IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 12 July 2023, accepted 9 August 2023, date of publication 17 August 2023, date of current version 23 August 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3305962

RESEARCH ARTICLE

Medicare Fraud Detection Using Graph Analysis: A Comparative Study of Machine Learning and Graph Neural Networks

YEEUN YOO[1], JINHO SHIN[1], AND SUNGHYON KYEONG[2]
[1]Division of Research and Development, KakaoBank, Seongnam-si 13529, Republic of Korea
[2]Division of Data Intelligence, KakaoBank, Seongnam-si 13529, Republic of Korea

Corresponding author: Sunghyon Kyeong (sunghyon.kyeong@gmail.com)

ABSTRACT Insurance companies have focused on medicare fraud detection to reduce financial losses and reputational harm because medicare fraud causes tens of billions of dollars in damage annually. This study demonstrates that medicare fraud detection can be significantly enhanced by introducing graph analysis with considering the relationships among medical providers, beneficiaries, and physicians. We use open-source tabular datasets containing beneficiary information, inpatient claims, outpatient claims, and indications about potential fraudulent providers. We then aggregated them into a single dataset by converting them into a graph structure. Furthermore, we developed medicare fraud detection models using two approaches to reflect graph information, i.e., graph neural network (GNN) models and traditional machine learning models using graph centrality measures. Therefore, the machine learning model with graph centrality features showed improved precision of 4 percent point (%p), recall of 24 %p, and F1-score of 14 %p compared to the best GNN model. The improvement in recall to this extent could result in substantial cost savings of 3.1 billion euros and 5 billion dollars in the United States and Europe, respectively, benefiting governmental institutions and insurance companies involved in healthcare insurance operations. Furthermore, the required learning time of the best GNN model was approximately 250–300 times more than that of the best machine-learning model. This outcome suggests that successful and efficient detection of medicare fraud can be achieved if graph centrality measures are used to capture the relationships among medical providers, physicians, and beneficiaries.

INDEX TERMS Graph neural network, graph centrality measure, machine learning, medicare fraud detection.

I. INTRODUCTION

The world is becoming connected complexly with the development of network-related technologies; thus, fraudulent cases that utilize new connections are emerging in various fields, such as social networks and finance. Because through fraud economic profits are illegally obtained, financial damage is inflicted on the counterparty. Therefore, judicial institutions and financial institutions, such as banks and insurance companies, are making great efforts to detect fraud before it occurs.

It is difficult to estimate the amount of fraud damage accurately, but it is considerable. For example, medicare fraud costs approximately 13 billion euros in Europe and 21–71 billion dollars in the United States annually [1]. The National Health Care Anti-Fraud Association estimates that medicare fraud costs billions of US dollars annually. According to [2], a conservative estimate of the medicare fraud amount is about 3% of total medicare spending. Additionally, some government and law enforcement agencies place a loss as high as 10% of annual healthcare expenses in the U.S., worth more than $300 billion [3].

Fraud may occur alone, but two or more collusions are also common; in this case, the amount of damage due to the fraud is tremendous [4]. Therefore, analyzing this collusion

The associate editor coordinating the review of this manuscript and approving it for publication was Giacomo Fiumara.

relationship is essential for improving fraud detection performance [5]. Graph analysis or network analysis is one of the methods used to capture collusion in medicare fraud. For example, a study utilized knowledge discovery in databases (KDD) methodologies to detect fraudulent medical providers using medical insurance data. In this study, network features, such as centrality measures from the network between healthcare providers, contribute to the prediction of medicare fraud [6]. In another study, after creating a graph composed of healthcare provider nodes from health insurance data, fraud detection was performed using similarities in medical procedures and drug prescriptions between fraudulent and non-fraudulent healthcare providers [7]. Graph-based features, such as community size, community density, and average dollar amount, were used from a heterogeneous graph consisting of medical providers, patients, and doctors as nodes [8].

Recently, with the development of machine learning algorithms, significant research has been conducted on graph neural networks (GNNs) that learn graph-structured datasets using artificial intelligence. Because GNNs can learn the information of neighboring nodes of graph-structured datasets directly, the GNN model demonstrates a high prediction performance in a graph-structured dataset [9], [10], [11]. Various GNN algorithms, such as the graph convolutional network (GCN), graph sample and aggregate (GraphSAGE), graph attention network (GAT), heterogeneous graph attention network (HAN), and heterogeneous graph transformer (HGT), have emerged for effectively considering the relationship between nodes.

In medical fraud, false diagnosis is commonly made through collusion between a patient and a doctor or hospital and it is used to receive medical insurance benefits illegally to be distributed among conspirators. However, despite these recent studies on fraud detection using GNN, few studies have attempted to detect medicare fraud through collusion among patients, doctors, and hospitals with GNN. According to a recent study, the GraphSAGE algorithm contributes to better medicare fraud detection performance than traditional machine learning methods [2]. However, the study did not perform experiments to investigate whether the incorporation of graph-related features, such as graph centrality, into traditional machine learning models would improve performance of the medicare fraud detection.

However, because the GNN directly learns the connected relationship between agents as nodes and edges, the computational burden is quite significant when learning with a large amount of data and a large number of agents. This computational burden is a significant challenge in practice because fraud-detection systems operate in real time. As an alternative, we propose a learning method that extracts graph information from a network among providers, physicians, and beneficiaries and uses this information as a feature of conventional machine learning algorithms, such as logistic regression, XGBoost, and multi-layer perceptron (MLP) [12]. Next, we examine whether the performance of medicare fraud detection improves by introducing the GNN model and adding features generated from graphs to conventional machine learning algorithms.

Thus, graph centrality measures extracted from the provider–beneficiary network appear to have strong discrimination power compared to centrality measures obtained from the provider–physician network. This result implies that the relationship between medical providers and beneficiaries is important for medicare fraud detection because patients rather than physicians play a central role in medicare fraud claims. In addition, compared to the HAN with heterogeneous mini-batch sampling, which demonstrates the best performance among various GNN models, the logistic regression model with graph centrality measures shows improved performance.

This study also found that the performance of machine learning models with graph information is better than that of GNN models. However, GNN has the advantage that each node can adaptively learn the information of its neighbor nodes in the graph structure. These results suggest that developing a machine learning model using node centrality measures can be more efficient and effective than using GNN algorithms for the medicare fraud detection task, given that the computational burden of the GNN is enormous.

II. RELATED WORKS

A. FRAUD DETECTION USING GRAPH ANALYSIS

Many studies have attempted to improve the performance of the classification task by considering the relationship among the objects included in the dataset. For example, in social networks, there are cases in which economic benefits are obtained by influencing other people's purchasing decisions through the creation of false reviews or accounts through collective collusion. To reflect this collusion relationship, they set reviewers, reviews, and stores as nodes and use a graph model to predict fraudulent reviews [13]. Another study detected fake user accounts based on a relationship graph of users. Specifically, for new accounts created within seven days and with fewer than 50 friend requests, the probability of a fake account is calculated based on the response to the friend request and the result of accepting or rejecting the request [14]. In addition, random walk-based methods have been proposed for detecting fake user accounts through random walks of a series of nodes, based on the assumption that fake user accounts need more hops on average than standard accounts in the social network graph [15], [16], [17], [18], [19], [20].

Graph analysis is also widely used in risk prediction, a classification task, in the financial area. For example, the performance of credit rating can be improved by extracting the centrality measure from the correlation network for companies that have borrowed money from financial institutions and using it as a variable in logistic regression [21]. In particular, the performance of the corporate credit rating model using statistical and machine learning methods can be improved by

additionally using graph analysis considering the transaction relationship between companies. In this case, a weighted customer score employing graph analysis and the centrality information of the graph are used [22]. These studies are similar to ours in that they use the centrality information of the graph as features of machine learning models. Apart from the credit risk field, there is a study analyzing the systemic risk of the commodity derivatives market by representing the correlation between future prices over time in a graph structure, to consider the yield relationship in the derivatives market [23]. For example, graph-based analysis is practical when analyzing data composed of a binding network between molecules, such as protein structure [24], [25].

B. FRAUD DETECTION USING GNN

Recently, studies on GNN, a model that can directly learn feature information about neighboring nodes from a graph-structured dataset, have emerged in various fields, such as credit rating, fraud detection, and anomaly detection, because GNNs improve classification performance on a graph dataset. For example, a credit rating model was developed by learning the relationship between institutions by applying GNN to a loan guarantee network composed of financial institutions in graph units [26]. For recommendation tasks, a GNN is used to configure users and recommended items as nodes and learn the relationship between them in a graph structure [27], [28], [29], [30], [31].

In particular, various studies on fraud detection using the GNN have appeared in the social networks and financial fields. First, to detect false views or fake accounts in social networks, one study measured the possibility of false views in sub-graph units after clustering the entire graph data with a GCN and a deep modularity network [32]. Using the GCN, they detect fraudulent reviews in the online application review system by considering the reviewers' texts, behaviors, and relationships [33]. Second, in finance, there are various types of fraud, such as insurance fraud, malicious accounts, transaction fraud, and Bitcoin fraud. One study presented a network learning method called node2vec to detect an organized insurance fraud group that commits fraudulent claims for Alibaba's return freight insurance [34]. To detect malicious accounts of Alipay that continuously send spam and monetary damage, a study proposed a heterogeneous GNN that can consider the behavior between accounts [35]. One study proposed a GAT using a heterogeneous graph that can consider transaction-level interactions to detect fraudulent transactions in e-commerce platforms [36]. As criminals use the anonymity of cryptocurrency to damage the financial system, a study exists that sets Bitcoin transactions and payment flows as nodes and edges to develop a GCN model to detect cryptocurrency-related fraudulent transactions [37].

From these previous studies, we can observe that GNN is used for fraud detection in various fields. It is vital to consider the relationship to detect fraud that is accurately caused by intricate interactions among various agents. Consequently, GNNs have become a potential method for fraud-detection tasks, detecting fraudulent nodes by aggregating neighbor information to consider several relations [35], [38], [39], [40], [41], [42], [43].

C. PREVIOUS LITERATURE ON VARIOUS GNN ALGORITHM

Various GNN algorithms depend on how the graph structure is learned. First, one of the most basic GNN models, the GCN, is motivated by a convolutional neural network [44]. The GCN propagates feature information from the neighboring nodes using graph convolutional layers. However, GCN requires the entire graph structure to use a spectral-based method. This limitation results in high memory consumption when the graph size increases [45]. Second, the GraphSAGE is a generalized model of the GCN and an inductive representation learning model on a large graph. GraphSAGE samples a fixed-size set of neighboring nodes and aggregates features from the nodes instead of using the entire neighborhood [46]. GraphSAGE outperforms the learning inductive for node classification for the benchmark graph dataset. Third, GAT introduces attention-based architecture to the graph-structured dataset. GAT aggregates the feature information of neighboring nodes to calculate the hidden representation of each node using the self-attention mechanism. As the self-attention mechanism demonstrates state-of-the-art performance in sequence-based tasks, GAT performs on several benchmark datasets in node-classification tasks [47]. The self-attention mechanism calculates the attention coefficients for the reference node, indicating the importance of each node from the reference node. The structural graph information is dropped, enabling inductive learning by calculating the attention coefficients for all nodes of the graph data [48].

