Semi-supervised Credit Card Fraud Detection via Attribute-Driven Graph Representation

Sheng Xiang[1], **Mingzhi Zhu**[2], **Dawei Cheng**[2,3*], **Enxia Li**[1],
Ruihui Zhao[4], **Yi Ouyang**[4], **Ling Chen**[1], **Yefeng Zheng**[4]
[1]Australian Artificial Intelligence Institute, University of Technology Sydney, Sydney, Australia
[2]Department of Computer Science and Technology, Tongji University, Shanghai, China
[3]Shanghai Artificial Intelligence Laboratory, Shanghai, China
[4]Tencent Jarvis Laboratory, Shenzhen, China
{sheng.xiang, ling.chen}@uts.edu.au, mz3379@nyu.edu, dcheng@tongji.edu.cn,
enxia.li@student.uts.edu.au, zachary@ruri.waseda.jp, {yiouyang, yefengzheng}@tencent.com

Abstract

Credit card fraud incurs a considerable cost for both cardholders and issuing banks. Contemporary methods apply machine learning-based classifiers to detect fraudulent behavior from labeled transaction records. But labeled data are usually a small proportion of billions of real transactions due to expensive labeling costs, which implies that they do not well exploit many natural features from unlabeled data. Therefore, we propose a semi-supervised graph neural network for fraud detection. Specifically, we leverage transaction records to construct a temporal transaction graph, which is composed of temporal transactions (nodes) and interactions (edges) among them. Then we pass messages among the nodes through a Gated Temporal Attention Network (GTAN) to learn the transaction representation. We further model the fraud patterns through risk propagation among transactions. The extensive experiments are conducted on a real-world transaction dataset and two publicly available fraud detection datasets. The result shows that our proposed method, namely GTAN, outperforms other state-of-the-art baselines on three fraud detection datasets. Semi-supervised experiments demonstrate the excellent fraud detection performance of our model with only a tiny proportion of labeled data.

Introduction

The great losses caused by financial fraud have attracted continuous attention from academia, industry, and regulatory agencies. For instance, as reported in (AlFalahi and Nobanee 2019; Máté et al. 2019), financial fraud detection plays a critical role to support the sustainable economic growth. However, fraudulent behaviors against online payments, such as illegal card swiping, have caused property losses to online payment users (Bhattacharyya et al. 2011b).

An important line of research in financial fraud detection is credit card fraud detection, where credit card fraud is a general term for the unauthorized use of funds in a transaction, typically by means of a credit or debit card (Bhattacharyya et al. 2011a). Figure 1 shows a typical fraud detection framework deployed in the commercial system (Cheng et al. 2020a). A direct way to detect fraud is to match each transaction according to specific rules such as card blacklists and budget checking. However, criminals will also obtain the knowledge of vulnerabilities from the response of the pre-designed rule system, thus invalidating the original system. To solve the invalidation problem, the predictive model is designed to automatically detect fraud patterns and produces a fraud risk score. Domain experts then can thereby focus on the high-risk transactions.

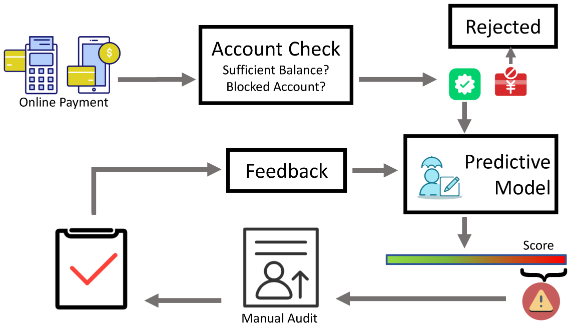


Figure 1: The illustration of typical credit card fraud detection process. The detection system of card issuer assesses each transaction with an online predictive model once it has passed account checking.

State-of-the-art. In the literature, many existing predictive models have been extensively studied to deal with fraud transactions (e.g., (Patidar, Sharma et al. 2011; Fu et al. 2016)), which can be classified into two categories: (1) *Rule-based methods* directly generate sophisticated rules by domain experts to identify the suspicious transactions. For instance, authors in (Seeja and Zareapoor 2014) proposed an association rule method for mining frequent fraud rules; (2) *Machine learning-based methods* learn static models by exploring large amounts of historical data. For example, authors in (Fiore et al. 2017) extracted features based on neural networks and built supervised classifiers for detecting fraudulent transactions. Recently, graph machine learning-based methods have been proposed (Wang et al. 2019a) where the transactions are modeled as a graph, and the advanced graph embedding techniques are deployed.

[*]Corresponding Author is Dawei Cheng.

Motivation. The state-of-the-art fraud detection techniques (Dou et al. 2020; Liu et al. 2020, 2021) can well capture the temporal or graph-based patterns of the transactions and significantly advance the performance of credit card fraud detection. However, they have at least one of the following three main limitations: (1) ignoring unlabeled data containing rich fraud pattern information; (2) ignoring categorical attributes, which are ubiquitous in the real production environment; and (3) requiring too much time on feature engineering, especially for categorical features.

These motivate us to design a semi-supervised graph neural network for credit card fraud detection. In particular, to capture the relationships among the credit card transactions associated with temporal information, we leverage a temporal transaction graph to model the time-relevant patterns. Besides, labeling the transactions is time-consuming and cost expensive. Only a tiny proportion (much less than 10%) of transactions are labeled in billions of real-life transactions, which contains many fraud patterns that have not been detected. Therefore, it is crucial to exploit the natural features from unlabeled data. In this paper, we design a Gated Temporal Attention Network (GTAN) for the temporal transaction graph, which can extract temporal fraud patterns and exploit both labeled and unlabeled data. In addition, categorical attributes are ubiquitous and useful in real applications. Therefore, it is necessary to leverage useful information through an attribute-driven model. In this paper, we introduce an attribute learning layer for preprocessing the transaction attributes and add risk embedding as new categorical attributes, which can better model fraud patterns (e.g., attribute embedding learning and risk propagation).

