Journal of Network Intelligence
Taiwan Ubiquitous Information
 ISSN 2414-8105 (Online)
Volume **7**, Number **2**, May **2022**

Medical Insurance Fraud Detection using Graph Neural Networks with Spatio-temporal Constraints

Jin-Po Chen

School of Computer and Information Engineering
Xiamen University of Technology
Xiamen, 361024, China
2022032015@s.xmut.edu.cn

Ping Lu*

School of Computer and Information Engineering
Xiamen University of Technology
Xiamen, 361024, China
Key Laboratory of Ecological Environment and Information Atlas, Fujian Provincial University
Putian 351100, China
Corresponding Author: luping@xmut.edu.cn

Fan Yang*

Shenzhen Research Institute of Xiamen University, Shenzhen, China
Dept.of Automation, Xiamen University, Xiamen, Fujian, China
Corresponding Author: yang@xmu.edu.cn

Ruicong Chen

School of Computer and Information Engineering
iamen University of Technology
Xiamen, Fujian, China
rcchen@163.com

Kaibiao Lin

School of Computer and Information Engineering
Xiamen University of Technology
Xiamen, 361024, China
kblin@xmut.edu.cn

Received January 2022; revised March 2022
(Communicated by Ping Lu* and Fan Yang*)

Abstract. *With the rapid development of social insurance in China, medical insurance fraud is increasing, seriously threatening the stable operation of the medical insurance system. Traditional machine learning, anomaly detection, and other medical insurance fraud detection methods focus only on feature attributes, ignoring the rich behavioral attributes in the medical process. Considering the behavioral attributes, medical insurance fraud mainly comprises two domains: spatial and temporal, corresponding to gang fraud and individual anomaly, respectively. Therefore, this paper proposes a medical insurance fraud detection method utilizing a spatiotemporal constraint graph neural network (StGNN). Specifically, we first construct multiple medical insurance heterogeneous graphs based on the medical insurance dataset. Next, groups with the same behavioral trajectory are sampled using multi-semantic metapaths, while a two-layer attention mechanism assigns neighboring nodes and different behavioral trajectory weights. Then we use Bi-LSTM to detect individual anomalies during the patients' treatment process. Finally, the proposed model is trained end-to-end, employing a cross-entropy loss function. Extensive experiments on two real datasets show that StGNN outperforms the existing methods on the anomaly detection task.*

Keywords: Medical Insurance, Fraud Detection, Graph Neural Network, Spatiotemporal

1. **Introduction.** Medical insurance generally involves stakeholders, i.e., the insured, designated medical institutions, e.g., hospitals and pharmacies, and doctors. Given the many links, long chains, and risk points in the medical insurance system, it is easy to breed health insurance fraud problems. A survey of 33 institutions in six European countries found that health care fraud loss rates ranged from 3.29% - 10.0% (5.59% on average), and health care fraud rates ranged from 0.47% - 7.1% (4.23% on average). Annual health insurance fraud amounts to €180 billion, equivalent to 6% of global health spending [1]. As China's health insurance coverage expands, several unscrupulous individuals engage in health insurance fraud for personal gain. In 2020 alone, 40,700 defaulting institutions and 26,100 participants were investigated and punished, recovering $3,487 million in health insurance funds [2]. Therefore, identifying fraudsters from the complex health insurance data is an urgent issue requiring attention to ensure the vigorous operation of the health insurance system and the proper use of health insurance funds.

Traditional fraud detection methods usually rely on handcrafted rules designed by experts that filter anomalous data. However, new fraud methods emerge as time passes, and it is difficult for the existing rules to detect new frauds. Hence, constant human effort is required to update the rules and ensure timeliness. With the rise of machine learning, several supervised [3] and unsupervised learning [4] methods have been applied to detect medical insurance fraud. Supervised learning methods require many labeled samples commonly labeled by experts. Supervised learning is a relatively well-performing scheme. On the contrary, the unsupervised learning approach does not require sample labels and focuses on finding outliers in the sample. For example, it aims to find outliers considering the number of drugs, the amount spent, and the number of doctor visits. Nevertheless, its effect is poor compared with supervised learning. Overall, each of these methods (supervised and unsupervised) has its strengths and limitations. However, they all solely focus on the sample's feature level, ignoring the behavioral attributes of the patient throughout the visit, which can also be an essential credential for fraud detection. For example, fraudulent patients swiped medical insurance cards from multiple hospitals to receive a large number of the same drugs during the same period or prescribed many drugs unrelated to a hospital's department. Nevertheless, these behavioral interactions are challenging to produce features.

Considering the behavioral attributes, health care fraud comprises two dimensions, space and time. Many frauds involving large amounts of money in medical insurance fraud are gang frauds. The fraudulent techniques may differ between each gang, but fraud may be similar or even identical within the same gang. This is because fraud gangs usually study several fraudulent methods, and the more people and times they use a method, a fraud pattern is created. In addition, researching a new fraud method requires a lot of time and effort, and due to cost and time considerations, fraudsters will not utilize a new method every time. As long as their existing fraudulent methods are not discovered, they use them continuously. For example, members of the fraudulent gang will often go to the same hospital to buy drugs from the same doctor, which indicates that the medical treatment trajectory of the fraudulent members is similar in the spatial dimension. The temporal dimension can detect abnormalities in individual patients during the entire treatment course. For example, a disease that can be cured in only half a year has been prescribed for a year continuously, or chronic disease with a stable treatment pattern presents a steep increase in drug spending during a particular month. Thus, the probability of detecting fraud can be improved by adding spatiotemporal constraints.

However, medical insurance data is intricate, comprising tables containing personal information, medical information, and medical details. Hence, it is difficult to mine behavioral attributes directly from tabular data, and this is where graph network analysis provides a new direction for mining behavioral attributes. Graph network techniques [5, 6] have previously achieved good results on citation networks [7, 8], social networks [9], and movie networks. Moreover, graph structures are also widely available in medical insurance datasets. The latter datasets can be viewed as a heterogeneous graph [10] consisting of patients, hospital departments, drugs, and visit dates. The edges in the heterogeneous graphs are the relationships between entities, while the heterogeneous graph itself can also be regarded as the mapping of relationships in the real-world medical insurance dataset. Once the graph network is generated it is possible to mine the behavioral anomalies employing techniques related to graph neural networks. In this way, we consider feature attributes and fuse them to detect anomalies in conjunction with the topology of the graph network.

Nevertheless, several critical challenges must be considered to solve the above problem. First, constructing heterogeneous graphs by extracting different entities and relationships from medical insurance data. Second, finding groups with the same spatial medical trajectory, distinguishing real fraudsters from innocent ordinary patients within the groups, and identifying essential behaviors. Third, how to effectively add temporal constraints. Hence, this study proposes a medical insurance fraud detection method called StGNN that relies on a spatiotemporal constraint graph neural network. Our method includes four steps: (1) Construct a heterogeneous graph with many entities and relationships based on a medical insurance dataset. (2) Employ a multi-semantic metapath to sample groups with the same behavior trajectory. (3) Use spatial constraints on the behavioral trajectories of patients. The corresponding weights and behavioral trajectories are given to different patients according to the attention mechanism. (4) Apply temporal constraints to patients' treatment processes. Excessive comparison of our method against several baseline methods on two real medical insurance data sets proves our method's effectiveness.

In summary, StGNN presents the following contributions:

(1) Given that current methods consider only the characteristics at a feature level, the proposed method considers the feature attribute and effectively integrates the feature and the topological information.