However, because the aforementioned homogeneous GNN algorithms cannot consider multiple types of nodes and edges in heterogeneous graphs because they learn only one type of node and edge, GNN algorithms for heterogeneous graphs are proposed [49]. The HAN is a novel model that considers semantic information for a heterogeneous graph structure. The HAN learns the importance of different types of nodes using node- and semantic-level attention based on meta-paths. This model can generate node embeddings by aggregating features from meta path-based neighbors using hierarchical attention. In addition, inductive learning can be conducted because the model shares parameters for the entire graph-structured dataset using hierarchical attention and does not rely on the scale of the graph [50]. Second, a HGT is designed for heterogeneous graph structures with two or more types of nodes and edges. The HGT aggregates information from source nodes to obtain a contextualized representation for the target node t by constructing a HGT using node- and edge-type-dependent parameters. HGT can be decomposed into three components (heterogeneous mutual attention, heterogeneous message-passing, and target-specific aggregation) modeling heterogeneity. In addi-

tion, a heterogeneous mini-batch graph sampling algorithm (HGSampling) is used for efficient and scalable learning of web-scale graph data [51].

D. PERFORMANCE OF MEDICARE FRAUD DETECTION USING GRAPH ANALYSIS AND MACHINE LEARNING

Many studies have conducted in the field of medicare fraud detection using GNN approaches and machine learning to utilize graph information, allowing for the consideration of relationships within the dataset. Each approach has its advantages. Machine learning models have the advantages of fast training time and relatively small and simple models. GNN models have the advantage of directly incorporating object relationships. Utilizing graph information in machine learning models offers the advantages of easier application of graph information and compatibility with existing machine learning models. Table 1 summarize various approaches for medicare fraud detection.

Machine learning models are becoming popular in the classification task of tabular datasets such as medicare fraud detection owing to their enhanced prediction capabilities, and convolutional neural network (CNN) algorithms are often used on image data as well [52], [53], [54], [55]. For example, a previous study applied a support vector machine algorithm to an anomaly detection model for insurance claims processes [56]. In another study, a decision tree model was used to medicare fraud detection model to capture abnormal patterns of insurance claims and patient data [57]. Another study employed risk metrics and distance-based correlations to detect Medicare prescription fraud [58]. The MLP model was used to develop a fraud detection model for insurance claims in Chile [59]. Furthermore, numerous studies have demonstrated the impressive performance of neural networks for medical fraud detection [1], [60].

In addition, GNN models are used to consider the structural information about the relationship between a patient and a doctor or hospital. A previous study developed a heterogeneous graph by setting the medicare providers and beneficiaries as nodes and applied GraphSAGE algorithm for detecting medicare fraud [2]. In another study, the attributed heterogeneous information network model was used to construct the behavioral relationships between patients' multiple visits [61].

Furthermore, a few studies have enhanced the traditional machine-learning model by adding graph-theoretical features. One study utilized the association among provider, drug prescription, and healthcare common procedure coding systems (HCPCS) to develop a fraud detection model [7]. The cosine similarity between the medical providers was used to generate the graph features. The ratio of fraudulent medical providers in the neighboring nodes was determined and used as a feature of supervised learning. Additionally, graph centrality information obtained from a medicare provider-provider network was used to determine the provider's fraud [6]. These studies employed graph inference outcomes as input features for machine learning algorithms, such as logistic regression, random forest, and gradient boosting [12].

E. CONTRIBUTIONS OF OUR STUDY

We identified areas for methodology improvement while developing fraud-detection models from the literature review. First, there is little research on how effectively the GNN detects medicare fraud. However, our study is the first to empirically examine which GNN algorithm and batch sampling approach successfully detects medicare fraud. Second, many studies have rarely introduced graph centrality metrics as input features of machine learning algorithms to detect medicare fraud. In contrast, our study found that the medicare fraud detection was significantly improved when the centrality measure, a key indicator of graph analysis, was utilized as a feature of machine learning models. Third, our study sheds light on the most effective fraud-detection method by filling the gap between GNN, graph analysis, and machine-learning research. In this respect, our studycontributes significantly to companies in which improving fraud detection performance is essential.

III. MATERIALS AND METHODS

A. DESCRIPTION OF DATASETS

We conducted experiments using open-source tabular datasets of fraudulent healthcare providers provided from Kaggle's repository [62]. The dataset consists of four datasets on inpatient and outpatient claims, beneficiary information, and medicare provider information on whether the medicare provider is a potential fraudster. Fig 1 describes the detail structure of each dataset. The inpatient data provided information on claims against hospitalized patients, such as admission and discharge dates. The outpatient data contained claim information for patients who were not hospitalized. This dataset provides billing information about medicare service, such as the billing date and amount. The beneficiary data provided information about beneficiaries receiving medical services, such as health status and region. The provider data included an indication of whether the medicare provider was a potential scammer, which is used as a label for developing a medicare fraud detection model. These four datasets also have information such as the unique identifier of the medicare provider (ProviderID), the identifier of the beneficiary (BeneID), and the identifier of the claims (ClaimID).

We created a combined dataset for developing our proposed models by merging multiple datasets using key columns marked in red, as depicted in Fig 1. Firstly, we combined the inpatient and outpatient data, as they share the same attributes related to Medicare claims, using a union operation. Secondly, we merged the beneficiary data with the combined claim dataset, utilizing the BeneID column as the key, to gather beneficiary information. Finally, we integrated the Medicare provider data into the combined claim dataset, using the ProviderID column as the key. As a result of these

TABLE 1. Summary of different categories of medicare fraud detection.

Category	Paper	Methodology	Results	Pros	Cons
Machine Learning	Kumar et al. (2010) [56]	SVM	Better precision (hit rate) over existing approaches which is accurate enough to potentially result in over $15-25 million in savings for a typical insurer	* Short training time * Relatively simple and small model	* Deep Learning methods show better performance
	Shin et al. (2012) [57]	DT	The proposed model was largely consistent with the manual detection techniques currently used to identify potential abusers and automate abuse detection is flexible and easy to update.		
	Aral et al. (2012) [58]	Distance based on data-mining approach	A true positive rate of 77.4% and a false positive rate of 6% for the fraudulent medical prescriptions		
	Ortega et al. (2006) [59]	MLP	A detection rate of approximately 75 fraudulent and abusive cases per month, making the detection 6.6 months earlier than without the system		
	Mayaki et al. (2022) [1]	LSTM autoencoder	The proposed method and some state-of-the-art methods and AUC(PRC) is 0.745		
	Johnson et al. (2019) [60]	ROS-RUS (Random OverSampling - Random UnderSampling) + NN(Neural Networks)	The ROS and ROS–RUS perform significantly better than baseline and algorithm-level methods with average AUC scores of 0.8505 and 0.8509, while ROS–RUS maximizes efficiency with a 4× speedup in training time		
Graph Neural Network	Yoo et al. (2022) [2]	RL / GNN	The GraphSAGE model using graph-structured data outperformed the baseline model based on all the performance metrics	* It can improve performance considering the connection relationship	* Oversmoothing problem occurs by stacking multiple layers of graph neural networks * Diffculty configuring graph-structured dataset * Computational cost is too high
	Lu et al. (2023) [62]	GNN	This attention-based model can improve interpretable results to provide more insights about the health care fraud detection task and outcomes		
Machine Learning with graph information	Liu et al. (2016) [8]	graph-theoretical feature	The tools and algorithms have been able to improve user productivity and allow users to produce results that were difficult or time consuming to produce previously.	* Graph information can be applied more easily than GNN * Additional relationships can be considered in existing ML models.	* Diffculty configuring graph-structured dataset
	Branting et al. (2016) [7]	graph-theoretical feature	A combination of 11 features achieved an f-score of 0.919 and a ROC area of 0.960 in exclusion prediction.		
	Herland et al. (2018) [12]	ML + graph-theoretical feature	Our results show that the Combined dataset with the Logistic Regression (LR) learner yielded the best overall score at 0.816, closely followed by the Part B dataset with LR at 0.805.		

merging processes, our final dataset encompasses valuable information about claims against patients, beneficiaries, and providers. The final dataset contains 56 features, as described partially in Table 2. The data labeled as ''Final data'' refers to the inputs used for developing the fraud detection model in this studsy. It comprises a total of 556,703 rows (or samples), and among them, 212,232 rows are identified as medicare fraud, indicating almost no class imbalance.

B. TWO APPROACHES FOR FRAUD DETECTION MODEL DEVELOPMENT

To develop a medicare fraud detection model, we considered two different approaches. One is to develop a model using GNN algorithms, and the other is to develop a model using a traditional machine-learning approach with graph features. Fig 2 shows a schematic of the two approaches. First, for the GNN approach, we transformed the tabular

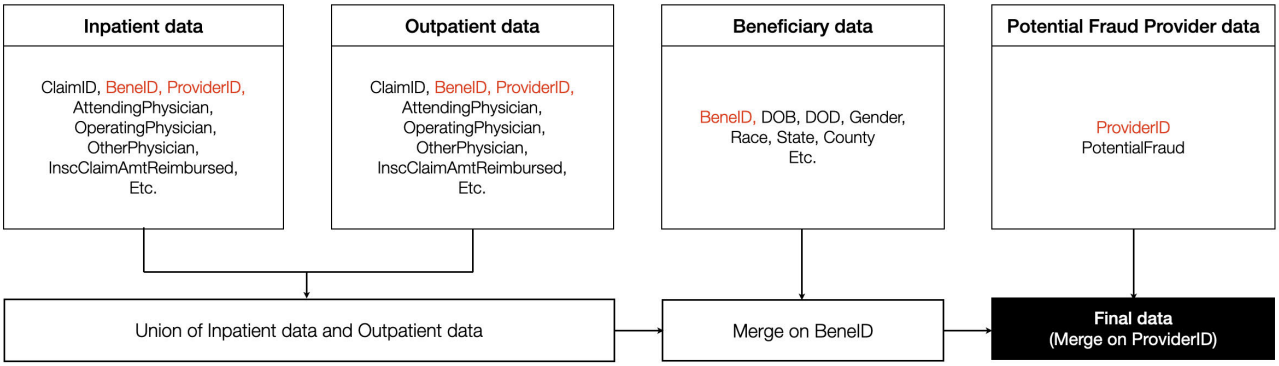

FIGURE 1. Dataset Merging Process. The variable names, highlighted in red, represent the key columns used for joining tables.

TABLE 2. Description of features in the final data.

Feature	Description
BeneID	Unique beneficiary identification number
ClaimID	Unique claim identification number
ClaimStartDt	The date on which the claim has been started
ClaimEndDt	The date on which the claim has been ended
Provider	Unique identification number of the healthcare provider
InscClaimAmtReimbursed	The amount of money reimbursed in an insurance claim
AttendingPhysician	Unique identification of the physician who takes the lead in overseeing the care of a patient
DOB	The date of birth of the beneficiary
…	…
PotentialFraud	Potential fraud labels

merged data (i.e., the final dataset in Fig 1) into graph-structured data. We then developed GNN models to detect fraudulent providers through a node classification task, as shown in panel A (see Section III-C). Second, concerning the machine learning approach, we extracted two types of relationships between Providers-Physician and Providers-Beneficiary from the merged dataset and then created two bipartite graphs using these relationships, as shown in panel B, to introduce graph centrality information as input features to the conventional machine learning algorithms (see Section III-D).