Contributions of our work are summarized as follows:

- We model credit card behaviors as a temporal transaction graph and formulate a credit card fraud detection problem as a semi-supervised node classification task.
- We present a novel attribute-driven temporal graph neural network for credit card fraud detection. Specifically, we propose a gated temporal attention network to extract temporal and attribute information. And we pass attributes and risk information on the temporal transaction graph to exploit both labeled and unlabeled data.
- Extensive experiments on three datasets show the superiority of our proposed GTAN on fraud detection. Semi-supervised experiment results show that, when leveraging rich information from the unlabeled data and a bit of labeled data, our proposed method detects more fraud transactions than baselines.

Related Works

Credit Card Fraud Detection

Several machine learning techniques have been proposed in the literature to address the credit card fraud detection problem. For instance, in (Maes et al. 2002), Bayesian Belief Networks (BBN) and Artificial Neural Networks (ANN) were applied on a real dataset obtained from Europay International. In (Şahin and Duman 2011) decision trees and support vector machines (SVMs) are applied on a real-world national bank dataset. The authors in (Fu et al. 2016) showed that using convolution to extract patterns can achieve higher accuracy than non-convolution neural networks. Recently, graph-based fraud detection techniques were proposed. For instance, CARE-GNN (Dou et al. 2020) was proposed to tackle fraud detection on relational graphs. PC-GNN (Liu et al. 2021) was proposed for imbalanced supervised learning on graphs. (Fiore et al. 2017) also proposed a generative adversarial network to improve the classification performance. In (Cheng et al. 2020a,b), authors proposed joint feature learning based on spatial and temporal patterns. However, they modeled the fraud patterns by using only one transaction/cardholder, thereby not being able to exploit the unlabeled data in real-life credit card transactions. The approach we present in this paper is radically different, as we employ a semi-supervised architecture, where the fraud patterns on both unlabeled and labeled data are jointly learned within an attribute-driven graph neural network framework.

Graph-Based Semi-supervised Learning

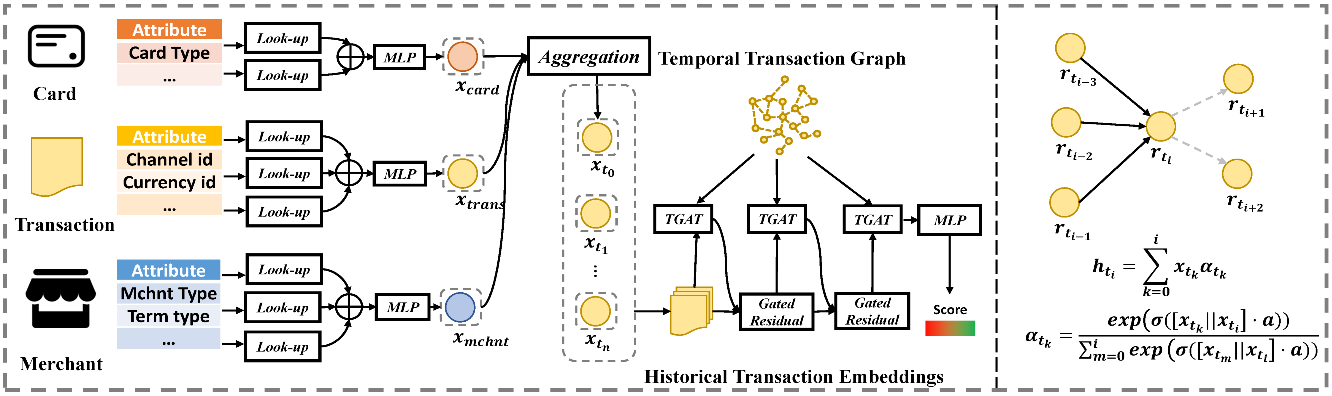
Many recent works have shown the benefit of using unlabeled node attributes in graph neural networks for a wide range of prediction tasks (Vaswani et al. 2017; Song et al. 2022), such as text classification (Xu et al. 2018), molecule property prediction (Guan et al. 2018) or language understanding (Shen et al. 2018). For instance, graph convolutional networks (GCN) were employed on partially labeled citation networks for node property prediction (Kipf and Welling 2016). GraphSAGE (Hamilton, Ying, and Leskovec 2017) was proposed to generate low-dimensional embeddings for previously unseen data. Graph attentive network model and random walks (Wang et al. 2019a) were deployed on social graphs to link the unlabeled and labeled data and pass messages among them. However, they still face at least one of the following two limitations: (1) cannot scale up to real-world graphs over millions of nodes (e.g., vanilla graph attention networks (Velickovic et al. 2018) have a space complexity $O(N^2)$, where N denotes the number of nodes, which is unaffordable for tasks with millions of nodes); (2) cannot propagate and learn the categorical attribute embeddings, especially for risk embeddings. Differently, our approach addresses the fraud detection problem via a message-passing model using categorical attributes, including risk information from node neighbors. Our work exploits attribute-driven model and semi-supervised graph neural networks to find more fraud patterns, which significantly improve the accuracy of credit card fraud detection.

Proposed Method

In this section, we first introduce the framework of our proposed GTAN. Then, we present the process of feature engineering, the gated temporal attention networks, risk embedding and propagation, and the fraud detection classifier. Lastly, we introduce the optimization strategy.