(2) This paper designs an embedding method based on a spatiotemporal constraint graph neural network, which affords to detect gang fraud in the spatial dimension and individual anomalies in the temporal dimension.

(3) This paper exploits attention mechanisms, which can adaptively assign the weights of neighboring nodes with the same medical visit behavior and the weights of different behavioral trajectories.

2. **Related work.**

2.1. **Traditional method.** Traditional fraud detection methods can be divided into rule-based, supervised, and unsupervised learning methods. Rule-based learning relies on the expert's domain knowledge, who sets a priori thresholds for different metrics to filter anomalous data. Supervised methods can be regarded as a binary classification problem of unbalanced samples appropriate for anomaly detection. Two common solutions to the sample imbalance problem are: giving positive samples more weight and achieving sample balance through upsampling or downsampling. Unsupervised learning does not require labels and aims to find outliers within the samples by utilizing a variety of statistical, distance, and quantitative density indicators to describe the degree of estrangement between the data sample and other samples to find abnormal points with a significant deviation.

Sadiq et al. [11] developed the Patient Rule Induction Method (PRIM) based on the bump hunting method to identify the spaces of higher modes and masses that indicated the peak anomalies in the CMS 2014 dataset. Zhang et al. [12] proposed an improved LOF algorithm, considering the patient's Class A, Class B, non-basic, and payment expenses as features. To solve the drawback that the LOF algorithm cannot recognize small clusters [13], the DBSCAN method was introduced to check and adjust the anomaly score according to the cluster size. Liu et al. [14] proposed a financial fraud detection model that combined random forest for financial fraud detection, feature selection, variables' importance measurement, partial correlation analysis, and multidimensional analysis. The results showed that combining eight variables afforded the highest accuracy. Maranzato et al. [15] optimized the logistic regression model utilizing a stepwise regression scheme and used real data from a large Brazilian electronic market to compare the detection model relying on the fraud feature and the logistic regression model. The results indicated that logistic regression achieves high accuracy when the model is optimized. Yusuf et al. [16] proposed a new cost-sensitive decision tree method that minimizes the sum of the classification costs while each non-terminal node chooses to split. In this method, the misclassification cost is considered to be variable. A fraud detection system utilizing this method significantly reduces the financial losses caused by fraudulent transactions. Wang et al. [17] proposed a method for automatically searching for robust and efficient neural network structures for AIoT systems. By introducing a skip connection structure, a feature map with reduced front-end influence can be used for calculations during the classification process. Chen et al. [18] proposed an algorithm to discover potential high utility-occupancy patterns in uncertain databases. This algorithm divides user preferences into three factors, including support, probability, and utility occupancy.

Nevertheless, current methods only use the original features to train the model, ignoring the interaction between the entities in the fraud process. Hence, the model's performance largely depends on the selected features, requiring excessive feature engineering, while the behavior attributes themselves are difficult to depend on the features.

2.2. **Graph-based approach.** With the popularity of graph neural networks (GNN) [19, 20], more and more GNN methods have been applied to anomaly detection. The GNN inputs are the adjacency and the feature matrices, simultaneously considering the sample's

feature and topology information. GNNs aim to learn an informative low-dimensional vector representation for each node and then apply it to diverse downstream tasks.

Luan et al. [21] proposed the NSLPA algorithm, which improved the SLPA algorithm against the medical industry's background. In NSLPA, the initial label assignment exploits the patient's existing information, with new labeling rules based on specific medical data being formulated to suit medical data. Peng [22] developed a method for detecting irrational drug use accounted in medical insurance. This method constructs a graph network comprising diseases and drugs, uses the Bron-Kerbosch algorithm to find the maximal clique and employs the association rule algorithm for the maximal clique to obtain the most frequent itemset. Finally, by comparing the frequent itemset with the existing treatment mode, the algorithm judges whether the patient is taking medication abnormally. Li et al. [23] proposed AddGraph to detect the dynamic graphs' characteristics of abnormal users that generate false data to obtain potential benefits. AddGraph uses time windows and attention mechanisms to obtain short-term features. The GCN model obtains long-term features and finally uses GRU to merge short-term and long-term features to check false edges. Dou et al. [24] proposed the CARE-GNN model. Aiming at the feature and relationship camouflage of fraudsters in the graph, the authors proposed a similarity measure for label perception and designed a similarity Degree neighbor selector to select similar neighbors of the center node. In addition, reinforcement learning is used to find the optimal neighborhood threshold adaptively.

Since the heterogeneous graph is a natural manifestation of the relationship between different entities, it is more flexible in complex relationship analysis and can better discover the abnormal relationship between entities, providing a new direction for fraud detection research. The node embedding learned by graph network technology on the medical insurance heterogeneous graph contains both feature and topological information enhancing its robustness for downstream tasks compared to solely utilizing feature information.

3. **Methods.**

The overall StGNN structure is illustrated in Figure 1. First, we build multiple heterogeneous graphs G based on medical insurance data of different periods. Then we employ multi-semantic metapaths to sample groups with the same behavior trajectory on the heterogeneous graphs and use spatial constraint aggregation group information. Then we use temporal constraints on the entire medical time axis to obtain the final embedding representation X of the node. X has a better expressive ability than the original node feature due to fusing temporal and spatial information. Finally, the model passes through a multilayer perceptron and is optimized using cross-entropy loss. The following subsections introduce in detail how the model operates. The proposed method can be divided into four steps:

(1) Constructing a heterogeneous graph with many entities and relationships based on the medical insurance dataset.

(2) Using a multi-semantic metapath to sample groups with the same behavior trajectory.

(3) Employing spatial constraints on the patients' behavioral trajectories and assigning weights to different patients within the group and different behavioral trajectories through the attention mechanism.

(4) Utilizing temporal constraints on the patients' treatment process.

3.1. **Construct heterogeneous graphs.**

The heterogeneous medical insurance graph is illustrated in Figure 2. The medical insurance dataset involves millions of patient transaction records, which we transform into a heterogeneous medical insurance graph to understand better the patients' behavior. Precisely, we extract for the selected patients

all visit records and construct four node types: patient, hospital department, date, and medicine. For further spatial refinement, we treat hospitals and departments as a whole, i.e., departments of the same name in different hospitals are treated as different nodes. Date nodes are refined to days, and considering the medicine nodes, we eliminate nodes with drug unit prices less than ¥20 to avoid the graph being too dense.

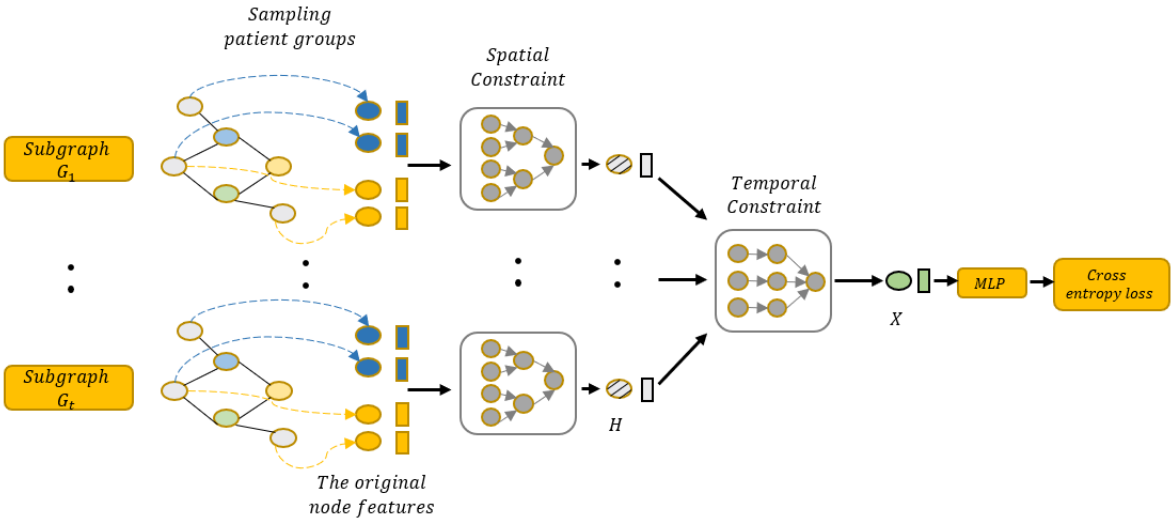


FIGURE 1. The overall architecture of StGNN.