C. FRAUD DETECTION MODELS BASED ON GRAPH NEURAL NETWORKS

1) GRAPH CONSTRUCTION

When tabular data are provided, they must be converted into graph-structured data for training with the GNN. However, there is no unique way to represent graphs for any particular dataset. Therefore, transforming a graph database is a kind of modeling activity, and the best representation is to facilitate the algorithm of interest [7].

Fig 3 shows the graph-structured dataset in this study. We established the medicare providers and beneficiaries as nodes and the connections between medical providers, beneficiaries, and physicians as edges. The relationships between the medicare providers and beneficiaries were connected through the "CHARGE" edges. The relationships among medicare providers were connected through "PROJECT_PROVIDER" edges extracted from the bipartite graph projection between providers and physicians. In this study, the node classification models were constructed using this heterogeneous graph to predict fraudulent medical providers through GNN models.

2) GRAPH NEURAL NETWORK MODELS

GNNs have emerged as a powerful tool in fraud detection, revealing fraudulent nodes that have been identified by aggregating neighbor information through different relationships. Generally, a GNN learns through "message passing" which "aggregates" information from neighboring nodes and "updates" this information to the next layer. Suppose that $H_u^{(l)}$ is the node representation of node u at the (l)-th GNN layer. $H_u^{(l+1)}$ at the next $(l+1)$-th layer is updated based on (l)-th layer as follows:

$$H_u^{(l+1)} = Up^{(l)}\left(H_u^{(l)}, Agg^{(l)}\left(\left\{H_v^{(l)}, \forall_v \in N(u)\right\}\right)\right), \quad (1)$$

where $N(u)$ denotes neighboring nodes of node u, $Agg\cdot$ is an operator that collects information about neighboring nodes using aggregating operations such as mean, sum, max, and neural network. $Up\cdot$ is an operator that transforms the aggregated messages into the next hidden layer [63].

The GNN models considered in this study varied according to the updating framework in Equation (1). The GCN aggregates information regarding local nodes through a convolution layer and updates the normalized information regarding each node. GraphSAGE proposes a method of sampling a fixed size of neighboring nodes and various methods (average, sum, max, and RNN) of aggregating the information of sampled nodes [46]. GAT introduces an algorithm based

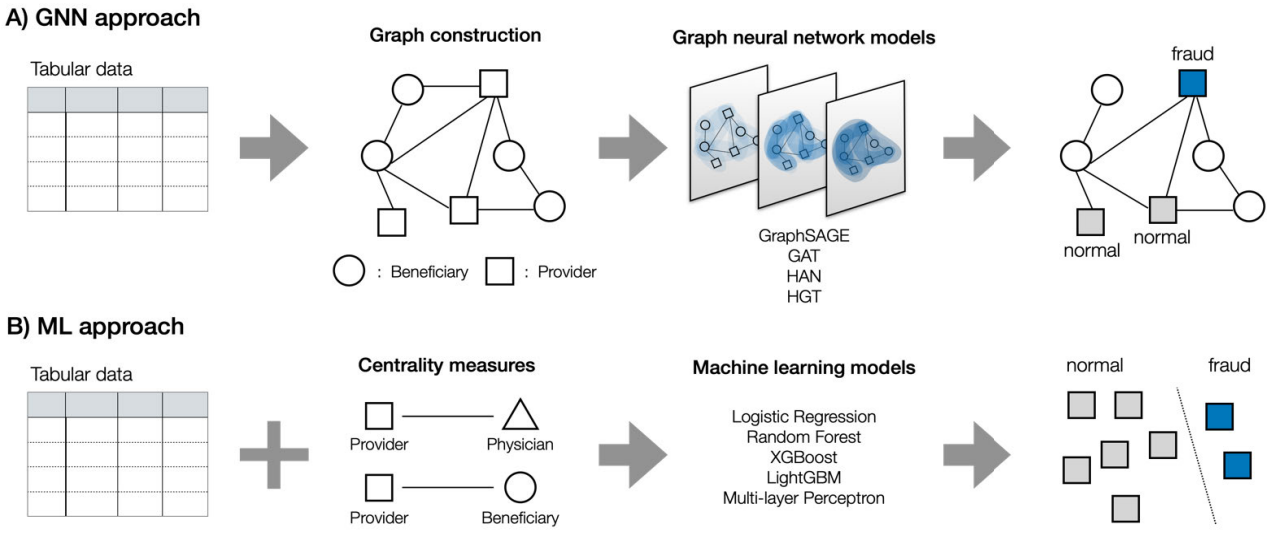

FIGURE 2. Visualization of two different model development approaches.

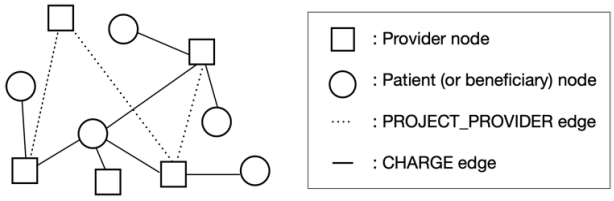

FIGURE 3. Graph data structure.

on the attention mechanism and calculates the importance of nodes through attention weights [48]. HAN performs node embedding using node-level and semantic-level attention based on a meta-path [50]. HGT calculates the weight parameters for different nodes and edges using heterogeneous mutual attention and message passing. In addition, HGT integrates information from the source nodes to the target nodes using target-specific aggregation [51].

In this study, when fraudulent medical providers are detected, a tabular dataset is constructed as a graph-structured dataset to consider the relationship between medical providers, physicians, and beneficiaries. In addition, we developed four GNN models (GraphSAGE, GAT, HAN, and HGT) to conduct node classification inductively.

3) DROPOUT AND MINI-BATCH TRAINING IN GNN MODELS

It is known that GNNs suffer from unstable learning and overfitting problems in node classification tasks with an increase in the model depth [64]. Overfitting occurs when we utilize an over-parametric model to fit a distribution with limited training data. The model we learn fits the training and validation data well but generalizes poorly to the testing data [65]. Dropout is widely adopted to address overfitting problems by reducing the co-adapting effect [66]. Therefore, we applied dropout to each model to address overfitting problems.

Learning a large graph dataset using a GNN requires considerable computational effort. To address these issues: Many studies have been conducted on learning graph datasets using mini-batch training [46], [51], [67], [68]. Mini-batch training refers to a method that uses a randomly selected subset of the training dataset to optimize the loss function. In addition, it enables stable learning and improves computational efficiency [69], [70], [71]. We used neighbor sampling and HGsampling for mini-batch training, which can be applied to heterogeneous graphs and inductive learning [46], [51].

D. FRAUD DETECTION MODELS BASED ON CONVENTIONAL MACHINE LEARNINGS

1) GRAPH CENTRALITY MEASURES

Once we know the structure of a graph, we can generate a variety of valuable measures for quantifying the centrality, level of interactions, and similarity related to other qualities of the graph structure. Centrality, which is graph-theoretical information, defines the level of importance of a node within a network. A large volume of network research is devoted to the concept of centrality [72].

In this study, we aim to extract the nodal centrality. In this study, we aim to extract the nodal centrality features from two bipartite graphs: the bipartite graph that obtains edge information from providers and physicians and the bipartite graph that obtains edge information from providers and beneficiaries. We compute the degree centrality, eigenvector centrality, closeness centrality, and PageRank for the centrality measure. Next, we add these centrality measures as the input features of conventional machine learning models.

Degree centrality is the most straightforward measure that counts the number of edges connected to a node. Degree centrality (C_d) is defined as follows:

$$C_d(v_i) = d_i \tag{2}$$

where v_i is the ith node, and d_i is the degree of *the* ith node.

Eigenvector centrality represents the centrality of a node by reflecting the importance of neighboring nodes through an adjacency matrix. Eigenvector centrality is proportional to the sum of the centralities of neighboring nodes; thus, the definition of the eigenvector centrality (C_e) of node v_i is as follows:

$$C_e(v_i) = \frac{1}{\lambda} \sum_{j=1}^{n} A_{j,i} C_e(v_j) (3) \tag{3}$$

where $A_{j,i}$ is the adjacency matrix, λ is an eigenvalue of the adjacency matrix. Note that the largest eigenvalue is selected using the Perron–Frobenius theorem, even though a matrix can have multiple eigenvalues [72].

Closeness centrality refers to the concept that the closer it is to the central node, the faster it can reach the other nodes. The closeness centrality becomes larger as the other nodes become closer because it is the inverse of the average shortest distances to the other nodes. Closeness centrality (C_c) is defined as follows:

$$C_c(v_i) = 1/\bar{l}_{vi} \tag{4}$$

where $\bar{l}_{vi}$ is the average shortest path length from node v_i to other nodes.

PageRank is a centrality measure that improves the limitations of Katz's centrality; when a node becomes central in a network, it passes its centrality to all its neighbor nodes. When calculating the influence of each node using PageRank, the number of outgoing edges is used as a normalization factor to prevent excessively high importance from spreading. PageRank (C_p) is defined as follows:

$$C_p(v_i) = \alpha \sum_{j=1}^{n} A_{j,i} \frac{C_p(v_j)}{d_j^{out}} + \beta \tag{5}$$

where α is a constant, β is the bias term that avoids the zero centrality value, and d_j^{out} is the number of outgoing edges (out-degree).

2) MACHINE LEARNING MODELS

We examine whether there is a performance improvement in medicare fraud detection by adding features generated from graphs to conventional machine learning algorithms. Unlike GNNs, which directly learn graph-structured datasets, machine-learning models use the existing tabular dataset and graph centrality features as additional independent variables. We computed graph centralities, such as degree centrality, eigenvector centrality, closeness centrality, and PageRank.

We developed logistic regression, random forest, XGBoost, LightGBM, and MLP as supervised-learning models to conduct classification tasks for Medicare fraud detection and compared their performances. We used these algorithms because they are the most widely used in financial fraud detection modeling [73], [74], [75].

E. TRAINING, VALIDATION, AND TEST DATASETS

As in [2], we divided deduplicated medicare providers into training, validation, and test datasets in a ratio of 6:2:2. We used these datasets to estimate model parameters, validate the fitted model, and evaluate the proposed model's performance, respectively. Table 3 presents the graph structure for each dataset. The numbers of samples, providers, beneficiaries, "CHARGE" edge, "PROJECT PROVIDER" edge, and frauds for each dataset are as follows: The training dataset includes 330,288 claims, 3,246 providers, 111,236 beneficiaries, 330,288 "CHARGE" edges, 3,486 "PROJECT PROVIDER" edges, and 125,864 frauds. The validation dataset includes 122,543 samples, 1,082 providers, 59,570 beneficiaries, 122,543 "CHARGE" edges, 342 "PROJECT PROVIDER" edges, and 52,680 frauds; test dataset includes 103,872 samples, 1,082 providers, 53,618 beneficiaries, 103,872 "CHARGE" edges, 308 "PROJECT PROVIDER" edges, and 33,688 frauds. We considered two heterogeneous node types, i.e., medicare provider nodes and beneficiary nodes in the development of GNN models. The "CHARGE" edge connected the provider node and the beneficiary node. If two or more providers are connected to the same physician, these providers are reflected in the "PROJECT_PROVIDER" edge of the graph.