Model Architecture

The general model architecture of our proposed method is illustrated in Figure 2. Raw attributes of transaction records



(a) Attribute Embedding

(b) Gated Temporal Attention Networks

(c) Temporal Graph Attention (TGAT)

Figure 2: The illustration of the proposed model architecture and temporal graph attention mechanism.

are first learned by the attribute embedding look-up and feature learning layer, which includes feature aggregation with a multi-layer perception (MLP). In our implementation, the attributes of the card include the card type, cardholder type, card limit, remaining limit, etc. The transaction attributes include the channel ID, currency ID, transaction amount, etc. The merchant attributes contain merchant type, terminal type, merchant location, sector, charge ratio, etc. Then, we devise a gated temporal attention network to aggregate and learn the importance of historical transaction embeddings. Afterward, we leverage a two-layer MLP to learn the fraud probability from these representations. The whole model can be optimized in an end-to-end mechanism jointly with the existing stochastic gradient descent algorithm.

Attribute Embedding and Feature Learning

Given transaction records $\mathbf{r} = (r_1, r_2, \cdots, r_N)$, each record r_i contains card attributes f_c^i, transaction attributes f_r^i, and merchant attributes f_m^i as $r_i = f_c^i, f_r^i, f_m^i$. In preprocessing, different from (Cheng et al. 2020b), we do not filter out any cards or merchants with few authorized transaction records. As the number of cards and merchants which have never been checked manually is much larger than checked, we adopt full transaction records of users to maintain all potential frauds. Afterward, we construct the numerical attribute representation of each record into tensor format $\mathbf{X}_{num} \in \mathbb{R}^{N\times d}$, where N denotes the number of transactions, and d denotes the dimensions of features. Besides, we extract the card, transaction, and merchant category attributes $\mathbf{X}_{cat} \in \mathbb{R}^{N\times d}$ separately through attribute embedding layers, which can be formulated as follows:

$$\begin{aligned} e_{attr} =& \text{onehot}(f_{attr}) \odot \mathbf{E}_{attr}, \\ x_{cat,i} =& \text{MLP}_i\Big(\sum_{\forall j \in table_i} e_j\Big), i \in \{card, trans, mchnt\}, \end{aligned} \tag{1}$$

where $j \in table_i$ denotes the column j in our input table data i, $e_{attr} \in \mathbb{R}^{1\times d}$ denotes the embedding of attribute $attr$, onehot$(\cdot)$ denotes the one-hot encoding, f_{attr} denotes the single attribute of one transaction, and $\mathbf{E}_{attr} \in \mathbb{R}^{m\times d}$ denotes the embedding matrix of attribute $attr$, where m denotes the maximum number of attribute $attr$. After obtaining the embedding vector of each attribute in the card, transaction, and merchant tables, we aggregate these embeddings to obtain each transaction's categorical embedding through add-pooling with $x_{cat}^{(u)} = \sum_i x_{cat,i}^{(u)}, i \in \{card, trans, mchnt\}$, where $x_{cat}^{(u)} \in \mathbb{R}^{1\times d}$ denotes the category embedding vector of the u-th transaction record. To address the heterogeneity of categorical attributes, our proposed feature learning layer can model all categorical attributes and project them to a unified spatial dimension, which benefits our attribute-driven graph learning model.

Gated Temporal Attention Networks

To learn the temporal fraud patterns, we generate the temporal transaction graph (Xiang et al. 2021, 2022b) and aggregate messages on this graph to update the embedding of each transaction. Particularly, the directed temporal edges are generated with the previous transactions as the source and the current ones as the target, as illustrated in Figure 2(c). Then we aggregate messages through Temporal Graph Attention (TGAT) mechanism (Xiang et al. 2022a). The number of generated temporal edges per node is a hyper-parameter, which will be studied in the experiment section.

Temporal Graph Attention. After the feature engineering and attribute embedding, we leverage a series of transaction embeddings $\mathbf{X} = \{x_{t_0}, x_{t_1}, ...x_{t_n}\}$ to learn the temporal embedding of each transaction record. First, we combine categorical attributes and numerical attributes as the input of GTAN network with $x_{t_i} = x_{num}^{(t_i)} + x_{cat}^{(t_i)}$. At the first TGAT layer, we set $\mathbf{H}_0 = \mathbf{X}$ as the input embedding matrix. Afterward, we leverage multi-head attention to separately calculate the importance of each neighbor and update embeddings, which can be formulated as follows:

$$\mathbf{H} = \text{Concat}(\text{Head}_1, ..., \text{Head}_{h_{att}})\mathbf{W}_o, \tag{2}$$

where h_{att} denotes the number of heads, $\mathbf{W}_o \in \mathbb{R}^{d \times d}$ denotes learnable parameters, $\mathbf{H}$ denotes the aggregated embeddings with $\mathbf{H} = \{h_{t_0}, h_{t_1}, ..., h_{t_n}\}$, and each attention head is formulated as follows:

$$\begin{aligned} \text{Head} &= \sum_{x_i \in \mathcal{X}} \sigma(\sum_{x_t \in \mathcal{N}(x_i)} \alpha_{x_t, x_i} x_t), \\ \alpha_{x_t, x_i} &= \frac{\exp(\text{LeakyReLU}(\mathbf{a}^T [x_t || x_i))}{\sum_{x_j \in \mathcal{N}(x_t)} \exp(\text{LeakyReLU}(\mathbf{a}^T [x_t || x_j]))}, \end{aligned} \tag{3}$$

where $\mathcal{N}(x_i)$ denotes the temporal neighbors of the i-th transaction, α_{x_t, x_i} denotes the importance of temporal edge (x_t, x_i) in each attention head, and $\mathbf{a} \in \mathbb{R}^{2d}$ denotes the weight vector of each head. In practice, to avoid extra space consumption in extreme cases (such as high-frequency transactions in a short period), we use a neighbor sampling and truncation strategy to control the number of neighbor nodes $|\mathcal{N}(x_t)|$ (i.e., the number of associated temporal edges per node) through which the temporal graph attention layer propagates messages. Besides, to avoid borrowing future information, the neighbor transactions sampled for each transaction must be the past transactions from the same cardholder so that we can model the temporal fraud pattern through message passing on the temporal transaction graph.