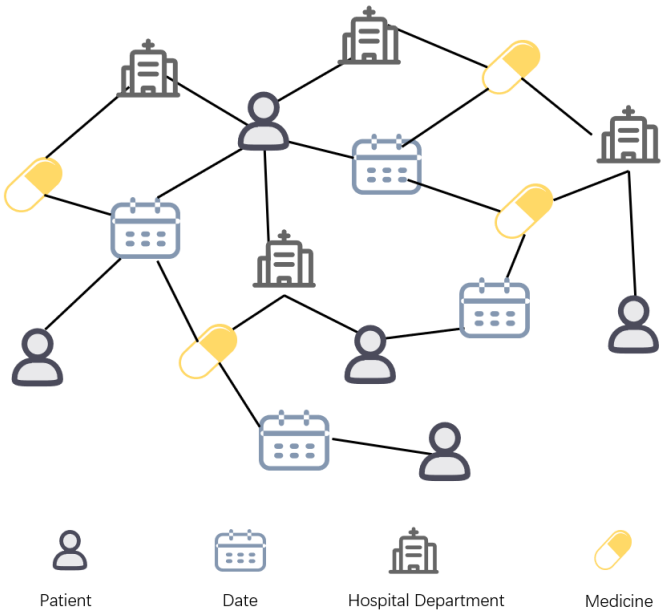


FIGURE 2. Medical insurance networks.

3.2. **Sampling patient groups based on multi-semantic metapath.** Figure 3 highlights that after obtaining the medical insurance heterogeneous graph, we design multi-semantic metapaths to sample neighbors from the heterogeneous graph and obtain different patient groups. For ease of illustration, we only show four types of nodes in the heterogeneous graph: patient, fraudster, hospital department, and medicine. As illustrated in Figure 3(a), patients P1, P2, and P3 were prescribed medicine M at hospital

H1, and patients P4 and P5 were also prescribed medicine M but from hospital H2. The multi-semantic metapath PHMHP (patient-hospital department-medicine-hospital department-patient) is presented in Figure 3(b). Sampling neighbors using the multi-semantic metapath PHMHP starts from the patient node in the heterogeneous graph, travels among different node types according to the metapath order, and returns to the patient node. For example, starting from fraudster P2, passing through hospital H1, medicine M, then hospital H1, and finally returning to fraudster P3. By analogy, we get the two groups depicted in Figure 3(c). The semantic information of the multi-semantic metapath PHMHP can be understood as the patients prescribed the same drugs in the same hospital department. The patient groups eventually obtained are all groups with the same behavioral attributes in the spatial dimension.

Due to the nodes' heterogeneity, different node types belong in different feature spaces in the medical insurance heterogeneous graph. Although a one-hot encoding scheme can construct the features of dates, drugs, and hospital departments and then use the weight matrix to map them into a unified feature space, due to the characteristics of the heterogeneous graphs, forcibly integrating different node types may cause "incompatibility" between the features. Thus, we construct the medical insurance heterogeneous graph to reflect the behavioral attributes through the topological structure and learn helpful information, while the medical insurance fraud only considers the patient nodes. Therefore, we only need to exploit the patient's features without employing the features of other node types within the heterogeneous graph. Using multi-semantic metapaths can transform complex medical insurance heterogeneous graphs into homogeneous patient graphs containing only the critical nodes and simplify the calculations while preserving the topological information of the original heterogeneous graph as much as possible.

Multiple metapaths can be used simultaneously, and the semantic information contained in each metapath can be regarded as a behavioral attribute in the spatial dimension. In addition to PHMHP, there can also be PDMDP (patient-date-medicine-date-patient), PHP (patient-hospital department-patient), or PDP (patient-medicine-patient) metapaths. Among them, PDMDP refers to patients who were prescribed the same drug on the same date, PHP refers to patients treated in the same hospital department, and PMP refers to patients who use the same medicine. The reason for using the multi-semantic metapath is because it contains more information. For example, compared with the single-semantic metapath, such as PMP and PHP, the extra part of the relationship between the department and the medicine in PHMHP provides finer constraints assisting in discovering the disorderly prescripted drugs in departments. Thus, multiple metapaths can capture groups of the same behavior trajectory from multiple aspects.

3.3. **Spatial constraint.** Combining the features of neighboring nodes within the group is the most effective way to obtain the behavioral attributes in the spatial dimension. This is because the members of each group have similar medical behavior trajectories, which can provide better mining information from the perspective of the entire group. The process of combining the features of neighboring nodes is named aggregation. Through the aggregation operation, the target node can integrate the feature of the entire group.

The entire spatial-constraint process is illustrated in Figure 4 below, highlighting that first, multiple multi-semantic metapaths are used to sample neighbors, and then the node features in each group are aggregated to the target node. During node-level aggregation, we may encounter a special situation depicted in Figure 3(c), where not all fraud patients' neighbors within a group with the same behavior trajectory are fraud patients. In general, the neighbors of fraud patients are more likely to be fraud patients, and the neighbors of

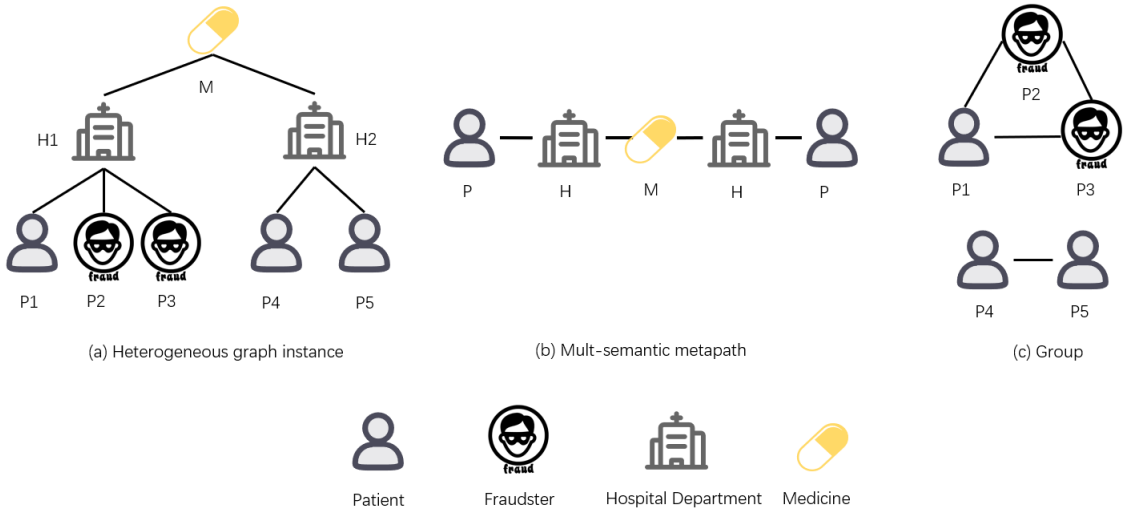


FIGURE 3. Neighbor sampling process.

ordinary patients are more likely to be ordinary patients. Thus, aggregating the neighbors directly reduces the node representation's performance due to noise. Therefore, we use the attention mechanism to assign different weights to different node neighbors. Finally, we aggregate the target node representation through semantic level aggregation. Since, at this time, the target node has no neighbors with the same behavior trajectory after node-level aggregation, we group the target nodes of other behavior trajectories to achieve aggregation. In this way, the target node representation contains information with different semantics. Similarly, we will also use an attention mechanism to measure the importance of different behaviors.