F. PERFORMANCE MEASURES FOR THE MODELS

To compare the effectiveness of the various GNN and machine learning models, we evaluated fraud detection models using four performance metrics, such as precision, recall, F1-score, and area under the receiver-operating characteristic (AUROC) as in [2]. The following formulas were generally used to compute the performance of the models: precision = true positive / (true positive + false positive), recall = true positive / (true positive + false negative), F1-score = 2 / (1/precision + 1/recall), and AUROC = the area under the graph on two axes: false positive and true positive. Good model performance is demonstrated by higher precision and recall values [76]. The F1-score is defined as the harmonic mean of the precision and recall, and the closer to 1, the better the performance. We selected the best model in terms of the F1-score in this study.

IV. RESULTS

A. PERFORMANCE OF GNN MODELS

We developed various GNN models, such as GraphSAGE, GAT, HAN, and HGT, using training and validation datasets generated from our heterogeneous graph-structured data, and the performance of each model was compared and evaluated using the test dataset. Fig 4 reports the performance of both models for fraud detection. Note that all GNN models were trained ten times, and the average value of the performance metrics in the test dataset was calculated, and the best model was selected in terms of the F1-score, as shown in Fig 4. The best performing GNN model was HAN with heterogeneous mini-batch sampling, and recall, F1-score, and AUROC were

TABLE 3. Basic graph structure of each datasets.

	Training dataset	Validation dataset	Test dataset
Number of Provider Node	3,246	1,082	1,082
Number of Beneficiary Node	111,236	59,570	53,618
Number of "CHARGE" edge	330,288	122,543	103,872
Number of "PROJECT_PROVIDER" edge	3,486	342	308

the highest at 0.52, 0.51, and 0.74, respectively. In addition, for modeling without sampling, the HAN showed the highest recall, F1-score, and AUROC.

B. PERFORMANCE OF MACHINE LEARNING MODELS

1) LOGISTIC REGRESSION

We first selected a combination of independent variables with statistical significance for medicare fraud detection to develop a logistic regression model. To select significant independent variables, we used the forward selection method and selected variables based on the p-value (<0.05). As shown in Fig 5A, the baseline logistic regression model without node centrality features exhibited the lowest performance. The baseline model's F1-score of 0.19 is insufficient for fraud detection. However, the performance of the logistic regression model was significantly enhanced by considering the graph centrality features. In particular, we found that the graph centrality information of the provider-beneficiary network greatly influenced the effectiveness of fraud detection. Surprisingly, the performance of the logistic regression model increased by more than three times based on the F1-score than the baseline model. These results indicate that patients, rather than physicians, play a central role in medicare fraud.

Table 4 reports the estimated results of the baseline logistic regression without the node centrality features. If the sign of the estimation coefficient is positive, the variable increases the possibility of fraud, and vice versa if the sign is negative. First, nine variables positively correlate with the probability of a fraudulent medical provider. They are "State_encoded" (encoded variable about the states of the US), "Ip_indicator_encoded" (encoded variable about whether a physician was an inpatient or outpatient), "Race_encoded" (encoded variable about a patient's race), and "InscClaimAmtReimbursed" (variable about the amount reimbursed to the claimant). The others include "AllAnnualDeductibleAmt" (variable about the annual amount paid as insurance deductible), "ClmDiagnosisCode_6_encoded" (encoded variable about 6th claim diagnosis code), "NoOfMonths_PartACov" (variable about the number of months spent paying for coverage of PartA), "ChronicCond_stroke_encoded" (encoded variable about whether the patient has chronic stroke disease), and "ChronicCond_KidneyDisease_encoded" (encoded variable about whether the patient has chronic kidney disease). However, two variables negatively correlated with the probability of fraudulent medical providers. They are "ChronicCond_Depression_encoded" (encoded variable about whether the patient has chronic depression disease) and "ClmProcedureCode_4_encoded" (encoded variable about 4th claim procedure code).

The logistic regression results obtained by additionally using node centrality features from the provider-physician network are reported in Table 5. The variables were selected considering multicollinearity between them. Although various centrality features were considered, only the PageRank of provider nodes was significant because of their high multicollinearity. The PageRank of provider nodes positively correlated with the probability of fraudulent medical providers and had the most significant influence on prediction because of its largest Z-statistic. In addition, compared with the baseline model, the performance was improved in terms of the F1-score by 0.18 (Fig 5A).

Table 6 shows the logistic regression results with node centrality features obtained from the provider-beneficiary network to the independent variables of the baseline model. Considering the multicollinearity between the independent variables, we selected eight variables. In this case, only the PageRank of provider nodes was selected as a statistically significant variable among the various centrality features. The PageRank of provider nodes positively correlated with the probability of fraudulent medical providers, and significantly influenced prediction.

To compare all cases in the development of logistic regression-based fraud detection models, we summarized the performance metrics, as shown in Fig 5A. The best logistic regression model showed an improved F1-score of 0.46 compared to the baseline model. In summary, although only one node centrality measure appeared to be statistically significant in the logistic regression results due to multicollinearity, it significantly improved the performance of medicare fraud detection compared to the baseline model. These results suggest that node centrality measures contribute to the detection of fraudulent medical providers.

2) TREE-BASED MACHINE LEARNING ALGORITHM

Random forest, XGBoost, and LightGBM are tree-based machine-learning models. Fig 5B-D shows the performance of the models. Unlike in logistic regression, it is unnecessary to consider multicollinearity for variable selection in modeling conventional machine-learning algorithms; thus, we con-

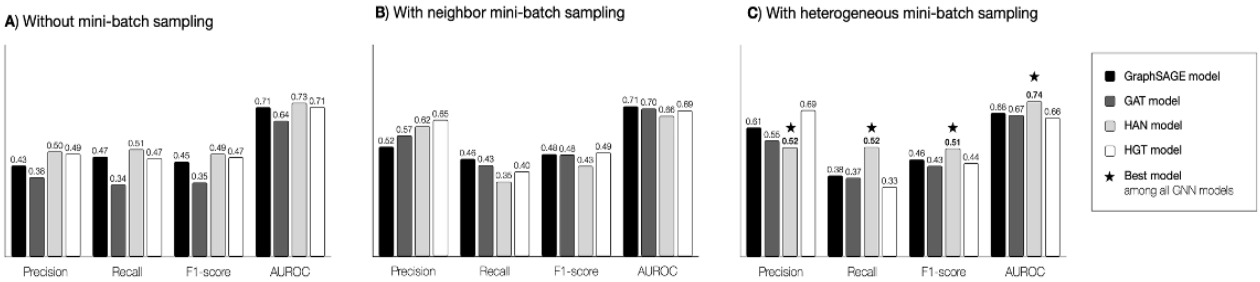

FIGURE 4. Performance of GNN models.

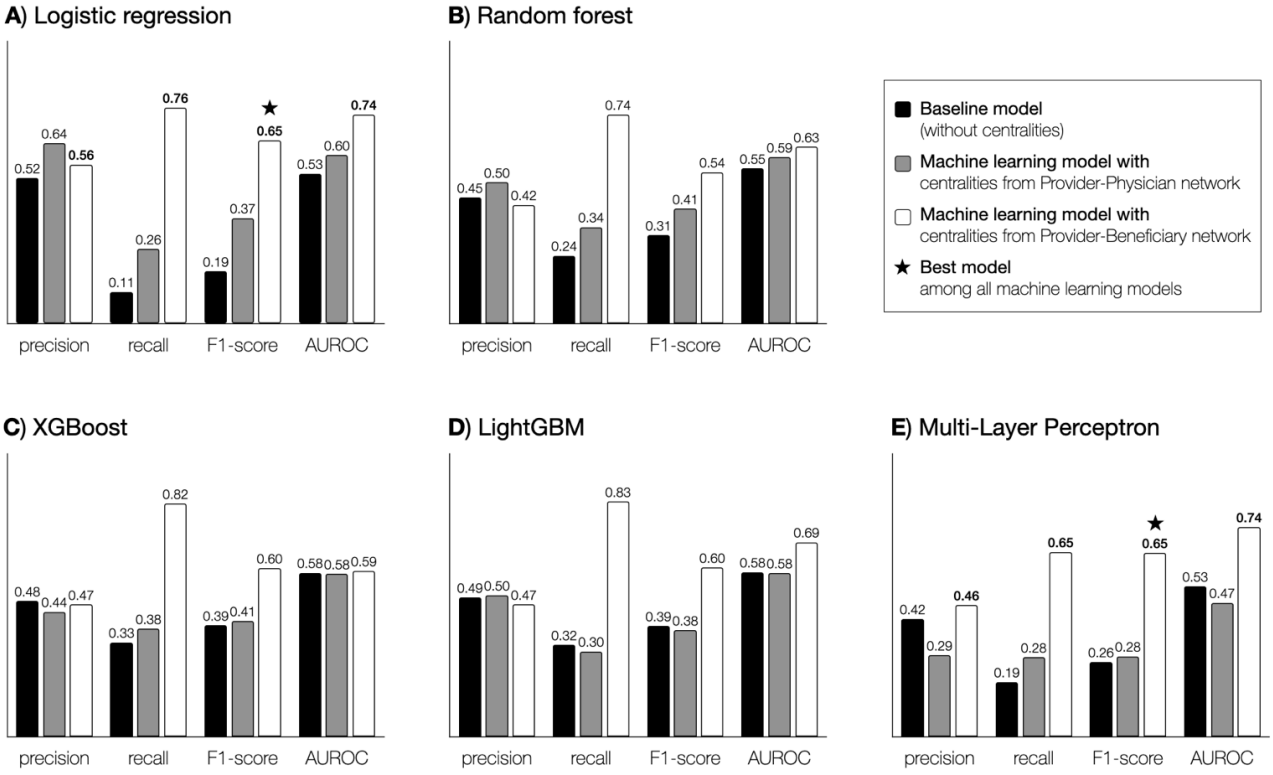

FIGURE 5. Performance of machine learning models with and without graph centrality features.

structed the models by adding all node centrality measures as input features. Therefore, the models with centrality measures obtained from the provider–beneficiary network show much more significant performance improvement than the provider–physician network. Interestingly, it appears that the improvement in precision was insignificant, while the recall was significantly improved by including centrality measures, as shown in Fig 5B-D. These results suggest that considering node centrality features in machine-learning models contributes significantly to fraud detection.

3) MLP MODEL

Fig 5E shows the performance of the MLP model for detecting fraudulent medical providers. The MLP showed slight performance improvement when using the centrality information obtained from the provider-physician network. However, when using the centrality information obtained from the provider-beneficiary network, the recall improved by 0.46 compared to the baseline model, and the F1-score increased to 0.65. In particular, MLP significantly outperformed the tree-based machine learning models based on the F1-score when adding the node centrality measures of the provider-beneficiary network.