Attribute-driven Gated Residual. To further improve the effectiveness and interpretability of our method, after obtaining aggregated embeddings, we leverage the embeddings and raw attributes to infer the importances of the aggregated embeddings and raw attributes after each layer of TGAT, which can be formulated as follows:

$$\begin{aligned} \text{gate}_{t_i} =& \sigma([x_{cat,t_i} || x_{num,t_i} || h_{t_i}] \beta_{t_i}), \\ z_{t_i} =& \text{gate}_{t_i} \cdot h_{t_i} + (1 - \text{gate}_{t_i}) \cdot x_{t_i}, \end{aligned} \tag{4}$$

where $\text{gate}_{t_i} \in [0, 1]$ denotes the gate variable of the t_i-th transaction, σ denotes the sigmoid activation function, $\beta_{t_i} \in \mathbb{R}^{3d \times 1}$ denotes the gate vector, and z_{t_i} denotes the output vector of each TGAT layer, which is fed into the next layer as input. According to our framework, if we stack a new TGAT layer with the attribute-driven gated residual mechanism, we use the output of the k-th gating mechanism as the input of the $k+1$-th TGAT.

Risk Embedding and Propagation

Inspired by unifying label propagation with feature propagation (Shi et al. 2021), we propose to take the manually annotated label as one of the categorical attributes of the transaction, and get the embedding of this categorical attribute, which we call *risk embedding*. Specifically, we take the manually annotated label as the risk feature of each transaction, where the category of unlabeled data is 'unlabeled', and the category of the rest of the data is 'fraud' or 'legitimate'. Then, we add this feature to the transaction data as one of our input categorical attributes. Due to concerns about label leakage, this attribute has not been used in previous fraud detection solutions. We will discuss the techniques for avoiding label leakage later. Then, we propose to embed the partially observed risk attributes (i.e., labels) into the same space as the other node features, which consist of the risk embedding vectors for labeled nodes and zero embedding vectors for the unlabeled ones. Then, we add the node features and risk embeddings together as input node features with $x_{t_i} = x_{num}^{(t_i)} + x_{cat}^{(t_i)} + \tilde{y}^{(t_1)} \mathbf{W}_r$, where $\mathbf{W}_r$ denotes the learnable parameters of risk embedding. (Shi et al. 2021) have proved that by mapping partially-labeled $\hat{\mathbf{Y}}$ and node features $\mathbf{X}$ into the same space and adding them up, we can use one graph neural network to achieve both attribute propagation and label propagation. Therefore, our fraud detection model can jointly model the temporal fraud patterns and risk propagation by adding the transaction label as one of the transaction categorical attributes.

Fraud Risk Prediction

After obtaining the aggregated embeddings of transactions, we leverage a two-layer MLP to predict the fraud risk, which is formulated as follows:

$$\hat{\mathbf{y}} = \sigma(\text{PReLU}(\mathbf{H}\mathbf{W}_0 + \mathbf{b}_0)\mathbf{W}_1 + \mathbf{b}_1), \tag{5}$$

where $\hat{\mathbf{y}} \in \mathbb{R}^{N \times 1}$ denotes the risk prediction results of all transactions, and $\mathbf{W}$ and $\mathbf{b}$ denote the learnable parameters of MLP. Afterward, we calculate the objective function $\mathcal{L}$ via binary cross-entropy, which is formulated as follows:

$$\begin{aligned} \mathcal{L} = -\frac{1}{N} \sum_{i=0}^{N} [\mathbf{y}_i \cdot \log p(\hat{\mathbf{y}}_i | \mathbf{X}, \mathbf{A}) + \\ (1 - \mathbf{y}_i) \cdot \log(1 - p(\hat{\mathbf{y}}_i | \mathbf{X}, \mathbf{A}))], \end{aligned} \tag{6}$$

where $\mathbf{y}$ denotes the ground-truth label of transactions. The proposed GTAN can be optimized through the standard SGD-based algorithms.

Masking to Avoid Label Leakage

Previous works only took risk information as optimization objectives to supervise their fraud detection model training. Unlike previous credit card fraud detection solutions, we semi-supervise our model by propagating transaction attributes and risk embeddings among labeled and unlabeled transactions. Using an unmasked objective for our fraud detection model will result in label leakage in the training process. In this case, our model will directly take observed labels and neglect the complicated hidden fraud patterns, which cannot be generalized in predicting future fraud transactions. Therefore, we propose to learn from the risk information of each transaction's neighbor transactions instead of learning from the label of itself. Specifically, a masked fraud detection training strategy is leveraged. During each training step, we randomly sample a batch of nodes, namely center nodes, along with the neighbor nodes corresponding to each center node. Then, we convert the partially observed labels $\hat{\mathbf{Y}}$ into $\tilde{\mathbf{Y}}$ by masking all the center nodes' risk embeddings to zero embeddings and keeping the others unchanged. Then, our objective function is to predict $\hat{\mathbf{Y}}$ with given $\mathbf{X}$, $\tilde{\mathbf{Y}}$ and $\mathbf{A}$:

$$\begin{aligned} \mathcal{L} = -\frac{1}{|V|} \sum_{i=0}^{|V|} [\mathbf{y}_i \cdot \log p(\hat{\mathbf{y}}_i | \mathbf{X}, \tilde{\mathbf{Y}}, \mathbf{A}) + \\ (1 - \mathbf{y}_i) \cdot \log(1 - p(\hat{\mathbf{y}}_i | \mathbf{X}, \tilde{\mathbf{Y}}, \mathbf{A}))], \end{aligned} \tag{7}$$

Dataset	YelpChi	Amazon	FFSD
#Node	45,954	11,948	1,820,840
#Edge	7,739,912	8,808,728	31,619,440
#Fraud	6,677	821	33,858
#Legitimate	39,277	11,127	141,861
#Unlabeled	-	-	1,645,121

Table 1: Statistics of the three fraud detection datasets.

where $|V|$ represents the number of center nodes with masked labels. In this way, we can train our model without the self-loop leakage of risk information; and during inference, we employ all observed labels $\hat{\mathbf{Y}}$ as input categorical attributes to predict the risk of the transactions out of the training set. The optimization objective of our model can be intuitively summarized as modeling the fraud patterns by the attribute information of neighboring transaction nodes and the attribute information.