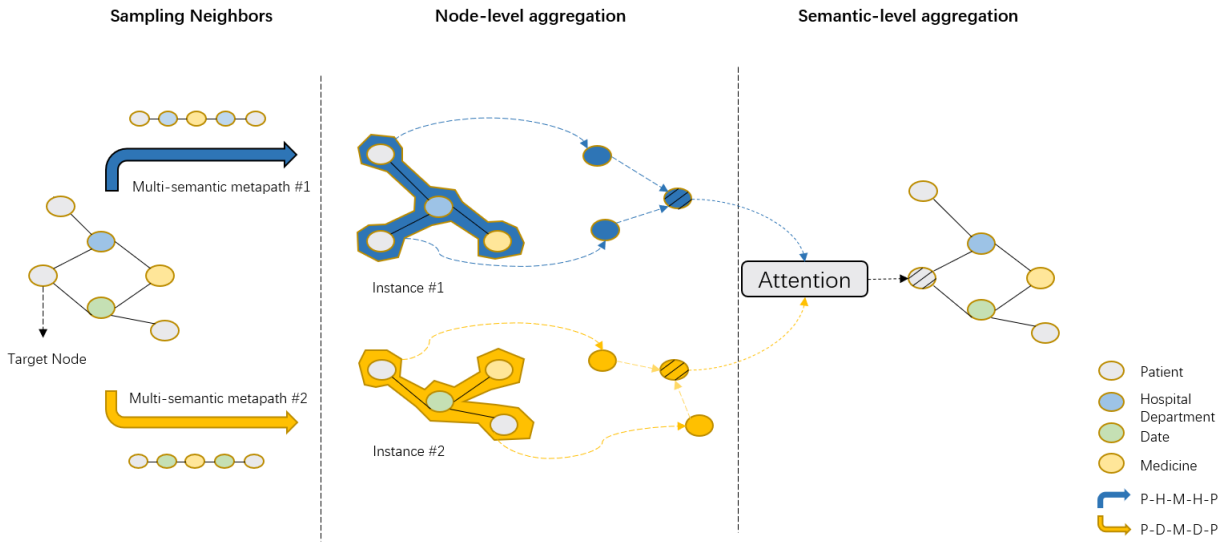


FIGURE 4. The spatial constraint process.

3.3.1. *Node level aggregation.* After sampling a node's neighbors through the multi-semantic metapath, the new patient's node representation is aggregated from these neighboring nodes. However, each neighbor node contributes differently and cannot be treated equally.

For example, the proportion of gangs that defraud patients should be relatively large, while the proportion of general patients is relatively small.

Given a node pair (i, j) connected via multi-semantic metapath P, the attention cross-correlation coefficient e_{ij}^p is presented in Eq. (1), representing the importance of node j to i. Here h_i and h_j denote the original features of node i and j, $W \in R^{F' \times F}$ is the mapping matrix, $attnode \in R^{2F'}$ refers to the deep neural network performing the node-level attention, and $||$ is the concatenate operation. Since the weights between the node pairs depend on their characteristics, different neighbors can be assigned different importance.

$$\dot{e}_{ij}^p = attnode\,[Wh_i||Wh_j] \tag{1}$$

After obtaining the mutual attention coefficient e_{ij}^p, we employ a LeakyReLU activation function, then use the softmax function to normalize all neighbors, and finally, we obtain the normalized attention weight a_{ij}^p. The latter denotes the weight of neighbor node j among all neighbors of node i, with the specific details of a_{ij}^p presented in Eq. (2). Among them, N_i^P denotes the set of all neighbor nodes of node i.

$$\dot{a}_{ij}^p = \frac{\exp(LeakyReLU\left(e_{ij}^p\right))}{\sum_{s \in N_i^P} \exp(LeakyReLU\left(e_{is}^p\right))} \tag{2}$$

Finally, the node-level embedding representation h_i^P of node i based on the multi-semantic metapath P is shown in Eq. (3), where $\sigma(\cdot)$ denotes the activation function. Since each node is aggregated by its neighbors, its embedding representation can capture well the behavior of the entire group.

$$\dot{h}_i^p = \sigma\left(\sum_{j \in N_i^P} a_{ij}^p \cdot hj\right) \tag{3}$$

This attention mechanism can also be extended to multiple heads, stabilizing the learning process, i.e., we execute K independent attention mechanisms and then concatenate their outputs, resulting in the following formulation:

$$\dot{h}_i^p = \mathop{\Big\|}_{k=1}^{K} \sigma\left(\sum_{j \in N_i^P} a_{ij}^p \cdot hj\right) \tag{4}$$

3.3.2. *Semantic level aggregation.* Since different multi-semantic metapaths can represent different semantic information, we often use multiple multi-semantic metapaths to extract more information from different aspects of the heterogeneous graphs. Suppose we have m multi-semantic metapaths $P_0, P_1, \ldots, P_m$, after the previous node-level aggregation, we obtain m semantic-level embedding representations $H_0, H_1, \ldots, H_m$. Similar to the node neighbors, the semantic level's embedding representation importance obtained by the node from different multi-semantic metapath is also different. Thus, to comprehensively obtain the nodes' embedding representation, we need to integrate embedding representations of different semantic levels and distinguish the importance of different multi-semantic metapaths. Therefore, the semantic level aggregation will also introduce an attention mechanism, which will automatically learn the weights of different multi-semantic metapaths.

The node vector that has undergone node-level aggregation is nonlinearly transformed, and then the transformed vector is multiplied with the trainable semantic-level attention vector q to obtain the importance of the semantic-level embedding representation. The importance of each metapath S_{Pi} is given by:

$$\dot{S}_{Pi} = \frac{1}{|V|}\sum_{i\in V} q^T \cdot \tanh(W \cdot h_i^p + b) \tag{5}$$

After obtaining the importance of each metapath, it is normalized by the softmax function and the attention weight β_{Pi} of each multi-semantic metapath is obtained:

$$\dot{\beta}_{Pi} = \frac{\exp(S_{Pi})}{\sum_{i=1}^{m} \exp(S_{Pi})} \tag{6}$$

Then, the attention weight of each multi-semantic metapath and the node-level embedding representation are weighted and summed to obtain the semantic-level embedding representation h_i^{pm}. The larger the attention weight corresponding to the multi-semantic metapath, the closer the final semantic level embedding representation.

$$\dot{h}_i^{Pm} = \sum_{i=1}^{m} \beta_{Pi} \cdot h_i^p \tag{7}$$

3.4. **Temporal constraint.** Spatial constraints use behavior trajectories to mine information from the group's perspective. After performing node-level and semantic-level aggregation, the new node representation contains various behavior attributes. Then we employ temporal constraints to identify abnormalities on the entire timeline of the medical treatment from the perspective of individuals. In order to construct time-series data, we divide the heterogeneous graphs into T subgraphs, where each subgraph G_t represents the heterogeneous graph constructed according to the medical records in the corresponding time. In this way, after spatial constraints are applied to each subgraph, a node representation is obtained, and all target node representations are serially connected to obtain time-series data.

