Multi-Range Gated Graph Neural Network for Telecommunication Fraud Detection

Shuyun Ji
State Key Laboratory of Networking and Switching Technology
Beijing University of Posts and Telecommunications
Beijing, China
hmdpsbg@bupt.edu.cn

Jinglin Li*
State Key Laboratory of Networking and Switching Technology
Beijing University of Posts and Telecommunications
Beijing, China
jlli@bupt.edu.cn

Quan Yuan
State Key Laboratory of Networking and Switching Technology
Beijing University of Posts and Telecommunications
Beijing, China
yuanquan@bupt.edu.cn

Jiawei Lu
State Key Laboratory of Networking and Switching Technology
Beijing University of Posts and Telecommunications
Beijing, China
lu1997jiawei@gmail.com

***Abstract*—With the expansion of the mobile communication technology, telecommunication fraud is increasing dramatically which results in the loss of billions of dollars worldwide every year. In recent years, some detection methods utilize data mining and statistical techniques to detect fraud from large amount of subscriber content data, and some methods transform the social relation into a set of topological features, such as *degree*, *k-core* etc. However, both content and relation have not been fully explored for identifying fraudsters. In this paper, we propose the *Multi-Range Gated Graph Neural Network* (MRG-GNN) for learning latent features of social network. Specifically, we first model a social network as a directed graph where vertices with subscriber features represent subscribers and edges with relational features represent activities between them. Then, a novel method based on efficient short walks and node-merging is proposed to structure the convolutions, and graph convolution block captures content information and relation information between users. The multi-range gated readout operation is proposed to aggregate informative features in multiple nodes and automatically learns the representation of user social network. Finally, experiments on a real-world telecommunication network show that our MRG-GNN achieves the state-of-the-art results.**

Keywords—Fraud detection, Social Network, Graph Neural Networks, Classification, Telecommunications

I. INTRODUCTION

Telecommunication fraud is a fast-growing field of criminal activity and a low-risk alternative to traditional methods of financial crime, resulting in substantial losses. Due to the complicated interpersonal relationships and essential uncertainty brought by the telecommunication network, telecommunication fraud detection has become a complex and important issue to be explored.

The past decade has seen significant development in telecom fraud detection techniques, including content-based approaches and relation-based approaches. Nearly all of existing content-based approaches, implemented by a number of methods such as statistical methods and data mining [1], explore the subscriber information or behavior evidence according to given all Call Detail Records (CDR). On the other hand, considering that the telecommunication network contains rich information, relation-based approaches rely on social network analysis and transform the social relation into a set of topological features [4], such as *degree*, *k-core*, score of *Page Rank* etc. However, the multidimensional nature of these interactions has largely been ignored. To date, unlikely purely content-based or structure-based models, graph neural networks (GNNs) have achieved remarkable success for representation learning on graphs, leveraging both content information as well as graph structure. The core idea behind GNNs is to learn how to iteratively aggregate feature information from local graph neighborhoods using neural networks. Moreover, [8] reformulates existing models into a unified framework called Message Passing Neural Network (MPNN), with the appropriate message, update, and readout functions. Among the most interesting recent models, an edge convolutional operator [8] is proposed to allow vector valuing edge features and graph isomorphism network [9] is a powerful model with neighborhood aggregation schemas at the node feature level. Hence, applying GNNs on the social network for fraud detection could be developed.

In this paper, our goal is to detect the fraudulent activities at a city-scale and it is a non-trivial task faced with the following three major challenges. (1) *Large-scale social network*. A city-wide telecommunication network consists of millions of users and traditional methods are not scalable to large graphs. The first challenge is how to operate on the large-scale social network and how to construct the computation graph for each user. (2) *Information fusion*. Fraudster's tricks are continuously evolving but there is limited research on fraud detection methods solely on behavior content or social relation. How to integrate content information and relation information is another challenge. (3) *Multiple range information*. In a social network, a small neighborhood range indicates the local dependency, and a large range tends to capture higher-order features. Moreover, information in different ranges dose not contribute equally in normal network and fraudulent network. The third challenge is how to leverage multiple range information.