Experiments

In this section, we first describe the datasets used in the experiments, then compare our fraud detection performance with other state-of-the-art baselines on two supervised graph-based fraud detection datasets and one semi-supervised dataset. Then, we perform ablation studies by evaluating two variants of the proposed GTAN, which demonstrates the effectiveness of our proposed method and attribute-driven mechanism.

Experiment Settings

Datasets To the best of our knowledge, we did not find any public semi-supervised credit card fraud detection dataset. Therefore, we collect the partially labeled records from our collaborated partners, namely Finacial Fraud Semi-supervised Dataset (**FFSD**). The ground truth labels are obtained on cases reported by consumers and confirmed by financial domain experts. If a transaction is reported by a cardholder or identified by financial experts as fraudulent, we label it as 1; otherwise, it is labeled as 0. Besides, we also experimented on two public supervised fraud detection datasets. The **YelpChi** graph dataset (Rayana and Akoglu 2015) contains a selection of hotel and restaurant reviews on Yelp. Nodes in the graph of the YelpChi dataset are reviews with 32-dimensional features, and the edges are the relationships among reviews. The **Amazon** graph dataset (McAuley and Leskovec 2013) includes product reviews of musical instruments. The nodes in the graph are users with 25-dimensional features, and the edges are the relationships among reviews. Some basic statistics of the three datasets are shown in Table 1.

Compared Methods. The following methods are compared to highlight the effectiveness of the proposed GTAN.

- *GEM.* Heterogeneous GNN-based model proposed in (Liu et al. 2018). We set the learning rate to 0.1 and the number of hops of neighbors to 5.
- *FdGars.* Fraudster detection via the graph convolutional networks proposed in (Wang et al. 2019b). We set the learning rate to 0.01 and the hidden dimension to 256.
- *Player2Vec.* Attributed heterogeneous information network proposed in (Zhang et al. 2019). We set the same parameters as the FdGars model.
- *Semi-GNN.* A semi-supervised graph attentive network for financial fraud detection proposed in (Wang et al. 2019a). We set the learning rate to 0.001.
- *GraphSAGE.* The inductive graph learning model proposed in (Hamilton, Ying, and Leskovec 2017). We set the embedding dimension to 128.
- *GraphConsis.* The GNN-based model tackling the inconsistency problem, proposed in (Liu et al. 2020). We used the default parameters suggested by the original paper.
- *CARE-GNN.* The GNN-based model tackling fraud detection on a relational graph (Dou et al. 2020). We used default parameters from the original paper.
- *PC-GNN.* A GNN-based model remedying the class imbalance problem, proposed in (Liu et al. 2021). We used the default parameters from the original paper.
- **GTAN.** The proposed gated temporal attention network model.[1] We also evaluate two variants of our model, GTAN-A and GTAN-R, in which the temporal graph attention component and risk embedding component are not considered, respectively. We set the batch size to 256, the learning rate to 0.0003, the input dropout ratio to 0.2, the number of heads to 4, the hidden dimension d to 256, and train the model with the Adam optimizer for 100 epochs with early stopping.

Evaluation Metrics We evaluate the experimental results on credit card fraud detection and opinion fraud datasets by the area under the ROC curve (AUC), macro average of F1 score (F1-macro), and averaged precision (AP), which are calculated as follows:

We count the number of True Positive N_{TP} (i.e. correct identification of positive labels), False Positive N_{FP} (i.e., incorrect identification of positive labels), and False Negatives N_{FN} (i.e., incorrect identification of negative labels). Then, F1 score and AP as formulated with $F1_{macro} = \frac{1}{l}\sum_{i=1}^{l}\frac{2\times P_i\times R_i}{P_i+R_i}$ and $AP = \sum_{i=1}^{l}(R_i - R_{i-1})P_i$, where $P_i = N_{TP}/(N_{TP}+N_{FP})$ and $R_i = N_{TP}/(N_{TP}+N_{FN})$. We also report the AUC[2] in our experiments.

Fraud Detection Performance

In YelpChi and Amazon datasets, we set the ratio of training to testing as 2:3. In the FFSD dataset, transactions of the first 7 months are used as training data, and then we detect fraud transactions in the following 3 months (August, September, and October of 2021). We repeat the experiments ten times

[1] The sources of our proposed method GTAN will be available at https://github.com/finint/antifraud.

[2] https://scikit-learn.org/stable/modules/generated/sklearn.metrics.auc.html

Dataset	YelpChi			Amazon			FFSD		
	AUC	F1	AP	AUC	F1	AP	AUC	F1	AP
GEM	0.5270	0.1060	0.1807	0.5261	0.0941	0.1159	0.5383	0.1490	0.1889
Player2Vec	0.7003	0.4121	0.2473	0.6185	0.2451	0.1291	0.5278	0.2147	0.2041
FdGars	0.7332	0.4420	0.2709	0.6556	0.2713	0.1438	0.6965	0.4089	0.2449
Semi-GNN	0.5161	0.1023	0.1811	0.7063	0.5492	0.2254	0.5473	0.4485	0.2758
GraphSAGE	0.5364	0.4508	0.1712	0.7502	0.5795	0.2624	0.6527	0.5370	0.3844
GraphConsis	0.7060	0.6041	0.3331	0.8782	0.7819	0.7336	0.6579	0.5466	0.3876
CARE-GNN	0.7934	0.6493	0.4268	0.9115	0.8531	0.8219	0.6623	0.5771	0.4060
PC-GNN	0.8174	0.6682	0.4810	0.9581	0.9153	0.8549	0.6795	0.6077	0.4487
GTAN	**0.9241***	**0.7988***	**0.7513***	**0.9630***	**0.9213***	**0.8838***	**0.7616***	**0.6764***	**0.5767***

Table 2: Fraud detection performance on three datasets, compared with popular benchmark methods. We report the results of the area under the roc curve (AUC), macro average of F1 score (F1-macro), and averaged precision (AP).