We use Bi-LSTM to mine the anomalies on the individual time axis. This strategy combines contextual relations for bilateral reasoning to capture deep-level feature interactions and obtain more powerful embedding expression capabilities. The entire temporal constraint architecture is illustrated in Figure 5. Specifically, we input the output of each subgraph's target node into the Bi-LSTM model and splice the forward and backward propagation results. The final target node embedding representation x' is obtained through the mean pooling layer:

$$\dot{x}^{'} = \frac{\sum_{t\in T}\left[\overrightarrow{LSTM}\left(x_t\right) \bigoplus \overleftarrow{LSTM}\left(x_t\right)\right]}{|T|} \tag{8}$$

where T denotes the set of all subgraphs, x_t is the embedding representation obtained by spatial aggregation in the t subgraph, $\oplus$ denotes the splicing operation, and the $LSTM$ is defined as:

$$
\begin{aligned}
\dot{z}_t &= \sigma\left(U_z x_t + W_z h_{t-1} + b_z\right) \\
f_t &= \sigma\left(U_f x_t + W_f h_{t-1} + b_f\right) \\
o_t &= \sigma\left(U_o x_t + W_o h_{t-1} + b_o\right) \\
\hat{c}_t &= \tanh(U_c x_t + W_c h_{t-1} + b_c) \\
c_t &= f_t \circ c_{t-1} + z_t \circ \hat{c}_t \\
h_t &= \tanh(c_t) \circ o_t
\end{aligned} \tag{9}
$$

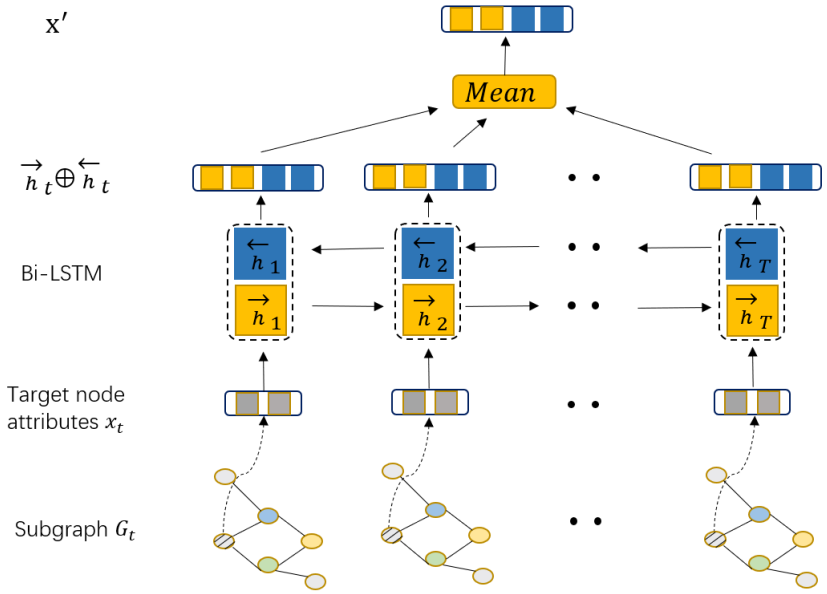


FIGURE 5. The temporal constraint process.

4. **Experiments.** This section demonstrates the StGNN's efficiency through several experiments aiming to address the following research questions:

RQ1. How does StGNN perform in node classification?
RQ2. How does StGNN perform in anomaly detection?
RQ3. What is the impact of the various components of StGNN?
RQ4. How sensitive is the StGNN model to parameter changes?
RQ5. How to understand the representation capabilities of different graph embedding methods?

4.1. **Dataset.** We utilized two real datasets of a particular municipal medical insurance bureau in 2018, involving more than 10,000 patients and more than 3 million detailed records. The fraud samples in the Medical-1 and Medical-2 datasets are the same, considering abnormal patients detected by methods such as abnormal kidney disease, repeated medication, and simultaneous outpatient hospitalization. The difference is that Medical-1 considers a balanced sample with the positive to negative sample ratio being 1:2, and medical-2 is an unbalanced sample with the corresponding sample ratio being 1:70. Further information on both datasets is presented in Table 1.

Table 1. Statistics of datasets.

Dataset	Number of nodes	Positive:negative	Metapath
Medical-1	Patient:440 Hospital departments:708 Date:351 Medicine:2328	152:288	Single-semantic: PHP,PDP,PMP Multi-semantic: PHMHP,PDMDP
Medical-2	Patient:10647 Hospital departments:2751 Date:364 Medicine:4718	152:10495	Single-semantic: PHP,PDP,PMP Multi-semantic: PHMHP,PDMDP

4.2. **Baseline. LR** [25] This is a generalized linear regression analysis model, which assumes that the data obey a distribution and then uses a maximum likelihood estimation scheme to estimate the parameters.

RF [26] This is a classifier containing multiple decision trees, where a random forest based architecture integrates all classification voting results and designates the category with the most votes as the final output.

GCN [27] This is an extensible method for semi-supervised learning on graph structure data. It is based on a convolutional neural network that performs convolution operations on structured graph data. It aggregates neighbors through a local first-order approximation of spectral graph convolution. Since this model is used for homogeneous graphs, we sample every semantic metapath and report the best result.

GAT [28] This GNN for homogeneous graphs uses an attention mechanism to effectively aggregate neighbors to obtain node embeddings. Similar to GCN, we test GAT on homogeneous graphs based on a single-semantic metapath and report the results of the best metapath.

HAN [29] This is a heterogeneous graph neural network based on hierarchical attention that includes node-level and semantic-level attention and fully considers the importance of nodes and metapaths.

Metapath2vec [30] This is a traditional heterogeneous model, which generates node embeddings by providing random walk sequences to the skip-gram model guided by metapath.

4.3. **Feature engineering.** Feature engineering includes two parts: local and global features. To construct a time window per person, the local features are divided into 7 days, 15 days, and 30 days. Specifically, we select the time window in which the person spends the most time and count the maximum, minimum, and average values of various expenses within the window: nursing, examination, radiation, chemotherapy, treatment, blood transfusion, surgery, examination, bed, meals, and other expenses, along with registration expenses, western medicine, and traditional Chinese medicine. Global features include the average interval between visits, the proportion of multiple hospitals, total personal expenses, total personal fund expenses, and the number of personal visits.

4.4. **Reproducibility.** For the LR, Xgboost, and RF models, the weights are set according to the proportion of positive and negative samples. Among them, the LR model adds the L2 regular term, the Xgboost and RF models set the number of decision trees to 256, and the maximum depth of the tree is set to six. For the Metapath2vec model, we conducted a random walk for the patient nodes in the order of patient - medicine – patient - date – patient - hospital department - patient. Each patient node is randomly sampled 20 times, and the number of negative samples is set to five. For all graph-based

models, including GAT, GCN, HAN, and StGNN, we set the number of attention heads to eight, dropout to 0.6, weight decay to 0.001, and employed the Adam optimizer. The learning rate of our model on the Medical-1 dataset is set to 0.005. For Medical-2, it is set to 0.003, and the subgraphs are divided according to months. Regarding the Bi-LSTM temporal constraints, we set the dimension of the final embedding to 256. We train the GNNs for 100 epochs and apply early stopping with a patience of 20.