C. COMPARISONS BETWEEN GNN MODELS AND MACHINE LEARNING MODELS

As observed, graph centrality measures significantly contributed to the improved performance of the machine learning models regarding recall and F1-score. In particular, centrality measures obtained from the provider–beneficiary

TABLE 4. Summary of logistic regression (baseline model).

Variable name	Coefficient	Standard Error	Z-statistics	P-values
Constant	-0.242	0.456	-0.53	0.5959
State_encoded	0.765	0.013	60.84	< 0.0001
Ip_Indicator_encoded	0.850	0.021	41.22	< 0.0001
Race_encoded	0.257	0.017	14.80	< 0.0001
InscClaimAmtReimbursed	0.673	0.161	4.18	< 0.0001
AllAnnualDeductibleAmt	0.310	0.099	3.15	0.0017
ChronicCond_Depression_encoded	-0.020	0.007	-2.72	0.0065
ClmDiagnosisCode_6_encoded	0.052	0.020	2.64	0.0082
NoOfMonths_PartACov	0.120	0.049	2.44	0.0147
ChronicCond_stroke_encoded	0.026	0.012	2.12	0.0343
ChronicCond_KidneyDisease_encoded	0.016	0.008	2.06	0.0395
ClmProcedureCode_4_ encoded	-0.888	0.453	-1.96	0.0498

TABLE 5. Summary of logistic regression (added centralities from provider-physician network).

Variable name	Coefficient	Standard Error	Z-statistic	P-value
Constant	-1.349	0.054	-24.88	< 0.0001
Provider_PageRank	2.882	0.034	86.01	< 0.0001
State_encoded	0.789	0.013	61.82	< 0.0001
Ip_Indicator_encoded	0.812	0.021	38.71	< 0.0001
Race_encoded	0.271	0.018	15.46	< 0.0001
InscClaimAmtReimbursed	0.698	0.162	4.31	< 0.0001
AllAnnualDeductibleAmt	0.399	0.095	4.20	0.0000
ChronicCond_Depression_encoded	-0.021	0.008	-2.76	0.0059
ClmDiagnosisCode_6_encoded	0.051	0.020	2.60	0.0092
ChronicCond_stroke_encoded	0.032	0.012	2.57	0.0102
NoOfMonths_PartACov	0.129	0.050	2.57	0.0103
OtherPhy_Indicator_encoded	-0.017	0.008	-2.12	0.0337

network contributed to a higher recall of the fraud detection model than those obtained from the provider–physician network.

We compared the performance of the proposed models, including graph centrality information as inputs, with that of the GNN models and the baseline model, which did not include graph centrality information as inputs. The HAN showed the best F1-score performance among the GNN-based models, as shown in Fig 4. In addition, by adding graph centrality features from the provider-beneficiary network, the logistic regression model demonstrated the best performance in the F1-score among the traditional machine learning-based models, as shown in Fig 5A. In summary, the machine learning models that employed the graph centrality measures obtained from provider-beneficiary showed better performance than the GNN models regarding recall and F1-score. Specifically, the best machine learning model showed improved precision of 4%p, recall of 24%p, and F1-score of 14%p compared to the best GNN model.

In addition, we calculated the learning time to confirm the efficiency of each model and measure the computational burden for learning, as shown in Fig 6. The training time for the GNN models represents the average time required to learn the 1000 epoch model ten times. We found that the training time required without mini-batch sampling and the neighbor mini-batch sampling was longer than the heterogeneous mini-batch sampling, and the training time required to learn the GNN models was much longer than that required for machine learning models. In particular, the learning time of the HAN, which was the least among the GNN models, was approximately 250–350 times more than that of logistic regression analysis. Consequently, the machine learning models using

TABLE 6. Summary of logistic regression (added centralities from provider-beneficiary network).

Variable name	Coefficient	Standard Error	Z-statistic	P-value
Constant	-2.790	0.027	-104.81	< 0.0001
Provider_PageRank	15.226	0.061	251.43	< 0.0001
Ip_Indicator	1.931	0.023	82.70	< 0.0001
State	0.478	0.016	29.38	< 0.0001
InscClaimAmtReimbursed	0.921	0.172	5.35	< 0.0001
ChronicCond_stroke	0.062	0.015	4.04	0.0001
Race	0.104	0.024	4.40	< 0.0001
Gender	-0.020	0.010	-2.06	0.0391
ClmDiagnosisCode_6	0.056	0.025	2.26	0.0237

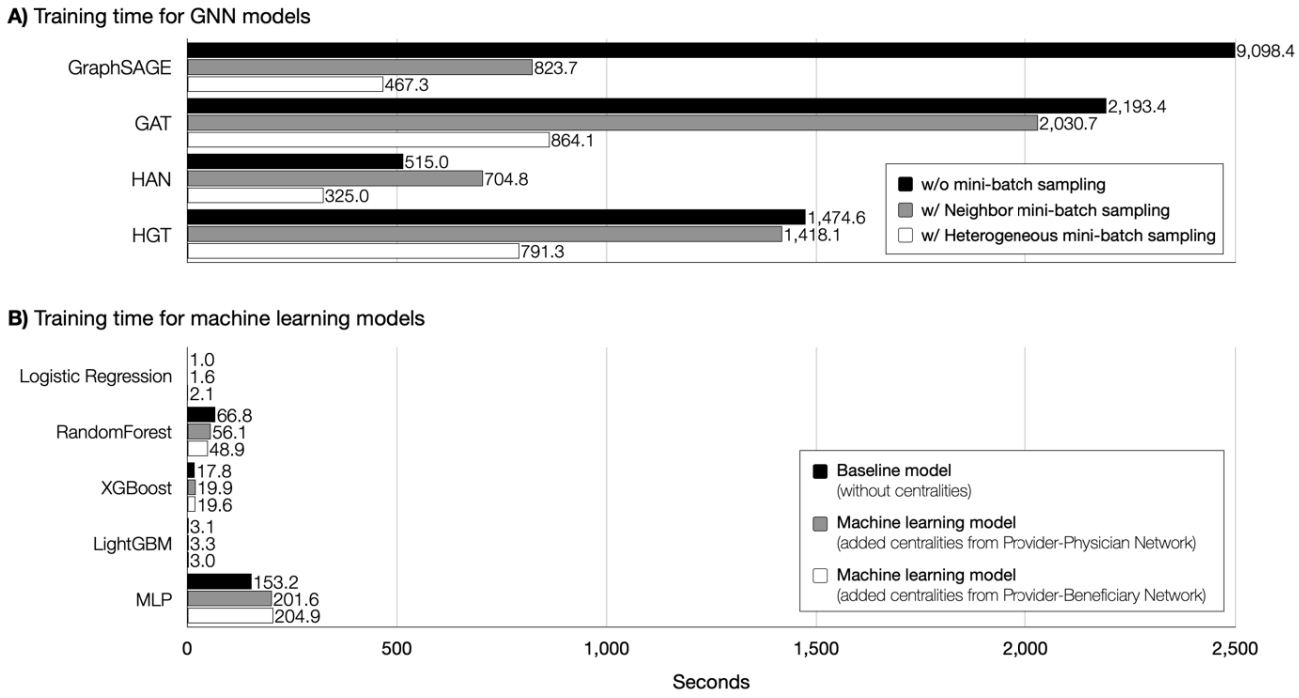

FIGURE 6. Time for training GNN and machine learning models.

node centrality measures showed better performance and shorter training times for learning than the GNN models.

V. DISCUSSION

This study focuses on applying graph information in developing a Medicare fraud detection model because complex relations exist among those who benefit from false insurance claims.

A. PERFORMANCE DIFFERENCE WITH PREVIOUS RESEARCH

We performed experiments to evaluate the performance improvement of machine learning models and GNN models when considering the relationships among multiple entities for medicare fraud detection. Table 7 summarizes the performance of machine learning or GNN models from our study and previous studies, comparing their performance with reference models used in each study.

In our study, the baseline model showed precision ranging from 0.42 to 0.52, recall ranging from 0.11 to 0.33, F1-score ranging from 0.19 to 0.39, and an AUROC value ranging from 0.53 to 0.58. When we added graph information to the machine learning models, we observed performance improvements. The best model achieved a precision of 0.56, recall of 0.76, F1-score of 0.65, and AUROC of 0.74. The performance differences compared to the baseline model appeared to be increases of 0.04∼0.14 in precision, 0.43∼0.65 in recall, 0.26∼0.46 in F1-score, and

TABLE 7. The summary of the performance difference.

Reference Paper	Reference model Performance		Proposed model Performance		The performance difference	
Kumar et al. (2010) [56]	Precision:	0.4	Precision:	0.93	Precision:	0.53
Aral et al. (2012) [58]	TP(True Positive):	0.71	TP(True Positive):	0.77	TP(True Positive):	0.06
	FP(False Positive):	0.06	FP(False Positive):	0.06	FP(False Positive):	0
	FN(False Negative):	0.08	FN(False Negative):	0.08	FN(False Negative):	0
	AUROC	0.82	AUROC:	0.86	AUROC:	0.04
Mayaki et al. (2022) [1]	Precision:	0.43 ~ 0.71	Precision:	0.77	Precision:	0.05 ~ 0.34
	AUROC:	0.71 ~ 0.86	AUROC:	0.79	AUROC:	-0.07 ~ 0.08
	AUPRC:	0.51 ~ 0.74	AUPRC:	0.76	AUPRC:	0.02 ~ 0.25
Johnson et al. (2019) [60]	AUROC:	0.80 ~ 0.81	AUROC:	0.85	AUROC:	0.04 ~ 0.05
Yoo et al. (2022) [2]	Precision:	0.52	Precision:	0.53	Precision:	0.01
	Recall:	0.11	Recall:	0.46	Recall:	0.35
	F1-score:	0.19	F1-score:	0.49	F1-score:	0.3
	AUROC:	0.53	AUROC:	0.71	ROC:	0.18
Lu et al. (2023) [61]	F1-score:	0.63 ~ 0.78	F1-score:	0.84	F1-score:	0.07 ~ 0.21
	Accuracy:	0.74 ~ 0.84	Accuracy:	0.87	Accuracy:	0.03 ~ 0.14
Branting et al. (2016) [7]	Accuracy:	0.88	F1-score:	0.95	AUROC:	0.28
	AUROC:	0.68	AUROC:	0.96		
Herland et al. (2018) [12]	AUROC:	0.74 ~ 0.8	AUROC:	0.82	AUROC:	0.01 ~ 0.07
Our Study	**Precision:**	**0.42 ~ 0.52**	**Precision:**	**0.56**	**Precision:**	**0.04 ~ 0.14**
	Recall	**0.11 ~ 0.33**	**Recall:**	**0.76**	**Recall:**	**0.43 ~ 0.65**
	F1-score:	**0.19 ~ 0.39**	**F1-score:**	**0.65**	**F1-score:**	**0.26 ~ 0.46**
	AUROC:	**0.53 ~ 0.58**	**AUROC:**	**0.74**	**AUROC:**	**0.16 ~ 0.21**

0.16∼0.21 in AUROC. When comparing the extent of performance improvement in various research studies on medicare fraud detection, our study showed greater performance improvements in most metrics. For example, compared to a previous study using a graph neural network [2], our study exhibited higher increases in precision, recall, F1-score, and AUROC.