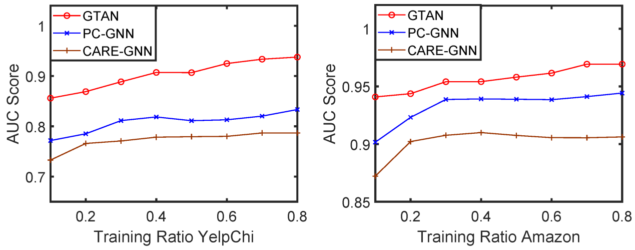


Figure 3: The result of semi-supervised experiments with different ratios of labeled training data.

for each method and show the average performance in Table 2. * denotes that the improvements are statistically significant for $p < 0.01$ according to the paired t-test.

The first five rows of Table 2 report the results of some classic graph-based methods, including GEM, Player2Vec, FdGars, Semi-GNN, and GraphSASE. It is clear that the result of GEM is not satisfactory, showing the limitation of a shallow model in addressing the complex fraud patterns. Player2Vec and FdGars improve the performance, partially due to the enlarged model capacity. Semi-GNN and GraphSAGE are close to each other and better than the first three methods. The result demonstrates the effectiveness of deep graph learning-based models in detecting fraud transactions. PC-GNN achieves more competitive results by employing the transaction graphs in the learning process, which is considerably better than the above baselines. The last row of Table 2 shows that our methods GTAN significantly outperforms all baselines with at least 10%, 0.5%, and 6% AUC improvements in three datasets, respectively. Besides, our proposed GTAN also outperforms other baselines with at least 27%, 2.9%, and 12.8% AP improvements in three datasets, respectively, which strongly proves the effectiveness of employing a semi-supervised temporal graph attention network for fraud detection.

Semi-supervised Experiment

To compare the capability of semi-supervised learning, we further set the ratio of the training set to the whole dataset to different values. For brevity of the diagrams, we select the two most competitive baselines (i.e., PC-GNN and CARE-GNN) for the following semi-supervised experiments. We vary the percentages of nodes used for training from 10% to 80% with an incremental of 10% for eight sets of experiments, with the remaining nodes as the test set in each set of experiments. We perform experiments on the YelpChi and Amazon datasets since they are fully annotated, which allows us to vary the ratio of labeled data in a wide range. The experimental results are shown in Figure 3.

On the YelpChi dataset, we can observe that GTAN always has the best performance under different training ratios. At the same time, in scenarios with little labeled data (10% training ratio), GTAN still performs well. As the number of labeled data increases, there is a steady improvement in the performance of GTAN. On the Amazon dataset, we can also observe that GTAN always has the best performance under different training ratios. Compared with the YelpChi dataset, the GTAN model on the Amazon dataset is less sensitive when changing the training ratio with no more than 2% variations in the AUC. This fully demonstrates that GTAN can achieve good performance even when there is a small portion of labeled data available (i.e., as low as 10%). Therefore, we conclude that the GTAN model is robust to training ratio changes and consistently outperforms PC-GNN and CARE-GNN, which shows the superiority of GTAN model in semi-supervised learning.

Ablation Study

The proposed model contains some key components, and we verify their effectiveness by ablating each component, respectively. Specifically, we evaluate two variants, namely GTAN-A and GTAN-R. In the GTAN-A model, we remove the TGAT component, and the central node aggregates messages directly collected from neighboring nodes. In this case, we obtain the transaction embeddings from all the neighbor embeddings with the same weight instead of adaptively adjusting the weights of neighbor nodes. In the GTAN-R model, we remove the risk embedding component and only use the original node attributes $\mathbf{X}$.

The grey bars in Figure 4 show the removal of the attention mechanism has the greatest impact on the accuracy met-

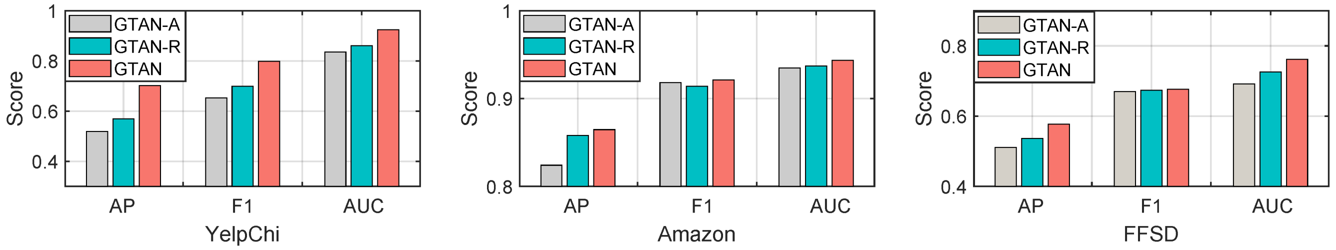


Figure 4: The ablation study results on three datasets. Gray bars represent the GTAN-A variant, blue bars represent the GTAN-R variant, and red bars represent the GTAN model.