To tackle above challenges, *Multi-Range Gated Graph Neural Network* (MRG-GNN) is proposed for learning social graph representation and tackling the fraud detection problem. The main contributions of this work are summarized as follows:

- An end-to-end learning GNNs model is proposed to incorporate both subscriber content and social relation for city-wide fraudulent activities identification. In addition, this model performs efficient, localized convolutions by dynamically constructing the computation graph with short-walk-based sampling and node-emerging strategies.

978-1-7281-6926-2/20/$31.00 ©2020 IEEE

- A hybridization of node convolutional operator and edge convolutional operator is proposed for information fusion and the multi-range gated readout operation which can aggregate more informative features from different node ranges, is introduced for the representation of social graph.
- Extensive experiments are conducted on the real-world telecommunication dataset collected from Shanghai, metropolis in China. The results demonstrate MRG-GNN achieves the best prediction performance against the baselines.

The rest of the paper is organized as follows. Section II briefly surveys the related works in telecommunication fraud detection. Section III defines the problem and presents the technical details for learning graph representation using MRG-GNN. In section IV, MRG-GNN is empirically evaluated on a real-world city-wide network and the results are analyzed. Finally, section V concludes with a brief summary.

II. Related Works

A. Telecommunication Fraud Detection

Previous works in the telecommunication fraud detection have concentrated on CDR data to create behavior profiles for the subscribers, and have detected deviations from these profiles. [3] proposes rule-discovery methodology combining two data levels, which are the customer data and behavior data (usage characteristics in a short time frame). [1] introduces a hybrid approach of data mining to detect fraudulent activities. Since neural networks can actually calculate user profiles in an independent manner, thus more elegantly adapting to the behavior of various users, [10] first uses a feed-forward neural network based on supervised learning. Aside from that, [11] successfully combines feedforward systems, Bayes methods and Gaussian mixture model together.

Unlike the majority of existing methods that focus on behavior analysis, some works exploit the network topology of social interactions. [4] raises a *k*-gram sequential feature with the help of Mixture Markov Model, [5] relies on the spectral space of underlying network topology to identify fraudsters, and [2] introduces "contrast suspiciousness", a dynamic weighting approach to detect fraudulent blocks.

B. Social Network Analysis

Recent advancements in deep learning methods for graph-structured data make it possible to model the complicated telecommunication social network. So far, GNNs have been applied to a wide range of social network analysis and have achieved considerable results. [6] introduces a novel approach for semi-supervised classification on citation networks, [7] explores the potential of online network representation learning in social influence analysis, and in social recommendation, [15] relies on the ratings and reviews of existing users which could indicate future customers' decision through social interaction.

C. Graph Classification with GNNs

Fraud detection can be converted to graph classification and the goal is to predict a label for user social graph. With GNNs algorithms, useful features for nodes can be learned by graph convolutional operators to tackle many node-focus tasks. To address graph-focus tasks, information of nodes need to be aggregated to form a graph-level representation. The most basic operations are simple statistics like taking sum, average or max-pooling [12]. Another commonly used approach to gather information is to add a fully connected (FC) layer as the final layer [13].

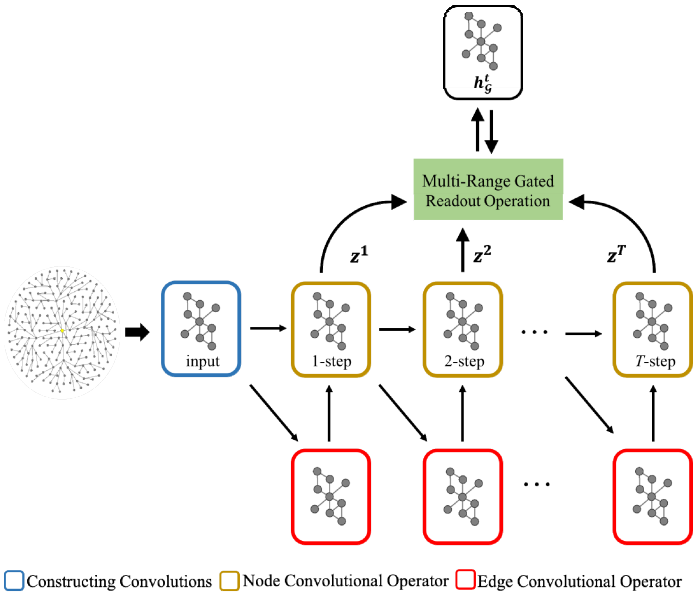


Figure 1: The architecture of the MRG-GNN.