In conclusion, our research suggests that considering graph information for medicare fraud detection leads to significant performance improvements, as well as outperforming the models proposed in previous studies in terms of the same performance metrics.

B. CONTRIBUTION OF GRAPH INFORMATION TO MEDICARE FRAUD DETECTION

We examined whether the performance of the machine learning algorithms improved by considering graph information and compared the performance of the various GNN models. As shown in Fig 5, in the case of models without graph information, the models' performance was similar across machine learning algorithms. In addition, the machine learning models without graph information represent the shortest training time as shown in Fig 6. However, intriguing results appeared in our proposed machine-learning models using the centrality information of the graph. Overall, the four performance measures were improved compared to the baseline models. In particular, the centrality features evaluated from the provider-beneficiary relationships contributed to a more significant performance improvement than that of the centrality features evaluated from the provider-physician relationships. Interestingly, the recall values increased significantly. The recall of the baseline models was approximately 0.11∼0.33, but it increased to 0.65∼0.83 in our proposed models. Note that recall is the proportion of actually fraudulent claims among the claims predicted to be fraudulent by the model, which is considered more important than other performance measures, such as precision or AUROC when evaluating the fraud detection performance.

These findings suggest that considering graph information leads to a more significant performance improvement [7], [8], [12] although previous research has shown the effectiveness of machine learning-based methodologies for medicare fraud detection [56], [57], [58], [59], [60]. Furthermore, the relationship between medicare providers and beneficiaries is more critical for detecting medicare fraud than information between providers and physicians. This is because patients, rather than physicians, play a central role in medicare fraud. In practice, medicare fraud cases are classified into seven types according to the transaction level, six of which include patients and hospitals [77].

As shown in Fig 4, although GNN is designed to learn various complex patterns of a graph-structured dataset, the GNN models showed worse performance than machine learning models with graph centrality features. Specifically, despite the highest performance of the HAN among our GNN models, its recall value was 0.52, which was significantly lower than those of the machine learning algorithms, which were 0.65 to 0.83. In addition, the F1-score of the HAN was

0.51, which is lower performance compared to the logistic regression or MLP with F1-scores of 0.65.

The previous research revealed that the GraphSAGE model demonstrated superior performance compared to the logistic regression model without considering graph centrality features [2]. However, the results of the current study indicate that the including the graph centrality features (Fig 5A) in the linear logistic regression model outperforms the HAN model (Fig 4C), which shows the best performance among GNN models. These results indicate that the GNN would be limited in the graph data structure used in our study. The GNN learns the information of neighboring nodes directly; therefore, the performance of the GNN is affected mainly by the graph structure [78]. The datasets in this study were originally tabular data; thus, we converted them to graph-structured data through the aggregation process, which would cause information loss, making it challenging to learn the graph.

This dependence of the GNN performance on the structure of medicare fraud data suggests that GNN may not be an efficient or effective method for the medicare fraud detection task, given that the computational burden of GNN is enormous. Although GNN has the advantage that each node can adaptively learn the importance of neighboring nodes in the graph structure [2], [61], it has a disadvantage in that the computational cost is too high, as shown in Fig 6. This is because the number of weight parameters to be estimated increases significantly as the amount of information at each node for every iteration increases. Therefore, it is difficult to avoid overfitting if the GNN is fitted with an over-parametric model with limited training datasets [65].

C. COST-EFFECTIVENESS OF OUR PROPOSED MACHINE LEARNING MODELS

Our proposed machine learning models to learn graph centralities, which captures the relationship between medical providers and beneficiaries, can improve performance of the fraud detection. Among the traditional tabular machine learning algorithms, logistic regression with graph centrality features (recall = 0.76) exhibited the highest performance by detecting a greater number of fraud cases compared to the GNN-based algorithm HAN model (recall = 0.52), resulting in a significant recall improvement of 24 percentage points. Such a notable performance enhancement can have broader societal advantages, serving as a means for governmental institutions and insurance companies managing healthcare insurance to operate cost-effectively by preventing medicare fraud losses. The recall represents the portion of predicted frauds by the model to the number of total actual frauds. Therefore, if we use the proposed logistic regression model with graph centrality features instead of the GNN model, we can detect more frauds at which the recall value is increased, and thus, reduce losses due to the frauds. Assuming that all insurance companies replace the GNN model with the proposed logistic regression model in real life, the social benefit by the model improvement would be evaluated as 3.1 billion euros (=13 billion euros * (76% recall of logistic model − 52% recall of GNN model)) and 5 billion dollars (=21 billion dollars * (76% recall of logistic model − 52% recall of GNN model)) in Europe and the United States, respectively. Additionally, it is worth noting that the training time required for GNN models was significantly longer compared to machine-learning models. Considering computational costs, developing a machine learning model using node centrality measures are more efficient and effective than employing GNN models. Consequently, the incorporation of graph centrality measures into traditional machine learning algorithms not only enhances fraud detection performance but also holds the potential for societal benefits, enabling more efficient cost management for governmental institutions and insurance companies involved in healthcare insurance operations.

D. LIMITATIONS

There are several limitations in this study. First, the level of class-balance of datasets we used in our research would be different from the class-imbalanced datasets commonly utilized in other fraud detection tasks [79], [80], [81], [82], [83]. The ratio of normal to fraudulent cases in the dataset used in this study is approximately 6:4, indicating almost no class imbalance. We did not consider any methods to address the class imbalance in order to focus on the performance enhancement when incorporating graph analysis. Therefore, the research of class-imbalance resolution methods to improve fraud detection performance is left as future research. Second, the adoption of GNN models necessitates the transformation of tabular datasets into graph-structured datasets. In this study, we have constructed a heterogeneous graph composed of medical providers and beneficiary nodes by aggregating the summation of node embeddings generated by different relations. However, there would be alternative methodologies for node representation because of the absence of a definitive aggregation method. Third, the dataset utilized in our study was obtained from Kaggle, a renowned ''machine learning and data science open community.'' Unfortunately, the data description on Kaggle did not provide details to understand the precise origin of the datasets, including whether they were sampled or obtained from a specific insurance company or governmental institute. Consequently, while the data holds substantial value for the development of a medicare fraud detection model, it is limited in that the specific population represented by the data cannot be accurately known.

VI. CONCLUSION

Many studies have been conducted on fraud detection using graph analysis because it considers the relationship between objects related to fraud. For graph analysis, information from the nodes and edges connected to the objects included in the dataset can be used. Recently, various fraud detection studies have been conducted with the emergence of GNN algorithms that can learn feature information from a graph-

structured dataset. Because GNNs learn information about neighboring nodes directly using artificial intelligence, the GNN model improves prediction performance in a graph-structured dataset.

This study is the first to apply various GNN and machine learning algorithms to detect medicare fraud. As collusion among physicians, beneficiaries, and providers often causes medicare fraud rather than individually, it is essential to consider the relationship between them using graph analysis. Nevertheless, to our knowledge, no studies have compared the performance of various GNN algorithms and other conventional methodologies for medicare fraud detection tasks.

In this study, we developed fraud-detection models using two approaches to reflect graph information: a graph GNN and a conventional machine-learning model with input features of graph centralities. For the GNN algorithm, we constructed a heterogeneous graph dataset composed of medical providers and beneficiary nodes using a combined tabular dataset. To conduct node classification inductively, we developed four GNN models (GraphSAGE, GAT, HAN, and HGT). We extracted nodal centrality features from two bipartite graphs for machine learning algorithms: the provider-physician and provider-beneficiary network. As for the centrality measures, we computed degree centrality, eigenvector centrality, closeness centrality, and PageRank and used these measures as input features of machine learning models. We developed five machine learning models (logistic regression, random forest, XGBoost, LightGBM, and MLP) to conduct classification to detect fraudulent medical providers.

This study applied graph-based learning to improve the performance of medicare fraud detection. Compared to the baseline model, the proposed machine learning model with graph centrality features obtained from the provider–physician network showed improved performance in terms of recall and F1-score. Furthermore, graph centralities obtainedfrom the provider–beneficiary network rather than the provider–physician network contributed more to performance improvement in measures such as recall, F1-score, and AUROC. These results imply that patients, rather than physicians, play a central role in medicare fraud.

In future studies, it would be necessary to find out a more generalized fraud detection model by dealing with a class-imbalanced problem. Additionally, it would be worth exploring various node embedding methods to contribute to the performance of GNN models. Further research in these directions leads to the development of a more robust medicare fraud detection model.

ACKNOWLEDGMENT

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

REFERENCES

[1] M. Zoubeirou, A. Mayaki, and M. Riveill, "Multiple inputs neural networks for medicare fraud detection," 2022, *arXiv:2203.05842*.

[2] Y. Yoo, D. Shin, D. Han, S. Kyeong, and J. Shin, "Medicare fraud detection using graph neural networks," in *Proc. Int. Conf. Electr., Comput. Energy Technol. (ICECET)*, Jul. 2022, pp. 1–5, doi: 10.1109/ICECET55527.2022.9872963.

[3] N. Kurani, J. Ortaliza, E. Wager, L. Fox, and K. Amin. (2022). *How Has U.S. Spending on Healthcare Changed Over Time?* Health Spending. [Online]. Available: http://www.healthsystemtracker.org/chart-collection/u-s-spending-healthcare-changed-time

[4] A. Islam, M. Corney, G. Mohay, A. Clark, S. Bracher, T. Raub, and U. Flegel, "Detecting collusive fraud in enterprise resource planning systems," in *Advances in Digital Forensics VII* (IFIP Advances in Information and Communication Technology). Berlin, Germany: Springer, 2011, pp. 143–153.

[5] G. Sadowski and P. Rathle. (2014). *Fraud Detection: Discovering Connections With Graph Databases*. Accessed: Oct. 6, 2022. [Online]. Available: https://go.neo4j.com/rs/710-RRC-335/images/Neo4j_WP-Fraud-Detection-with-Graph-Databases.pdf?_ga=2.152229817.1435723348.1577409683-120002542.1565112145

[6] V. Chandola, S. R. Sukumar, and J. C. Schryver, "Knowledge discovery from massive healthcare claims data," in *Proc. 19th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, New York, NY, USA, Aug. 2013, pp. 1312–1320, doi: 10.1145/2487575.2488205.

[7] L. K. Branting, F. Reeder, J. Gold, and T. Champney, "Graph analytics for healthcare fraud risk estimation," in *Proc. IEEE/ACM Int. Conf. Adv. Social Netw. Anal. Mining (ASONAM)*, Aug. 2016, pp. 845–851, doi: 10.1109/ASONAM.2016.7752336.

[8] J. Liu, E. Bier, A. Wilson, J. A. Guerra-Gomez, T. Honda, K. Sricharan, L. Gilpin, and D. Davies, "Graph analysis for detecting fraud, waste, and abuse in health-care data," *AI Mag.*, vol. 37, no. 2, pp. 33–46, Jun. 2016, doi: 10.1609/aimag.v37i2.2630.