4.5. **Node classification.** The experiments are conducted on the Medical-1 dataset, as it guarantees that the diseases, departments, and amounts of the positive and negative samples are the same and afford to compare the performance of different models on the node classification task. The dataset is divided into training, verification, and test set, considering three scenarios. The verification indicators employed are the F1-score, accuracy, and recall. We use an end-to-end training method for all models by connecting at the end of each model an MLP layer and utilizing a cross-entropy loss function to optimize the model.

Table 2 highlights that all StGNN's performance indicators are better than the competitor baseline methods regardless of the training ratio and that our model achieves its peak performance for a training set ratio of 20% and 40%. The overall effect of Metapath2vec is quite different from the other models, inferring that solely employing structural information in a small number of samples cannot learn a better embedding representation. The three models of GAT, GCN, and HAN are all groups that use metapaths to find the same behavior trajectory in the spatial dimension. Compared with them, StGNN has additional temporal constraints affording anomaly detection in the temporal dimension and achieving a 6% higher F1-score on average. This demonstrates that solely aggregating neighbors in the spatial dimension is not adequate, as the temporal dimension is also significant.

Table 2. Experiment results on the Medical-1 datasets for the node classification task.

Train:Val:Test	Metrics	Metapath2vec	GAT	GCN	HAN	StGNN
1:1:3	f1	0.5595	0.7979	0.7914	0.791	**0.8247**
	acc	0.7196	0.8523	0.8523	0.8598	**0.8712**
	recall	0.5222	0.8556	0.8222	0.7778	**0.8696**
2:1:2	f1	0.5964	0.7273	0.7547	0.7455	**0.8167**
	acc	0.7386	0.8295	0.8523	0.8409	**0.875**
	recall	0.68	0.8	0.8	0.82	**0.8305**
3:1:1	f1	0.6349	0.72	0.7772	0.7407	**0.8615**
	acc	0.7386	0.8409	0.8371	0.8409	**0.8977**
	recall	0.7692	0.6923	0.8333	0.7692	**0.875**

4.6. **Anomaly detection.** In the subsequent trials, we use the Medical-2 dataset that simulates a real medical environment appropriate to verify the performance of different models on node anomaly detection tasks. We use the weighted cross-entropy loss function for training and finally obtain the embedding representation of the model under different training sets. In order to verify the representation learning capabilities of different graph neural network models, after obtaining the low-dimensional embedding representation, we use a random forest classifier to verify the performance. For fairness, the random forest uses 256 trees uniformly, and the class weight is set to balance the subsamples.

As shown in Table 3, the effect of GCN is significantly reduced on unbalanced samples. This is because GCN treats all neighbors equally when aggregating neighbors, and since ordinary patients account for a large proportion, GCN suffers from noise during aggregation. Hence, the representation learned by GCN underperforms on anomaly detection tasks. As the training set increases, the effects of logistic regression and random forest improve. This reveals that the more labeled data, the better the supervised model's effect. Moreover, StGNN and HAN introduce an attention mechanism, allowing nodes to distinguish the neighbors' importance. StGNN also introduces temporal constraints based on HAN, achieving a 7% higher F1-score on average. The embedding representation of StGNN integrates structural and feature information. Compared with the effect of the original feature on random forest, the performance of the StGNN model is highly improved.

TABLE 3. Experiment results on the Medical-2 datasets for the node anomaly detection task.

Training size	Metrics	LR	RF	GCN	HAN	StGNN
20%	f1	0.7733	0.7517	0.5816	0.7708	**0.8519**
	auc	0.8256	0.8143	0.7295	0.8315	**0.8873**
40%	f1	0.819	0.8269	0.4396	0.75	**0.8319**
	auc	0.8581	0.8582	0.6654	0.8244	**0.891**
60%	f1	0.8387	0.8197	0.549	0.8	**0.875**
	auc	0.8712	0.8569	0.6995	0.8426	**0.8998**

4.7. **Ablation experiments.** This section evaluates several StGNN variants, specifically StGNN-LR, StGNN-RF, and StGNN-Xgb, revealing the effectiveness of StGNN embedding logistic regression, random forest, and Xgboost classifiers, respectively. StGNN-S and StGNN-M sample neighbors using only single semantic metapath and multi-semantic metapath, respectively.

The performance comparison of the representations learned by the StGNN model under different classifiers is illustrated in Figure 6. The abscissa represents the model's name, and the ordinate is the model's performance, while the performance indicators are the F1-score, recall, and precision. The following plots report the results utilizing a training set of 20%, 40%, and 60% of the entire Medical-2 dataset. Figure 6 highlights that logistic regression and the random forest perform slightly better than Xgboost, but random forest and Xgboost are much better than logistic regression in terms of stability. This is because logistic regression is relatively simple, while random forest and Xgboost use an integrated method, which is more robust than logistic regression. Nevertheless, the embedding representation learned by our model contains structural and feature information, which can be better applied to downstream tasks than the original features.

The results comparing different metapaths by StGNN are depicted in Figure 7. The abscissa represents the performance index, and the ordinate represents the model's performance. The results utilize a training set of 20%, 40%, and 60% of the entire Medical-1 dataset and reveal that using a multi-semantic metapath is better than using a single-semantic metapath because the multi-semantic metapath contains more information about the intermediate nodes than the single-semantic metapath. For example, compared with the single-semantic metapath PHP and PMP, the multi-semantic metapath PHMHP has more relations between departments and drugs, affording a better embedding representation than a single semantic metapath.

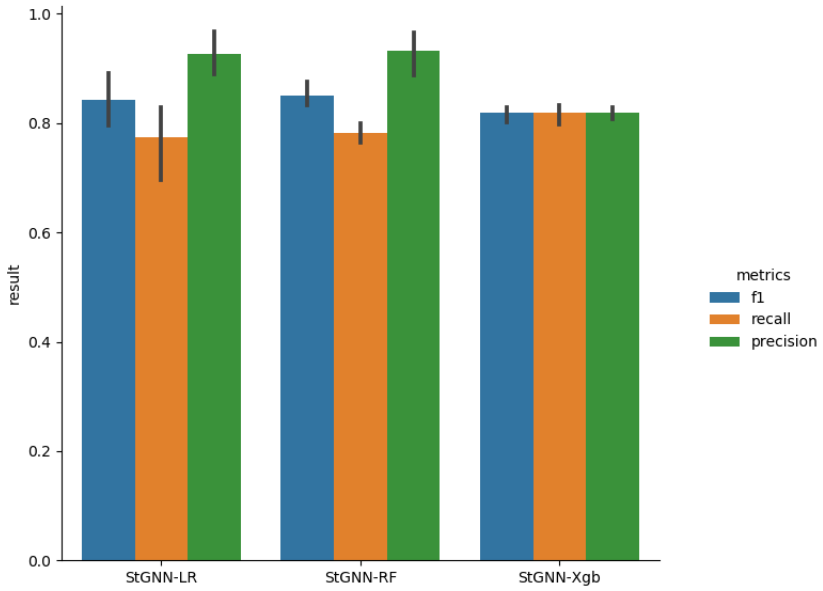


FIGURE 6. The results of the characterization learning experiment of the StGNN model.