III. Method

We assume a standard graph-based machine learning setup; the input directed graph G is a pair (V, E) where V is a finite set of nodes representing users with user features $x_v \in \mathbb{R}^{d_v}$ (e.g., user attribute, amount of user call) and E is a set of edges representing interactions among them with edge features $e_{vw} \in \mathbb{R}^{d_e}$ (e.g., number of call, call duration, call type). Here, d_v is the number of node features and d_e is the number of the edge features. The social fraud detection problem is to predict whether v_i with an unobserved label is a fraudster or not, based on the given social network G and a set of observed labels of already identified fraudster social network. Since GNNs could build blocks for learning data with non-Euclidean structures, we are interested in the representation of user social network, which can be easily classified later.

Figure 1 demonstrates the architecture of MRG-GNN, which consists of three parts: (1) constructing convolution module; (2) graph convolutional layers; and (3) multi-range gated readout operation. First, MRG-GNN constructs the computation graph for each user, then maps the graph to outputs. The graph convolutional layers consist of several node convolutional operators and edge convolutional operators, which model interactions among users and extract different range fused information. Then the multi-range gated readout operation leverages information in multiple range nodes and updates global graph embedding step by step. In addition, we combine MRG-GNN with a simple classifier (e.g., MLPs) for final fraudster prediction. The detailed implementation is specified in the following paragraphs.

A. Constructing Convolutions

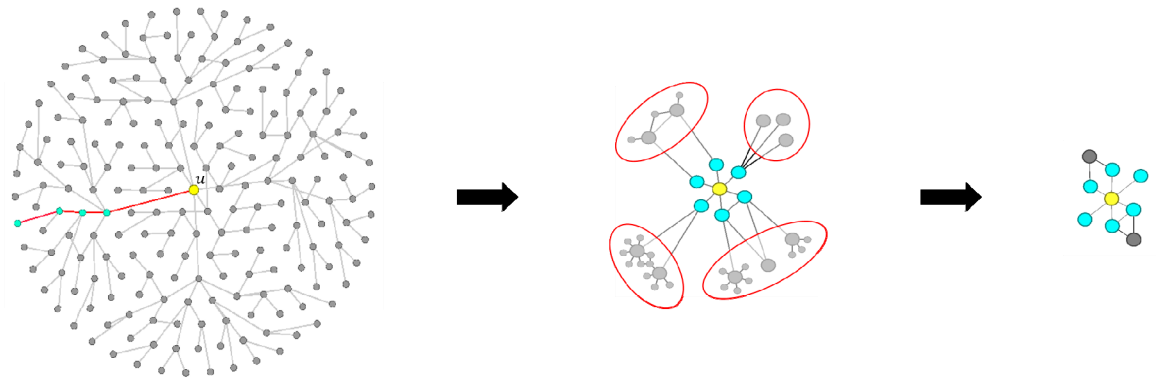


Figure 2: The illustration of the computation graph sampling and simplification. Left: a small example social network. For the source node *u*, an original subgraph is generated by short walks with fixed length l. Middle: the subgraph after short-walk sampling. For the subgraph, k-hops neighbors divided into clusters by immediate neighbors are merged and the new nodes with only one degree are removed. Right: the final computation graph after node-merging.

A social network successfully captures the correlations of users, e.g., "the degree of overlap of two people's friendship networks correlates with the strength of ties between them" [14]. Indeed, normal users who know each other are likely to share many common friends, and vice versa for fraudsters. However, according to the real-world dataset, the strong relationships in telecommunication network are rare and the activity between two users is likely to occur once. Another challenge is that GNNs could not efficiently take full neighborhoods of nodes to perform convolution during training when the underlying social network has millions of nodes and edges.