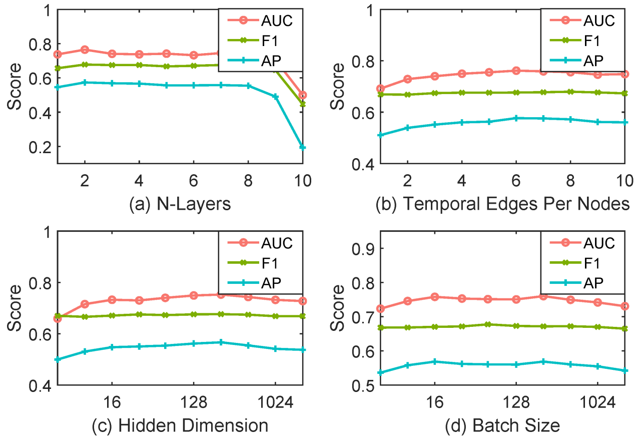


Figure 5: Parameter sensitivity analysis with respect to (a) the number of GNN layers; (b) the number of temporal edges per node; (c) hidden dimension; and (d) the batch size.

rics of the model. This proves that the reweighting of temporal transaction neighbors from the temporal graph attention mechanism is effective in our graph-based method. Besides, the green bars in Figure 4 gets the second-highest scores compared with GTAN, which proves that the risk embedding is effective in modeling credit card fraud patterns from transaction risk propagation. In summary, removing either component deteriorates the performance of GTAN, which proves that the temporal graph attention mechanism and risk embedding are both effective in graph-based fraud detection.

Parameter Sensitivity

In this section, we study the model parameter sensitivity by varying the depth of temporal graph attention layers, the number of temporal edges per node, the hidden dimension, and the batch size. The experimental results in the fraud detection dataset are reported in Figure 5.

We vary the depth of temporal graph attention layers from 1 to 10. As shown in Figure 5(a), our model's performance remains stable with up to 8 GNN layers. With deeper layers, our model tends to aggregate the temporal information from larger neighborhoods. Our model performs the best with two GNN layers when the AUC and AP reach the peak; therefore, we set the default depth to 2. The performance is degraded if we keep on increasing the depth of TGAT layers. The reason might be that deeper GNNs result in oversmoothing (Zhao and Akoglu 2020) in transaction embeddings. Figure 5(b) shows that when we increase the number of temporal edges per node from 1 to 10, our proposed model could consider neighbors in a wider range. Besides, our model requires at least 2 neighbors to learn graph-based transaction embeddings, and it reaches peak performance when the number of edges is 6. Beyond that, an excessive increase in the number of edges in the graph is not beneficial to the model's accuracy. Figure 5(c) shows that when we increase the hidden dimensions from 4 to 2048, our model maintains stable model performance and reaches a relative performance peak at 256. Figure 5(d) shows that our model performs best when the batch size is set as 64 or 256. Considering the training efficiency of the model, we set the batch size as 256. Generally, for values ranging from 16 to 512, the model is not sensitive to the hidden dimension and batch size, with less than 3% variations in the AUC.

Conclusion

In this paper, we studied an important real-world problem of credit card fraud detection. Considering that the labeling of fraud transactions is time-consuming and cost-expensive, we proposed an effective semi-supervised credit card fraud detection method by modeling data with temporal transaction graphs and developing attribute-driven gated temporal attention networks. Considering the ubiquitous categorical attributes and human-annotated labels, we proposed an attribute representation and risk propagation mechanism to model the fraud patterns accurately. The comprehensive experiments demonstrated the superiority of our proposed methods in three fraud detection datasets compared with other baselines. Semi-supervised experiments demonstrate the excellent fraud detection performance of our model with only a tiny proportion of manually annotated data. Our approach has been deployed in a transaction fraud analysis system. In the future, we will explore to study the temporal fraud patterns and risk-propagation fraud patterns in an effective and efficient way.

Acknowledgments

This work was supported by the National Science Fundation of China, 62102287, the Shanghai Science and Technology Innovation Action Plan Project (22511100700). Ling Chen was supported by ARC DP210101347.

References

AlFalahi, L.; and Nobanee, H. 2019. Conceptual Building of Sustainable Economic Growth and Corporate Bankruptcy. *Available at SSRN 3472409*.

Bhattacharyya, S.; Jha, S.; Tharakunnel, K.; and Westland, J. C. 2011a. Data mining for credit card fraud: A comparative study. *Decision Support Systems*, 50(3): 602–613.

Bhattacharyya, S.; Jha, S.; Tharakunnel, K. K.; and Westland, J. C. 2011b. Data mining for credit card fraud: A comparative study. *Decis. Support Syst.*, 50: 602–613.

Cheng, D.; Wang, X.; Zhang, Y.; and Zhang, L. 2020a. Graph neural network for fraud detection via spatial-temporal attention. *IEEE Transactions on Knowledge and Data Engineering*.

Cheng, D.; Xiang, S.; Shang, C.; Zhang, Y.; Yang, F.; and Zhang, L. 2020b. Spatio-Temporal Attention-Based Neural Network for Credit Card Fraud Detection. In *AAAI*, 362–369.

Dou, Y.; Liu, Z.; Sun, L.; Deng, Y.; Peng, H.; and Yu, P. S. 2020. Enhancing graph neural network-based fraud detectors against camouflaged fraudsters. In *Proceedings of the 29th ACM International Conference on Information & Knowledge Management*, 315–324.

Fiore, U.; De Santis, A.; Perla, F.; Zanetti, P.; and Palmieri, F. 2017. Using generative adversarial networks for improving classification effectiveness in credit card fraud detection. *Information Sciences*.

Fu, K.; Cheng, D.; Tu, Y.; and Zhang, L. 2016. Credit card fraud detection using convolutional neural networks. In *International Conference on Neural Information Processing*, 483–490. Springer.