[9] Z. Xie, R. Zhu, J. Liu, G. Zhou, J. X. Huang, and X. Cui, "GFCNet: Utilizing graph feature collection networks for coronavirus knowledge graph embeddings," *Inf. Sci.*, vol. 608, pp. 1557–1571, Aug. 2022, doi: 10.1016/j.ins.2022.07.031.

[10] Y. Ding, Z. Zhang, X. Zhao, D. Hong, W. Li, W. Cai, and Y. Zhan, "AF2GNN: Graph convolution with adaptive filters and aggregator fusion for hyperspectral image classification," *Inf. Sci.*, vol. 602, pp. 201–219, Jul. 2022, doi: 10.1016/j.ins.2022.04.006.

[11] S. Fu, W. Liu, D. Tao, Y. Zhou, and L. Nie, "HesGCN: Hessian graph convolutional networks for semi-supervised classification," *Inf. Sci.*, vol. 514, pp. 484–498, Apr. 2020, doi: 10.1016/j.ins.2019.11.019.

[12] M. Herland, T. M. Khoshgoftaar, and R. A. Bauder, "Big data fraud detection using multiple medicare data sources," *J. Big Data*, vol. 5, no. 1, p. 29, Dec. 2018, doi: 10.1186/s40537-018-0138-3.

[13] G. Wang, S. Xie, B. Liu, and P. S. Yu, "Review graph based online store review spammer detection," in *Proc. IEEE 11th Int. Conf. Data Mining*, Dec. 2011, pp. 1242–1247, doi: 10.1109/ICDM.2011.124.

[14] A. Breuer, R. Eilat, and U. Weinsberg, "Friend or faux: Graph-based early detection of fake accounts on social networks," in *Proc. Web Conf.*, New York, NY, USA, Apr. 2020, pp. 1287–1297, doi: 10.1145/3366423.3380204.

[15] J. Jia, B. Wang, and N. Z. Gong, "Random walk based fake account detection in online social networks," in *Proc. 47th Annu. IEEE/IFIP Int. Conf. Dependable Syst. Netw. (DSN)*, Jun. 2017, pp. 273–284, doi: 10.1109/DSN.2017.55.

[16] G. Danezis and P. Mittal, "SybilInfer: Detecting Sybil nodes using social networks," in *Proc. NDSS*, Sep. 2009, pp. 1–15.

[17] Y. Boshmaf, D. Logothetis, G. Siganos, J. Lería, J. Lorenzo, M. Ripeanu, K. Beznosov, and H. Halawa, "Ìntegro: Leveraging victim prediction for robust fake account detection in large scale OSNs," *Comput. Secur.*, vol. 61, pp. 142–168, Aug. 2016, doi: 10.1016/j.cose.2016.05.005.

[18] Q. Cao, M. Sirivianos, X. Yang, and T. Pregueiro, "Aiding the detection of fake accounts in large scale social online services," in *Proc. 9th USENIX Conf. Netw. Syst. Design Implement.*, 2012, p. 15.

[19] H. Yu, M. Kaminsky, P. B. Gibbons, and A. D. Flaxman, "SybilGuard: Defending against Sybil attacks via social networks," *IEEE/ACM Trans. Netw.*, vol. 16, no. 3, pp. 576–589, Jun. 2008, doi: 10.1109/TNET.2008.923723.

[20] H. Yu, P. B. Gibbons, M. Kaminsky, and F. Xiao, "SybilLimit: A near-optimal social network defense against sybil attacks," in *Proc. IEEE Symp. Secur. Privacy*, May 2008, pp. 3–17, doi: 10.1109/SP.2008.13.

[21] P. Giudici, B. Hadji-Misheva, and A. Spelta, "Network based credit risk models," *Qual. Eng.*, vol. 32, no. 2, pp. 199–211, Apr. 2020, doi: 10.1080/08982112.2019.1655159.

[22] M. Yıldırım, F. Y. Okay, and S. Özdemir, "Big data analytics for default prediction using graph theory," *Expert Syst. Appl.*, vol. 176, Aug. 2021, Art. no. 114840, doi: 10.1016/j.eswa.2021.114840.

[23] D. Lautier and F. Raynaud, "Systemic risk in energy derivative markets: A graph-theory analysis," *Energy J.*, vol. 33, no. 3, pp. 215–239, Jul. 2012. [Online]. Available: http://www.jstor.org/stable/23268099

[24] A. A. Canutescu, A. A. Shelenkov, and R. L. Dunbrack, "A graph-theory algorithm for rapid protein side-chain prediction," *Protein Sci.*, vol. 12, no. 9, pp. 2001–2014, Sep. 2003, doi: 10.1110/ps.03154503.

[25] D. J. Jacobs, A. J. Rader, L. A. Kuhn, and M. F. Thorpe, "Protein flexibility predictions using graph theory," *Proteins, Struct., Function, Genet.*, vol. 44, no. 2, pp. 150–165, Aug. 2001, doi: 10.1002/prot.1081.

[26] B. Feng, H. Xu, W. Xue, and B. Xue, "Every corporation owns its structure: Corporate credit ratings via graph neural networks," 2020, *arXiv:2012.01933*.

[27] S. Wu, W. Zhang, F. Sun, and B. Cui, "Graph neural networks in recommender systems: A survey," *ACM Comput. Surv.*, vol. 55, no.'5, pp. 1–37, 2022, doi: 10.1145/3535101.

[28] Z. Sun, B. Wu, Y. Wang, and Y. Ye, "Sequential graph collaborative filtering," *Inf. Sci.*, vol. 592, pp. 244–260, May 2022, doi: 10.1016/j.ins.2022.01.064.

[29] J. Liao, W. Zhou, F. Luo, J. Wen, M. Gao, X. Li, and J. Zeng, "SocialLGN: Light graph convolution network for social recommendation," *Inf. Sci.*, vol. 589, pp. 595–607, Apr. 2022, doi: 10.1016/j.ins.2022.01.001.

[30] N. Khan, Z. Ma, A. Ullah, and K. Polat, "Similarity attributed knowledge graph embedding enhancement for item recommendation," *Inf. Sci.*, vol. 613, pp. 69–95, Oct. 2022, doi: 10.1016/j.ins.2022.08.124.

[31] X. Gao, F. Feng, H. Huang, X.-L. Mao, T. Lan, and Z. Chi, "Food recommendation with graph convolutional network," *Inf. Sci.*, vol. 584, pp. 170–183, Jan. 2022, doi: 10.1016/j.ins.2021.10.040.

[32] C. Cao, S. Li, S. Yu, and Z. Chen, "Fake reviewer group detection in online review systems," 2021, *arXiv:2112.06403*.

[33] J. Wang, R. Wen, C. Wu, Y. Huang, and J. Xiong, "FdGars: Fraudster detection via graph convolutional networks in online app review system," in *Proc. Companion World Wide Web Conf.*, New York, NY, USA, May 2019, pp. 310–316, doi: 10.1145/3308560.3316586.

[34] C. Liang, Z. Liu, B. Liu, J. Zhou, X. Li, S. Yang, and Y. Qi, "Uncovering insurance fraud conspiracy with network learning," in *Proc. 42nd Int. ACM SIGIR Conf. Res. Develop. Inf. Retr.*, New York, NY, USA, Jul. 2019, pp. 1181–1184, doi: 10.1145/3331184.3331372.

[35] Z. Liu, C. Chen, X. Yang, J. Zhou, X. Li, and L. Song, "Heterogeneous graph neural networks for malicious account detection," in *Proc. 27th ACM Int. Conf. Inf. Knowl. Manage.*, New York, NY, USA, Oct. 2018, pp. 2077–2085, doi: 10.1145/3269206.3272010.

[36] C. Liu, L. Sun, X. Ao, J. Feng, Q. He, and H. Yang, "Intention-aware heterogeneous graph attention networks for fraud transactions detection," in *Proc. 27th ACM SIGKDD Conf. Knowl. Discovery Data Mining*, New York, NY, USA, Aug. 2021, pp. 3280–3288, doi: 10.1145/3447548.3467142.

[37] M. Weber, G. Domeniconi, J. Chen, D. Karl I. Weidele, C. Bellei, T. Robinson, and C. E. Leiserson, "Anti-money laundering in Bitcoin: Experimenting with graph convolutional networks for financial forensics," 2019, *arXiv:1908.02591*.

[38] A. Li, Z. Qin, R. Liu, Y. Yang, and D. Li, "Spam review detection with graph convolutional networks," in *Proc. 28th ACM Int. Conf. Inf. Knowl. Manage.*, Beijing, China, Nov. 2019, pp. 2703–2711, doi: 10.1145/3357384.3357820.

[39] Y. Dou, Z. Liu, L. Sun, Y. Deng, H. Peng, and P. S. Yu, "Enhancing graph neural network-based fraud detectors against camouflaged fraudsters," in *Proc. 29th ACM Int. Conf. Inf. Knowl. Manage.*, Oct. 2020, pp. 315–324, doi: 10.1145/3340531.3411903.

[40] J. Zhou, G. Cui, S. Hu, Z. Zhang, C. Yang, Z. Liu, L. Wang, C. Li, and M. Sun, "Graph neural networks: A review of methods and applications," *AI Open*, vol. 1, pp. 57–81, Jan. 2020, doi: 10.1016/j.aiopen.2021.01.001.

[41] C. Wang, Y. Dou, M. Chen, J. Chen, Z. Liu, and P. S. Yu, "Deep fraud detection on non-attributed graph," in *Proc. IEEE Int. Conf. Big Data*, Dec. 2021, pp. 5470–5473, doi: 10.1109/BigData52589.2021.9672028.

[42] Y. Liu, Z. Sun, and W. Zhang, "Improving fraud detection via hierarchical attention-based graph neural network," 2022, *arXiv:2202.06096*.

[43] J. Zhao, X. Liu, Q. Yan, B. Li, M. Shao, and H. Peng, "Multi-attributed heterogeneous graph convolutional network for bot detection," *Inf. Sci.*, vol. 537, pp. 380–393, Oct. 2020, doi: 10.1016/j.ins.2020.03.113.

[44] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," 2016, *arXiv:1609.02907*.

[45] W.-L. Chiang, X. Liu, S. Si, Y. Li, S. Bengio, and C.-J. Hsieh, "Cluster-GCN: An efficient algorithm for training deep and large graph convolutional networks," in *Proc. 25th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Anchorage, AK, USA, Jul. 2019, pp. 257–266, doi: 10.1145/3292500.3330925.

[46] W. L. Hamilton, R. Ying, and J. Leskovec, "Inductive representation learning on large graphs," in *Proc. Adv. Neural Inf. Process. Syst.*, Long Beach, CA, USA: Curran Associates, 2017, pp. 1025–1035.

[47] A. Vaswani, N. Shazeer, N. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," in *Proc. Adv. Neural Inf. Process. Syst.*, I. Guyon, U. von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, Eds. Red Hook, NY, USA: Curran Associates, 2017. [Online]. Available: https://proceedings.neurips.cc/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf

[48] P. Veličković, G. Cucurull, A. Casanova, A. Romero, P. Lioss, and Y. Bengio, "Graph attention networks," 2017, *arXiv:1710.1090s3*.