To tackle large-scale social network, we resort to sampling and node-merging. However, random sampling is suboptimal, and a new technique using short walks based on relatively strong relationship is proposed to sample the computation graph . And this sampling method mitigates the impact of invalid nodes. As shown in Figure 2, constructing convolutions is in two steps. First, beginning with the source node u (with the length l of walks being fixed), the short walks preserve source node's all immediate neighbors, which corresponds to behavior feature, and k-hops $(2 \le k \le l)$ neighbors are sampled based on the intimacy, which preserves relatively strong relationship. Second, to reduce the computational complexity, the strategy preserves the source node and its immediate neighbors, and merges *k*-hops neighbors. In addition, merged nodes with only one degree are removed.

B. Convolutional Layers

To model the interactions of both nodes and edges for a social network, one needs to stack multiple convolutional layers for learning the internal hidden representation of each node in the graph. In the original MPNN, the message passing phase runs for fixed steps T and is defined in terms of message function M_t and vertex update functions U_t. During the message passing phase, node's current hidden state $h_v^{(t)}$ at step t is updated based on message $m_v^{(t)}$according to

$$m_v^{(t)} = \sum_{w \in \mathcal{N}(v)} M_t\left(h_v^{(t-1)}, h_w^{(t-1)}, e_{vw}\right), \quad (1)$$

$$h_v^{(t)} = U_t\left(h_v^{(t-1)}, m_v^{(t)}\right), \quad (2)$$

where $\mathcal{N}(v)$ denotes neighbors of v in graph G. There may exist many powerful GNNs. Specifically, considering the edge information, the edge-wise convolution [8] is defined as

$$h_v^{(t)} = \Theta h_v^{(t-1)} + \sum_{w \in \mathcal{N}(v)} h_w^{(t-1)} \cdot f_\Theta{}^{(t)}(e_{vw}). \quad (3)$$

Then, aggregating node information from neighborhoods is the next step. Graph isomorphism network [9] is one of powerful node-wise GNNs, where the essential message of the model is that the best way to update the node features at step t is through the usage of

$$h_v^{(t)} = MLP^{(t)}\left(\left(1+\epsilon^{(t)}\right) \cdot h_v^{(t-1)} + \sum_{w \in \mathcal{N}(v)} h_w^{(t-1)}\right). \quad (4)$$

In this work, to integrate node features and edge features, we consider the following function as combination of message function and update function for information fusion:

$$h_v^{(t)} = f_\Theta{}^{(t)}\left(h_v^{(t-1)} + \sum_{w \in \mathcal{N}(v)} h_w^{(t-1)} \cdot g_\Theta{}^{(t)}(e_{vw})\right), \quad (5)$$

where $f_\Theta{}^{(t)}$ and $g_\Theta{}^{(t)}$ denote learned differentiable functions (e.g., MLPs). Considering the correlations between two users are complicated and interact with each other, a neural network $g_\Theta{}^{(t)}$ is used to map edge feature shape to node feature shape, which aggregates message from neighborhood and edges. In the node-wise condition, a neural network $f_\Theta{}^{(t)}$ can represent universal functions over a node and the multiset of its neighbors, which satisfies the injectiveness condition and updates the node hidden state.

C. Multi-Range Gated Readout Operation

The convolutional layer enables a model to learn different range node hidden states through integrating content information and relation information. However, with only the stack of convolutional layers, it is hard to leverage multiple range information for global graph representation. To remedy the limitation, our readout function is specifically proposed to operate on the set of multiple range node hidden