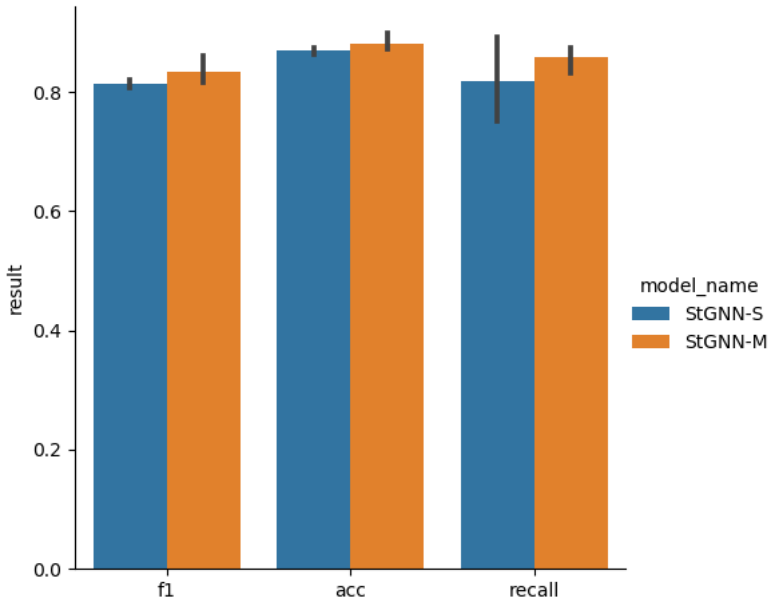


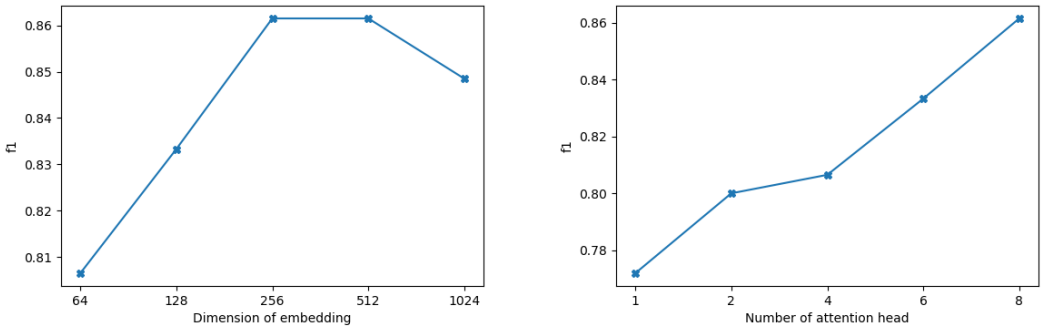
FIGURE 7. The results of the metapath comparison experiment of the StGNN model.

4.8. **Parameter analysis.** This section investigates the sensitivity of parameters and reports the classification results (F1-score) on the Medical-1 dataset with various parameters in Figure 8. We uniformly divide the data into 60% training set, 20% validation set, and 20% test set, and finally, we report the results on the test set.

We first test the effect of the dimension of the final embedding, with the corresponding results illustrated in Figure 8(a). We can see that with the growth of the embedding dimension, the performance raises first and then drops slowly. The effect is worst when

the embedding dimension is 64 due to insufficient information. When the embedding dimension reaches 256, the representation already contains enough information and achieves the best result. It is worth noting that although the effect of the 512-dimensional features is comparable to the 256-dimensional features, the training time is exponentially increased. Moreover, larger dimensions may introduce additional redundancy that can lead to performance degradation.

In order to check the impact of multi-head attention, we explore the performance of StGNN with various numbers of attention heads. The results are depicted in Figure 8(b), where multi-head attention can be understood as using the attention mechanism multiple times in one training session. Note that the multi-head attention is removed when the attention head is set to 1. Based on the results, we conclude that the more the attention heads, the more stable the training process, and the higher the performance of StGNN.



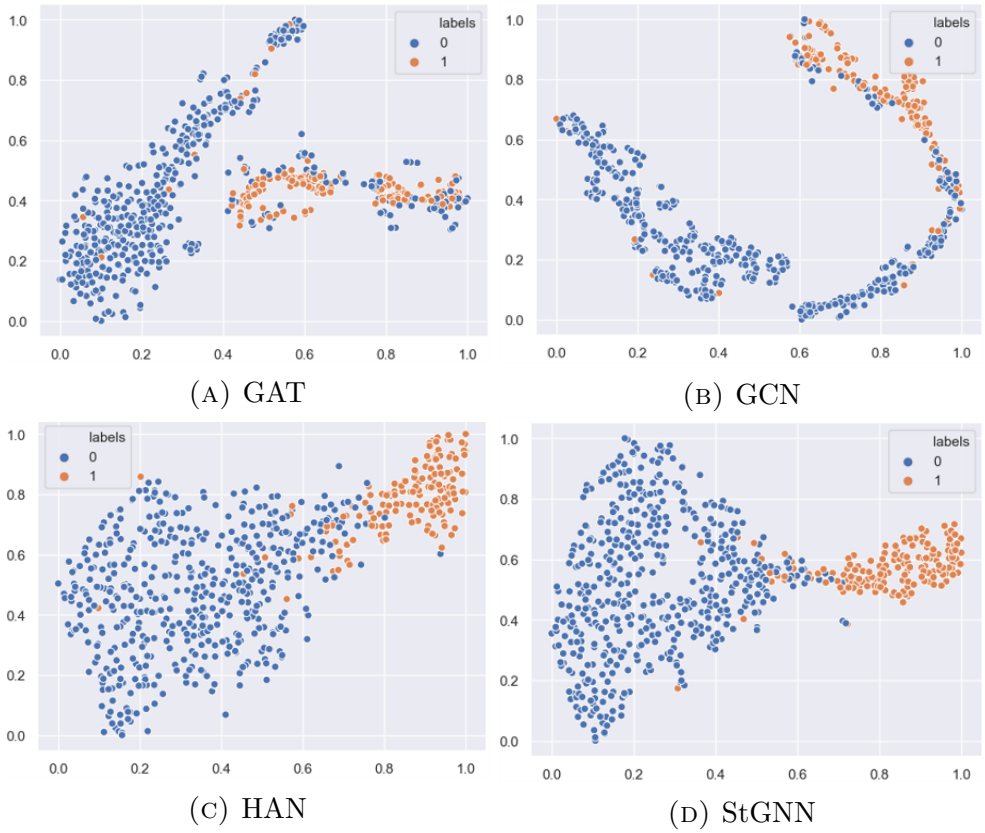
(A) Dimension of the final embedding (B) Number of attention head

FIGURE 8. Parameter sensitivity of StGNN w.r.t. Dimension of the final embedding and Number of attention head.

4.9. **Visualization.** In order to intuitively compare the embedding representation capabilities learned by different GNN models, we perform a visualization task, which represents the nodes' distribution in a low-dimensional space. We learn the node embedding of each model on the Medical-2 dataset and use the t-sne method to project the learned embedding onto a two-dimensional space. In order to facilitate observation, we consider all positive samples and randomly select 500 nodes from the negative samples.

As shown in Figure 9, the embedding representation ability learned by the GAT and GCN methods is not appealing. This is because a considerable number of positive and negative samples are mixed, and thus HAN has a better effect than GAT and GCN, as the HAN model employs multiple metapaths to obtain more semantic information. Finally, the proposed StGNN presents the best effect, better separating different classes. Compared to HAN, StGNN's intra-class distance is small, and the inter-class distance is considerable, proving that StGNN can learn a good node representation, further illustrating the effectiveness of spatiotemporal constraints.

