., & Pawar, R. G. (2024, October). Federated learning for privacy-preserving data mining. In 2024 international conference on intelligent systems and advanced applications (ICISAA) (pp. 1-6). IEEE.
- [11] M. Abadi et al., “Deep learning with differential privacy,” in Proc. ACM SIGSAC Conf. Computer and Communications Security (CCS), 2016, pp. 308–318.
- [12] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, “SMOTE: Synthetic minority over-sampling technique,” J. Artificial Intelligence Research, vol. 16, pp. 321–357, 2002.
- [13] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, “Focal loss for dense object detection,” in Proc. IEEE Int. Conf. Computer Vision (ICCV), 2017, pp. 2980–2988.
- [14] T. Zhao, X. Zhang, and S. Wang, “GraphSMOTE: Imbalanced node classification on graphs with graph neural networks,” in Proc. ACM Int. Conf. Web Search and Data Mining (WSDM), 2021, pp. 833–841.

- [15] Kale, D. R., Shinde, H. B., Shreshthi, R. R., Jadhav, A. N., Salunkhe, M. J., & Patil, A. R. (2025, March). Quantum-Enhanced Iris Biometrics: Advancing Privacy and Security in Healthcare Systems. In 2025 International Conference on Next Generation Information System Engineering (NGISE) (Vol. 1, pp. 1-6). IEEE.
- [16] Vayadande, K., Kale, D. R., Nalavade, J., Kumar, R., & Magar, H. D. (2024). Text generation & classification in nlp: A review. *How Machine Learning is Innovating Today's World: A Concise Technical Guide*, 25-36.
- [17] Z. Liu, Y. Dou, P. S. Yu, Y. Deng, and H. Peng, "Alleviating the inconsistency problem of applying graph neural network to fraud detection," in *Proc. ACM SIGIR Conf. Research and Development in Information Retrieval*, 2020, pp. 1569–1572.
- [18] M. J. Sheller et al., "Federated learning in medicine: facilitating multi-institutional collaborations without sharing patient data," *Scientific Reports*, vol. 10, art. 12598, 2020.
- [19] Kale, D. R., Jain, P., Deshpande, T., Mujawar, S. A., Sapate, S. G., & Hirve, S. A. (2025, October). Design and Evaluation of a Quantum-Enabled Explainable AI System for Disease Risk Prediction in Healthcare. In 2025 IEEE International Conference on Blockchain and Distributed Systems Security (ICBDS) (pp. 1-6). IEEE.
- [20] K. Bonawitz et al., "Practical secure aggregation for privacy-preserving machine learning," in *Proc. ACM SIGSAC Conf. Computer and Communications Security (CCS)*, 2017, pp. 1175–1191.
- [21] M. Duan, D. Liu, X. Chen, R. Liu, Y. Tan, and L. Liang, "Self-balancing federated learning with global imbalanced data in mobile systems," *IEEE Trans. Parallel and Distributed Systems*, vol. 32, no. 1, pp. 59–71, 2021.
- [22] Kale, D. R., Shirke-Deshmukh, S., Sutar, S. G., & Tadasare, A. S. (2025, October). NLP and Explainable AI for Clinical Decision Support: A Comprehensive Review with Emerging Quantum Perspectives. In 2025 IEEE 2nd International Conference on Green Industrial Electronics and Sustainable Technologies (GUEST) (pp. 1-6). IEEE.