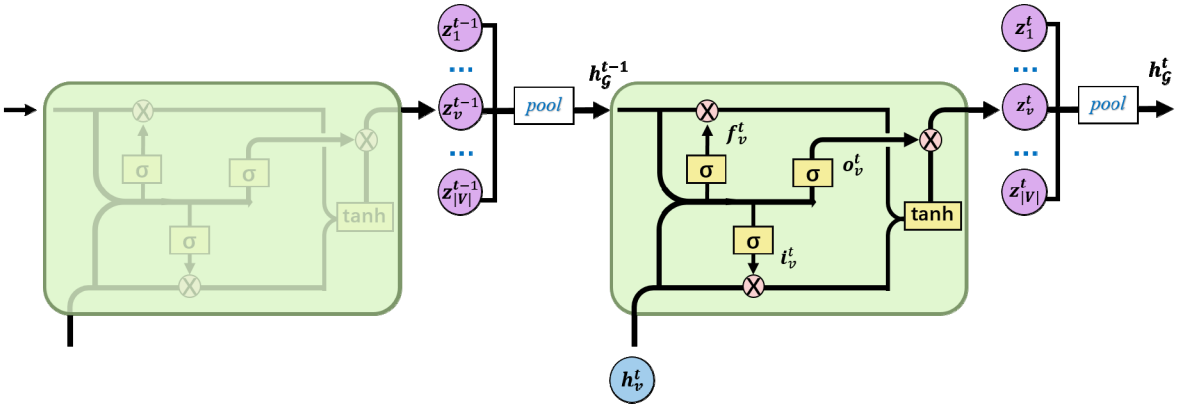


Figure 3: The illustration of multi-range gated readout operation.

states and must be invariant to permutations of the node hidden states. Inspired by LSTM and GRU, the biggest modification of readout function is that we apply multi-range gated units and unroll the recurrence for a fixed ran ge number of steps *T*, which updates the graph hidden state step by step. As is shown in Figure 3, multi-range gated readout operation contains three gates and a pooling layer. The details of the readout operation is followed.

Equation (6) is the node state initialization step, which copies node annotations into the first components of the hidden state and pads the rest with zeros. Equation (7) is the initialization of the graph state, which can be sum, average or max-pooling operation.

$$h_v^{(0)} = [x_v, 0], \tag{6}$$

$$h_{\mathcal{G}}^{(0)} = Pool\left(\left\{h_v^{(0)} | v \epsilon G\right\}\right). \tag{7}$$

The first step in gated units is to decode what information will be thrown away from graph previous hidden state. It is decided by the forget gate layer, which is a *sigmoid* layer defined as equation (8). The gate looks at graph previous hidden state $h_{\mathcal{G}}^{(t-1)}$ and node current hidden state $h_v^{(t)}$, and outputs a number between 0 and 1 for each number in $h_{\mathcal{G}}^{(t-1)}$.

$$f_v^{(t)} = \sigma\left(W_f\left[h_v^{(t)}, h_{\mathcal{G}}^{(t-1)}\right] + b_f\right). \tag{8}$$

The next step is to decide what new information will be stored in the graph current hidden state, which is decided by the input gate layer. The gate looks at $h_{\mathcal{G}}^{(t-1)}$ and $h_v^{(t)}$, and outputs a number for $h_v^{(t)}$, which is defined as:

$$i_v^{(t)} = \sigma\left(W_i\left[h_v^{(t)}, h_{\mathcal{G}}^{(t-1)}\right] + b_i\right). \tag{9}$$

The output gate layer defined as equation (10), decides what parts of information will be output. Then, the readout operation puts the previous state and the new information through *tanh* and multiplies the result by the output of the output gate. The task of gated readout operation is to learn the parameters, which are shared across all time-steps.

$$o_v^{(t)} = \sigma\left(W_o\left[h_v^{(t)}, h_{\mathcal{G}}^{(t-1)}\right] + b_o\right),$$

$$\tilde{z}_v^{(t)} = tanh\left(W\left(i_v^{(t)} * h_v^{(t)}\right) + U\left(f_v^{(t)} * h_{\mathcal{G}}^{(t-1)}\right)\right),$$

$$z_v^{(t)} = o_v^{(t)} * \tilde{z}_v^{(t)}. \tag{10}$$

At step t, the gated readout operation obtains information from different range node hidden states, $Z^t = \{z_1^t, z_2^t, \ldots, z_{|V|}^t\}$, $z_v^t \in \mathbb{R}^F$, where z_v^t is the range information from node *v*. Same with the equation (7), the graph hidden state at step *t* is formed by:

$$h_{\mathcal{G}}^{(t)} = pool\left(\left\{z_v^{(t)} | v \epsilon G\right\}\right). \tag{11}$$

In general, multi-range gated readout operation takes each element of node hidden state sequences as an input and combines the graph hidden state at the last time to leverage multi-range information for graph current hidden states step by step.