[49] J. Zhao, X. Wang, C. Shi, B. Hu, G. Song, and Y. Ye, "Heterogeneous graph structure learning for graph neural networks," in *Proc. Innov. Appl. Artif. Intell. Conf.*, May 2021, pp. 4697–4705, doi: 10.1609/aaai.v35i5.16600.

[50] S. Ma, J.-W. Liu, X. Zuo, and W.-M. Li, "Heterogeneous graph gated attention network," in *Proc. Int. Joint Conf. Neural Netw.*, New York, NY, USA, Jul. 2021, pp. 1–6, doi: 10.1109/IJCNN52387.2021.9533711.

[51] Z. Hu, Y. Dong, K. Wang, and Y. Sun, "Heterogeneous graph transformer," in *Proc. Web Conf.*, New York, NY, USA, Apr. 2020, pp. 2704–2710, doi: 10.1145/3366423.3380027.

[52] F. Yasmin, Md. M. Hassan, M. Hasan, S. Zaman, C. Kaushal, W. El-Shafai, and N. F. Soliman, "PoxNet22: A fine-tuned model for the classification of monkeypox disease using transfer learning," *IEEE Access*, vol. 11, pp. 24053–24076, 2023, doi: 10.1109/ACCESS.2023.3253868.

[53] M. Mehedi Hassan, S. Mollick, and F. Yasmin, "An unsupervised cluster-based feature grouping model for early diabetes detection," *Healthcare Anal.*, vol. 2, Nov. 2022, Art. no. 100112, doi: 10.1016/j.health.2022.100112.

[54] N. J. Prottasha, S. A. Murad, A. J. M. Muzahid, M. Rana, M. Kowsher, A. Adhikary, S. Biswas, and A. K. Bairagi, "Impact learning: A learning method from feature's impact and competition," *J. Comput. Sci.*, vol. 69, May 2023, Art. no. 102011, doi: 10.1016/j.jocs.2023.102011.

[55] M. M. Hassan, M. M. Hassan, F. Yasmin, M. A. R. Khan, S. Zaman, K. K. Islam, and A. K. Bairagi, "A comparative assessment of machine learning algorithms with the least absolute shrinkage and selection operator for breast cancer detection and prediction," *Decis. Anal. J.*, vol. 7, Jun. 2023, Art. no. 100245, doi: 10.1016/j.dajour.2023.100245.

[56] M. Kumar, R. Ghani, and Z.-S. Mei, "Data mining to predict and prevent errors in health insurance claims processing," in *Proc. 16th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Washington, DC, USA, Jul. 2010, pp. 65–74, doi: 10.1145/1835804.1835816.

[57] H. Shin, H. Park, J. Lee, and W. C. Jhee, "A scoring model to detect abusive billing patterns in health insurance claims," *Expert Syst. Appl.*, vol. 39, no. 8, pp. 7441–7450, Jun. 2012, doi: 10.1016/j.eswa.2012.01.105.

[58] K. D. Aral, H. A. Güvenir, İ. Sabuncuoğlu, and A. R. Akar, "A prescription fraud detection model," *Comput. Methods Programs Biomed.*, vol. 106, no. 1, pp. 37–46, Apr. 2012, doi: 10.1016/j.cmpb.2011.09.003.

[59] P. A. Ortega, C. J. Figueroa, and G. A. Ruz, "A medical claim fraud/abuse detection system based on data mining: A case study in Chile," in *Proc. Int. Conf. Data Mining*, Las Vegas, NV, USA, 2006, pp. 1–12.

[60] J. M. Johnson and T. M. Khoshgoftaar, "Medicare fraud detection using neural networks," *J. Big Data*, vol. 6, no. 1, p. 63, Dec. 2019, doi: 10.1186/s40537-019-0225-0.

[61] J. Lu, K. Lin, R. Chen, M. Lin, X. Chen, and P. Lu, "Health insurance fraud detection by using an attributed heterogeneous information network with a hierarchical attention mechanism," *BMC Med. Informat. Decis. Making*, vol. 23, no. 1, p. 62, Apr. 2023, doi: 10.1186/s12911-023-02152-0.

[62] R. A. Gupta. (2018). *Kaggle Healthcare Provider Fraud detection Datasets*. [Online]. Available: http://www.kaggle.com/datasets/rohitrox/healthcare-provider-fraud-detection-analysis

[63] W. L. Hamilton, "The graph neural network model," in *Graph Representation Learning*. Cham, Switzerland: Springer, 2020, pp. 51–70, doi: 10.1007/978-3-031-01588-5_5.
[64] K. Zhou, Y. Dong, K. Wang, W. Sun Lee, B. Hooi, H. Xu, and J. Feng, "Understanding and resolving performance degradation in graph convolutional networks," 2020, *arXiv:2006.07107*.
[65] Y. Rong, W. Huang, T. Xu, and J. Huang, "DropEdge: Towards deep graph convolutional networks on node classification," in *Proc. ICLR*, 2020, pp. 1–13.
[66] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A simple way to prevent neural networks from overfitting," *J. Mach. Learn. Res.*, vol. 15, no. 1, pp. 1929–1958, 2014.
[67] D. Zou, Z. Hu, Y. Wang, S. Jiang, Y. Sun, and Q. Gu, "Layer-dependent importance sampling for training deep and large graph convolutional networks," 2019, *arXiv:1911.07323*.
[68] J. Chen, T. Ma, and C. Xiao, "FastGCN: Fast learning with graph convolutional networks via importance sampling," 2018, *arXiv:1801.10247*.
[69] L. Bottou, F. E. Curtis, and J. Nocedal, "Optimization methods for large-scale machine learning," *SIAM Rev.*, vol. 60, no. 2, pp. 223–311, Jan. 2018, doi: 10.1137/16M1080173.
[70] D. Masters and C. Luschi, "Revisiting small batch training for deep neural networks," 2018, *arXiv:1804.07612*.
[71] X. Qian and D. Klabjan, "The impact of the mini-batch size on the variance of gradients in stochastic gradient descent," 2020, *arXiv:2004.13146*.
[72] M. Newman, *Networks: An Introduction*. New York, NY, USA: Oxford Univ. Press, 2010, doi: 10.1093/acprof:oso/9780199206650.001.0001.
[73] M. Seera, C. P. Lim, A. Kumar, L. Dhamotharan, and K. H. Tan, "An intelligent payment card fraud detection system," *Ann. Oper. Res.*, vol. 10, pp. 1–23, Jan. 2021, doi: 10.1007/s10479-021-04149-2.
[74] V. N. Dornadula and S. Geetha, "Credit card fraud detection using machine learning algorithms," *Proc. Comput. Sci.*, vol. 165, pp. 631–641, Jan. 2019, doi: 10.1016/j.procs.2020.01.057.
[75] N. Khare and S. Y. Sait, "Credit card fraud detection using machine learning models and collating machine learning models," *Int. J. Appl. Math.*, vol. 118, no. 20, pp. 825–838, 2018.
[76] M. Buckland and F. Gey, "The relationship between recall and precision," *J. Amer. Soc. Inf. Sci.*, vol. 45, no. 1, pp. 12–19 2014.
[77] D. Thornton, R. M. Mueller, P. Schoutsen, and J. van Hillegersberg, "Predicting healthcare fraud in medicaid: A multidimensional data model and analysis techniques for fraud detection," *Proc. Technol.*, vol. 9, pp. 1252–1264, Jan. 2013, doi: 10.1016/j.protcy.2013.12.140.
[78] B. Sanchez-Lengeling, E. Reif, A. Pearce, and A. Wiltschko, "A gentle introduction to graph neural networks," *Distill*, vol. 6, no. 8, pp. 1–12, Aug. 2021, doi: 10.23915/distill.00033.
[79] S. Makki, Z. Assaghir, Y. Taher, R. Haque, M.-S. Hacid, and H. Zeineddine, "An experimental study with imbalanced classification approaches for credit card fraud detection," *IEEE Access*, vol. 7, pp. 93010–93022, 2019, doi: 10.1109/ACCESS.2019.2927266.
[80] S. N. Kalid, K.-H. Ng, G.-K. Tong, and K.-C. Khor, "A multiple classifiers system for anomaly detection in credit card data with unbalanced and overlapped classes," *IEEE Access*, vol. 8, pp. 28210–28221, 2020, doi: 10.1109/ACCESS.2020.2972009.
[81] E. Ileberi, Y. Sun, and Z. Wang, "Performance evaluation of machine learning methods for credit card fraud detection using SMOTE and AdaBoost," *IEEE Access*, vol. 9, pp. 165286–165294, 2021, doi: 10.1109/ACCESS.2021.3134330.
[82] G. Zhang, J. Wu, J. Yang, A. Beheshti, S. Xue, C. Zhou, and Q. Z. Sheng, "FRAUDRE: Fraud detection dual-resistant to graph inconsistency and imbalance," in *Proc. IEEE Int. Conf. Data Mining (ICDM)*, Dec. 2021, pp. 867–876, doi: 10.1109/ICDM51629.2021.00098.
[83] K. Ding, K. Shu, X. Shan, J. Li, and H. Liu, "Cross-domain graph anomaly detection," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 33, no. 6, pp. 2406–2415, Jun. 2022, doi: 10.1109/TNNLS.2021.3110982.

YEEUN YOO received the M.S. degree in statistics from Dongguk University, Seoul, Republic of Korea, in February 2020. She has been a Data Scientist with KakaoBank Corporation, Kyeonggi-do, Republic of Korea, since February 2022. She has worked on various projects related to these fields and has contributed to the development of several data-driven solutions. With her expertise in data science and machine learning, she strives to create value from data and apply it to real-world problems. Her research interests include machine learning, deep learning, fraud detection in time series data, and graph neural networks.

JINHO SHIN received the M.S. degree in management engineering from KAIST and the Ph.D. degree in economics from Sungkyunkwan University. He is currently the Head of the Research and Development Team, KakaoBank. He has accumulated extensive experience in the financial industry. His positions held include the Head of Consumer Banking or Credit Risk Management with Korea Credit Bureau (KCB), Standard Chartered Bank, and other financial institutions. His publications include research on credit scoring, fraud detection, housing market, and financial economics in various academic journals. His research interests include financial economics, real estate market, risk management, financial technology, big-data analytics, and machine learning.

SUNGHYON KYEONG received the M.S. degrees in physics and the Ph.D. degree in medical science from Yonsei University, Seoul, Republic of Korea, in February 2009 and February 2016, respectively. He has published more than 50 peer-reviewed international journals in the fields of artificial intelligence and machine learning. With a strong background in both physics and medical science, his research interests include the development of explainable machine learning models and fraud detection models. He has contributed to several healthcare and finance-related projects throughout his career. One of his notable contributions to the field is the development of an explainable credit scoring model, which has been successfully applied in the finance industry. He continues to conduct cutting-edge research and development in the financial and healthcare domains, seeking new, and innovative ways to apply artificial intelligence and data analytics to benefit our society.