Guan, Q.; Huang, Y.; Zhong, Z.; Zheng, Z.; Zheng, L.; and Yang, Y. 2018. Diagnose like a radiologist: Attention guided convolutional neural network for thorax disease classification. *arXiv preprint arXiv:1801.09927*.

Hamilton, W. L.; Ying, R.; and Leskovec, J. 2017. Inductive Representation Learning on Large Graphs. In *NIPS*.

Kipf, T. N.; and Welling, M. 2016. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907*.

Liu, Y.; Ao, X.; Qin, Z.; Chi, J.; Feng, J.; Yang, H.; and He, Q. 2021. Pick and choose: a GNN-based imbalanced learning approach for fraud detection. In *Proceedings of the Web Conference 2021*, 3168–3177.

Liu, Z.; Chen, C.; Yang, X.; Zhou, J.; Li, X.; and Song, L. 2018. Heterogeneous graph neural networks for malicious account detection. In *Proceedings of the 27th ACM International Conference on Information and Knowledge Management*, 2077–2085.

Liu, Z.; Dou, Y.; Yu, P. S.; Deng, Y.; and Peng, H. 2020. Alleviating the Inconsistency Problem of Applying Graph Neural Network to Fraud Detection. In *Proceedings of the 43nd International ACM SIGIR Conference on Research and Development in Information Retrieval*.

Maes, S.; Tuyls, K.; Vanschoenwinkel, B.; and Manderick, B. 2002. Credit card fraud detection using Bayesian and neural networks. In *Proceedings of the 1st international naiso congress on neuro fuzzy technologies*, 261–270.

Máté, D.; Sadaf, R.; Oláh, J.; Popp, J.; and Szűcs, E. 2019. The effects of accountability, governance capital, and legal origin on reported frauds. *Technological and Economic Development of Economy*, 25(6): 1213–1231.

McAuley, J. J.; and Leskovec, J. 2013. From amateurs to connoisseurs: modeling the evolution of user expertise through online reviews. In *Proceedings of the 22nd international conference on World Wide Web*, 897–908.

Patidar, R.; Sharma, L.; et al. 2011. Credit card fraud detection using neural network. *International Journal of Soft Computing and Engineering (IJSCE)*, 1(32-38).

Rayana, S.; and Akoglu, L. 2015. Collective opinion spam detection: Bridging review networks and metadata. In *Proceedings of the 21th acm sigkdd international conference on knowledge discovery and data mining*, 985–994.

Şahin, Y. G.; and Duman, E. 2011. Detecting credit card fraud by decision trees and support vector machines. In *Proceedings of the International MultiConference of Engineers and Computer Scientists*, volume 1, 442–447. Newswood Limited.

Seeja, K.; and Zareapoor, M. 2014. FraudMiner: A novel credit card fraud detection model based on frequent itemset mining. *The Scientific World Journal*, 2014.

Shen, T.; Zhou, T.; Long, G.; Jiang, J.; Pan, S.; and Zhang, C. 2018. Disan: Directional self-attention network for rnn/cnn-free language understanding. In *Thirty-Second AAAI Conference on Artificial Intelligence*.

Shi, Y.; Huang, Z.; Wang, W.; Zhong, H.; Feng, S.; and Sun, Y. 2021. Masked Label Prediction: Unified Massage Passing Model for Semi-Supervised Classification. In *IJCAI*.

Song, Z.; Yang, X.; Xu, Z.; and King, I. 2022. Graph-based Semi-supervised Learning: A Comprehensive Review. *IEEE transactions on neural networks and learning systems*, PP.

Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, Ł.; and Polosukhin, I. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, 5998–6008.

Velickovic, P.; Cucurull, G.; Casanova, A.; Romero, A.; Lio', P.; and Bengio, Y. 2018. Graph Attention Networks. *ArXiv*, abs/1710.10903.

Wang, D.; Lin, J.; Cui, P.; Jia, Q.; Wang, Z.; Fang, Y.; Yu, Q.; Zhou, J.; Yang, S.; and Qi, Y. 2019a. A Semi-supervised Graph Attentive Network for Financial Fraud Detection. In *2019 IEEE International Conference on Data Mining (ICDM)*, 598–607. IEEE.

Wang, J.; Wen, R.; Wu, C.; Huang, Y.; and Xion, J. 2019b. Fdgars: Fraudster detection via graph convolutional networks in online app review system. In *Companion Proceedings of The 2019 World Wide Web Conference*, 310–316.

Xiang, S.; Cheng, D.; Shang, C.; Zhang, Y.; and Liang, Y. 2022a. Temporal and Heterogeneous Graph Neural Network for Financial Time Series Prediction. *Proceedings of the*

31st ACM International Conference on Information &amp; Knowledge Management.

Xiang, S.; Cheng, D.; Zhang, J.; Ma, Z.; Wang, X.; and Zhang, Y. 2022b. Efficient Learning-based Community-Preserving Graph Generation. *2022 IEEE 38th International Conference on Data Engineering (ICDE)*, 1982–1994.

Xiang, S.; Wen, D.; Cheng, D.; Zhang, Y.; Qin, L.; Qian, Z.; and Lin, X. 2021. General graph generators: experiments, analyses, and improvements. *The VLDB Journal*.

Xu, D.; Wang, W.; Tang, H.; Liu, H.; Sebe, N.; and Ricci, E. 2018. Structured attention guided convolutional neural fields for monocular depth estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 3917–3925.

Zhang, Y.; Fan, Y.; Ye, Y.; Zhao, L.; and Shi, C. 2019. Key Player Identification in Underground Forums over Attributed Heterogeneous Information Network Embedding Framework. In *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*, 549–558.

Zhao, L.; and Akoglu, L. 2020. PairNorm: Tackling Oversmoothing in GNNs. *ArXiv*, abs/1909.12223.