IV. Experiments

A. Datasets

MRG-GNN is evaluated on the real-world dataset collected from Shanghai, metropolis in China. This anonymized dataset is ranged from May 10, 2019 to June 23, 2019. And all call detail records in the city are collected by Shanghai telecommunication operators. Furthermore, only users who perform an activity in the sampling period are included in the dataset. Z-score normalization is applied to the inputs. The dataset is split in chronological order with 70% for training, 10% for validation, and 20% for testing. Detailed statistics of the dataset are shown in Table 1.

Table 1: Data Sample Statistics.

Entity	Count
$\|\mathcal{V}\|$ (total users)	54,973,350
$\|\mathcal{E}\|$ (total actions)	525,150,320
Unique edge	307,511,138
Maximum degree	664,266
Total users labeled as fraudsters	(%0.04) 20,163

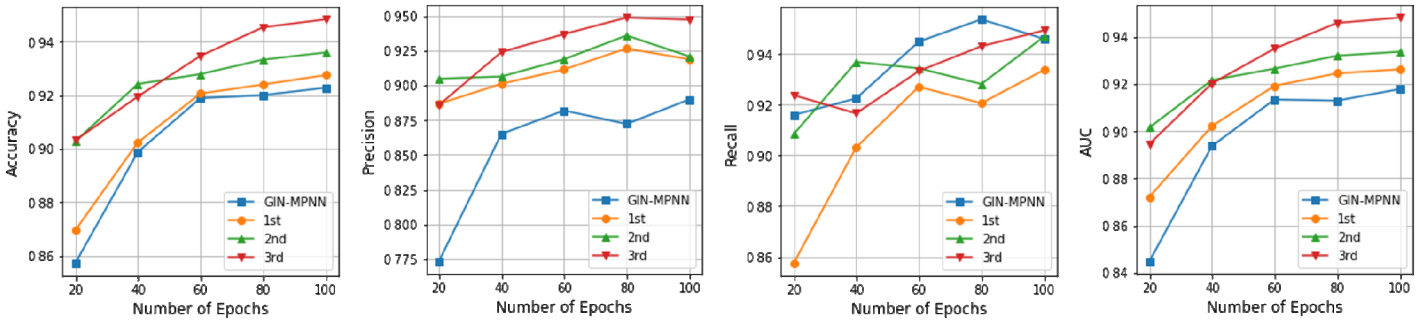
B. Baselines

MRG-GNN is compared with the following six models:

- SVM: support vector machine is a supervised learning model with associated learning. The user information from node features is the input for classification.

Table 2: Comparisons of classification accuracy, precision, recall and AUC.

	Accuracy	Precision	Recall	AUC
SVM	0.8685	0.8816	0.8290	0.8659
LGB	0.8696	0.8244	0.9149	0.8725
GCN	0.8739	0.9050	0.8632	0.8841
GIN	0.8952	0.9037	0.8893	0.8964
MPNN	0.9204	0.8401	0.9768	0.9084
GIN-MPNN	0.9228	0.8898	0.9459	0.9178
RNN-based	0.9230	0.9205	0.9256	0.9231
LSTM-based	0.9384	0.9079	0.9597	0.9338
GRU-based	0.9454	0.9208	**0.9626**	0.9417
1ST	0.9275	0.9187	0.9336	0.9262
2ND	0.9360	0.9208	0.9467	0.9337
3RD	**0.9484**	**0.9472**	0.9492	**0.9482**



(a) Accuracy (b) Precision (c) Recall (d) AUC

Figure 4: Impact of the number of epochs in models.