5. **Conclusion and future work.** Aiming at the problem of medical insurance fraud, this paper proposes a graph neural network based on Spatio-temporal constraints. The proposed StGNN can utilize multi-semantic metapaths to capture the behavioral attributes behind the medical insurance heterogeneous graph and aggregate neighbor nodes and different behavioral attributes through an attention mechanism. Finally, the spatial-temporal constraints capture group anomalies from the spatial dimension and individual anomalies from the temporal dimension. Experiments on two existing real data sets



(A) GAT (B) GCN

(C) HAN (D) StGNN

FIGURE 9. Visualization of the learned node embeddings on the Medical-2 dataset.

demonstrate the StGNN's superiority over current methods. Future work shall improve StGNN in three aspects: (1) When aggregating neighbors, it can first evaluate the similarity between nodes and neighbors to filter out neighbors with low similarity, improving noise resistance. (2) We divided the subgraphs according to the month in this work. However, targeting a single disease and dividing it according to its treatment cycle will improve the model's performance. (3) We divided the subgraph to construct time-series data and mined individual abnormalities in this paper. Nevertheless, starting from the perspective of a dynamic graph, capturing the structural change information of the graph is also possible.

Acknowledgment. This research was funded by the Science Foundation of Fujian Province (No.2021J011188, 2019Y0057), the Xiamen Science and Technology Planning Project (No.3502Z20206073), the Research Project of Xiamen Overseas Students (No.XRS202017206), the Open Fund of Key Laboratory of Ecological Environment and Information Atlas, Fujian Provincial University (No.ST18004), the Shenzhen Fundamental Research Program under Grant No.JCYJ20190809161603551, and the XMUT Scientific Research Project (No.YSK20002R, No.YKJCX2020120).

REFERENCES

[1] J. Sun, and Y. Y. Gan, Medical insurance anti-fraud mechanism from the perspective of cooperative governance: International experience and implications, *Chinese Journal of Health Policy*, vol. 10, no. 10, pp. 28–34, 2017.

[2] B. He, F. Huang, and X. Zheng, Research on the status quo, causes and countermeasures of my country's medical insurance fraud, *Shanghai Insurance*, vol. 37, no. 6, pp. 51–53, 2020.

[3] R. Chen, T. Chen, Y. Chien, and Y. Yang, Novel questionnaire-responded transaction approach with SVM for credit card fraud detection, *International Symposium on Neural Networks*, pp. 916—921, Springer, 2005.
[4] V. Vaishali, Fraud detection in credit card by clustering approach, *International Journal of Computer Applications*, vol. 98, no. 3, pp. 29–32, 2014.
[5] D. Bo, X. Wang, C. Shi, M. Zhu, E. Lu, and P. Cui, Structural deep clustering network, *Proceedings of The Web Conference 2020*, pp. 1400—1410, 2020.
[6] A. Grover and J. Leskovec, node2vec: Scalable feature learning for networks, *Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 855—864, 2016.
[7] W. L. Hamilton, R. Ying, and J. Leskovec, Inductive representation learning on large graphs, *arXiv preprint arXiv:1706.02216*, 2017.
[8] J. Atwood and D. Towsley, Diffusion-convolutional neural networks, *Advances in neural information processing systems*, pp. 1993—2001, 2016.
[9] D. Wang, P. Cui, and W. Zhu, Structural deep network embedding, *Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 1225—1234, 2016.
[10] C. Zhang, D. Song, C. Huang, A. Swami, and N. V. Chawla, Heterogeneous graph neural network, *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery Data Mining*, pp. 793—803, 2019.
[11] S. Sadiq, Y. Tao, Y. Yan, and M.-L. Shyu, Mining anomalies in medicare big data using patient rule induction method, *2017 IEEE third international conference on multimedia Big Data (BigMM)*, pp. 185—192, 2017.
[12] W. J. Zhang, Outlier Detection based Medicare Anomalous Data Mining, *East China Normal University*, 2018.
[13] M. M. Breunig, H.-P. Kriegel, R. T. Ng, and J. Sander, Lof: identifying density-based local outliers, *Proceedings of the 2000 ACM SIGMOD international conference on Management of data*, pp. 93–104, 2000.
[14] C. Liu, Y. Chan, S. H. Alam Kazmi, and H. Fu, Financial fraud detection model: Based on random forest, *International journal of economics and finance*, vol. 7, no. 7, pp. 178–188, 2015.
[15] R. Maranzato, A. Pereira, A. P. do Lago, and M. Neubert, Fraud detection in reputation systems in e-markets using logistic regression, *Proceedings of the 2010 ACM symposium on applied computing*, pp. 1454–1455, 2010.
[16] Y. Sahin, S. Bulkan, and E. Duman, A cost-sensitive decision tree approach for fraud detection, *Expert Systems with Applications*, vol. 40, no. 15, pp. 5916–5923, 2013.
[17] K. Wang, P. Xu, C.-M. Chen, S. Kumari, M. Shojafar, and M. Alazab, Neural architecture search for robust networks in 6g-enabled massive iot domain, *IEEE Internet of Things Journal*, vol. 8, no. 7, pp. 5332–5339, 2020.
[18] C.-M. Chen, L. Chen, W. Gan, L. Qiu, and W. Ding, Discovering high utility-occupancy patterns from uncertain data, *Information Sciences*, vol. 546, pp. 1208–1229, 2021.
[19] X. Fu, J. Zhang, Z. Meng, and I. King, Magnn: Metapath aggregated graph neural network for heterogeneous graph embedding, *Proceedings of The Web Conference 2020*, pp. 2331–2341, 2020.
[20] J. Zhao, X. Wang, C. Shi, Z. Liu, and Y. Ye, Network schema preserved heterogeneous information network embedding, *29th International Joint Conference on Artificial Intelligence (IJCAI)*, 2020.
[21] T. Luan, Z. Yan, S. Zhang, and Y. Zheng, Fraudster detection based on label propagation algorithm, *2018 IEEE 20th International Conference on High Performance Computing and Communications; IEEE 16th International Conference on Smart City; IEEE 4th International Conference on Data Science and Systems (HPCC/SmartCity/DSS)*, pp. 346–353, 2018.
[22] J. F. Peng, Research on the method of irrational drug use detection in medical insurance, *Shandong University*, 2018.
[23] L. Zheng, Z. Li, J. Li, Z. Li, and J. Gao, Addgraph: Anomaly detection in dynamic graph using attention-based temporal gcn, *IJCAI*, pp. 4419–4425, 2019.
[24] Y. Dou, Z. Liu, L. Sun, Y. Deng, H. Peng, and P. S. Yu, Enhancing graph neural network-based fraud detectors against camouflaged fraudsters, *Proceedings of the 29th ACM International Conference on Information Knowledge Management*, pp. 315–324, 2020.
[25] S. Menard, Logistic Regression, *American Statistician*, vol. 58, no. 4, pp. 364, 2004.
[26] L. Breiman, Random forests, *Machine learning*, vol. 45, no. 1, pp. 5–32, 2001.
[27] T. N. Kipf and M. Welling, Semi-supervised classification with graph convolutional networks, *arXiv preprint arXiv:1609.02907*, 2016.

[28] P. Velikovi, G. Cucurull, A. Casanova, A. Romero, P. Lio, and Y. Bengio, Graph attention networks, *arXiv preprint arXiv:1710.10903*, 2017.

[29] X. Wang, H. Ji, C. Shi, B. Wang, Y. Ye, P. Cui, and P. S. Yu, Heterogeneous graph attention network, *The World Wide Web Conference*, pp. 2022–2032, 2019.

[30] Y. Dong, N. V. Chawla, and A. Swami, metapath2vec: Scalable representation learning for heterogeneous networks, *Proceedings of the 23rd ACM SIGKDD international conference on knowledge discovery and data mining*, pp. 135–144, 2017.