- LGB: LightGBM is a gradient boosting framework that uses tree based learning algorithms. The user information is the input for classification.
- GCN: graph convolutional network is a semi-supervised learning on graph-structured data. It scales linearly over the number of graph edges and learns hidden layer representations that encode both local graph structure and features of nodes. Moreover, the edge weight is calculated by the intimacy between two users.
- GIN: graph isomorphism network [9] achieves state-of-the-art performance on many graph classification benchmarks. In the same way, the input graph is weighted.
- MPNN: message passing neural network [8] is the implementation of an edge convolutional operator as message passing function.
- GIN-MPNN: hybridization of graph isomorphism operator and edge convolutional operator without gated readout operation. This model is our backbone network.

C. Experimental Settings

To sample computation graph for each user, the maximum length l of short walks is set to 3. The number of convolution layers is set to 3 for each model, and node hidden dimension size and graph representation dimension size are both set to 16. The mean-pooling which uses the element-wise mean as a symmetric aggregation function is applied to equation (7) and equation (11). And 2-layer MLP classifier is applied for final prediction. We train our model for 100 epochs using Adam optimizer to minimize the mean absolute error (MAE) with a learning rate of 0.001 and the batch size as 32. In addition, we compare our method with different readout operations, e.g., RNN, LSTM and GRU.

D. Result

Since the ground-truth label of each user is provided by the dataset, we adopt standard metrics, including precision, recall, accuracy and AUC to evaluate the performance of MRG-GNN. As shown in Table 2, the representation formed by MRG-GNN shows better discriminability than the baselines, and AUC improvement of MRG-GNN are (8.23%, 7.57%) beyond SVM and LGB. Furthermore, GNNs with other readout operation (RNN, LSTM, GRU) also achieve (1.6%, 2.39%, 3.04%) improvements beyond our backbone network (GIN-MPNN), respectively, which shows readout operation can extract better representation for telecommunication network.

To further verify the effectiveness of multi-range gated readout operation, MRG-GNN is compared against another differentiable readout operation. And MRG-GNN achieves state-of-the-art results of the fraud detection. Moreover, Figure 4 illustrates that multi-range information is captured for graph representation tasks via the performance of each

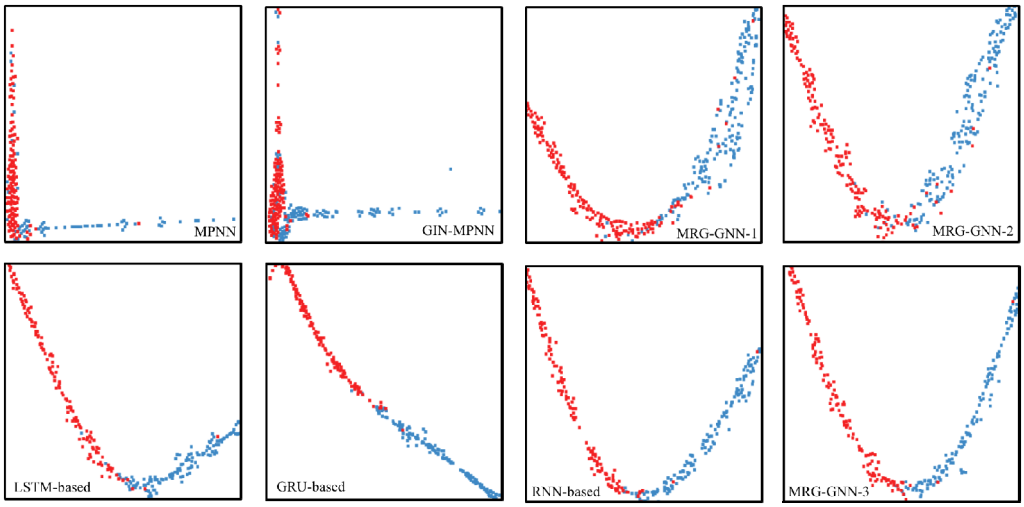


Figure 5: Visualization of graph representations from baselines and each MRG-GNN layer. Points of different colors indicate graph samples with different labels. Red indicates normal users and blue indicates telecommunication fraudsters. The data reduction and visualization for high-dimensional graph embedding is done with t-SNE [15].

convolution layer in MRG-GNN. The 3rd layer achieves (2.2%, 1.45%) AUC improvement beyond the 1st layer and 2nd layer for 100 epochs. This result demonstrates that the gradients from deeper convolution layers are of benefit to the training of shallow convolution layers.

To make the discussion more concrete, graph representations of different GNNs models are visualized in Figure 5. Points with different colors indicate the users with different labels and it can be observed that the accuracy of our models is always higher than other methods, which is consistent with the results in Table 2.

V. CONCLUTIONS

In this paper, we have evaluated the effect of *Multi-Range Gated Graph Neural Network* for city-wide fraudulent activities identification in telecommunication network. Firstly, a new technique using short-walk-based sampling and node-merging is proposed to structure graph convolutions. Then, a hybridization of graph isomorphism operator and edge convolutional operator capable of information fusion in large-scale social network, is proposed to convolutional layers. Finally, multi-range gated readout operation is proposed to efficiently leverage multiple range information for social network representation, which boosts the performance of the telecommunication fraud detection. MRG-GNN is evaluted on a real-world dataset, and the model achieves superior results compared with other fraud detection methods. Our work demonstrates the impact that graph convolutional methods can have in a production fraud detection system, and we believe that graph neural networks can be further extended in the future to tackle other graph representation learning tasks at large scale.

VI. REFERENCE

[1] Farvaresh, Hamid, and Mohammad Mehdi Sepehri. "A data mining framework for detecting subscription fraud in telecommunication." Engineering Applications of Artificial Intelligence 24.1 (2011): 182-194.

[2] Liu, Shenghua, Bryan Hooi, and Christos Faloutsos. "Holoscope: Topology-and-spike aware fraud detection." *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management. ACM, 2017.*

[3] Rosset, Saharon, et al. "Discovery of fraud rules for telecommunications—challenges and solutions." *Proceedings of the fifth ACM SIGKDD international conference on Knowledge discovery and data mining. ACM, 1999.*

[4] Fakhraei, Shobeir, et al. "Collective spammer detection in evolving multi-relational social networks." *Proceedings of the 21th acm sigkdd international conference on knowledge discovery and data mining. ACM, 2015.*

[5] Ying, Xiaowei, Xintao Wu, and Daniel Barbará. "Spectrum based fraud detection in social networks." *2011 IEEE 27th International Conference on Data Engineering. IEEE, 2011.*

[6] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," *ICLR 2017, 2017.*

[7] J. Qiu, J. Tang, H. Ma, Y. Dong, K. Wang, and J. Tang, "Deepinf: Social influence prediction with deep learning," in *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. ACM, 2018, pp. 2110–2119.

[8] J. Gilmer, S. S. Schoenholz, P. F. Riley, O. Vinyals, and G. E. Dahl, "Neural message passing for quantum chemistry," in *Proceedings of the International Conference on Machine Learning, 2017, pp. 1263–1272.*

[9] Xu, Keyulu, et al. "How powerful are graph neural networks?" *arXiv preprint arXiv:1810.00826 (2018).*

[10] Taniguchi, Michiaki, et al. "Fraud detection in communication networks using neural and probabilistic methods." *Proceedings of the 1998 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP'98 (Cat. No. 98CH36181). Vol. 2. IEEE, 1998.*

[11] Estévez, Pablo A., Claudio M. Held, and Claudio A. Perez. "Subscription fraud prevention in telecommunications using fuzzy rules and neural networks." Expert Systems with Applications 31.2 (2006): 337-344.

[12] D. K. Duvenaud, D. Maclaurin, J. Iparraguirre, R. Bombarell, T. Hirzel, A. Aspuru-Guzik, and R. P. Adams, "Convolutional networks on graphs for learning molecular fingerprints," in *Advances in Neural Information Processing Systems, 2015, pp. 2224–2232.*

[13] J. Bruna, W. Zaremba, A. Szlam, and Y. Lecun, "Spectral networks and locally connected networks on graphs," in *Proceedings of the 3rd International Conference on Learning Representations, 2014.*

[14] M. S. Granovetter. The strength of weak ties. American journal of sociology, pages 1360–1380, 1973.

[15] Ying, Rex, et al. "Graph convolutional neural networks for web-scale recommender systems." *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining.*