MutationGuard: A Graph and Temporal-Spatial Neural Method for Detecting Mutation Telecommunication Fraud

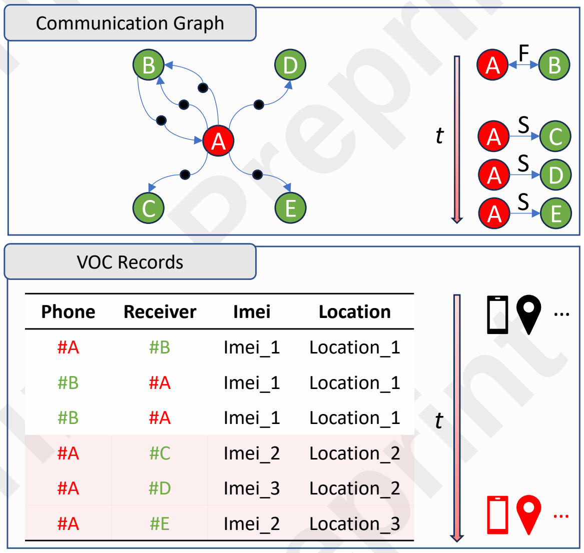
Haitao Bai[1], **Pinghui Wang**[1*], **Ruofei Zhang**[2], **Ziyang Zhou**[1], **Juxiang Zeng**[1], **Yulou Su**[3], **Li Xing**[3], **Zhou Su**[1], **Chen Zhang**[4], **Lizhen Cui**[5], **Jun Hao**[3], **Wei Wang**[3]
[1]Xi'an Jiaotong University
[2]Apple
[3]China Mobile Communications Group Shaanxi Co., Ltd.
[4]Zhejiang CreateLink Technology
[5]ShanDong University
haitao.bai@stu.xjtu.edu.cn, phwang@mail.xjtu.edu.cn, rfzhang@gmail.com, dakandao@stu.xjtu.edu.cn,
jxzeng@mail.xjtu.edu.cn, {suyulou, xingli}@sn.chinamobile.com, zhousu@ieee.org,
zhangchen@chuanglintech.com, clz@sdu.edu.cn, {haojun, wangwei34}@sn.chinamobile.com

Abstract

Telecommunication fraud refers to deceptive activities in the field of communication services. This research focuses on a category of fraud identified as "mutation telecommunication fraud". There is currently a lack of research on mutation telecommunication fraud detection, allowing this type of fraud to persist uncaught. We identify that detecting mutation fraud requires capturing multi-source patterns, including user communication graphs and temporal-spatial Voice of Call (VOC) features. Specifically, we introduce MutationGuard, which leverages Graph Neural Networks (GNN) to capture changes in user communication graphs. For VOC records, we map call start times onto a 3D cylindrical surface, thereby representing each VOC record in spatial coordinates and utilizing proposed LFFE and TCFE modules to capture local fraud behaviors and temporal behavior changes. The proposed neural modeling approach that facilitates multi-source information fusion constitutes a significant advancement in detecting mutation fraud. Experiment results reveal a significant improvement in the AUC score by 1.52% and the F_1 score by 1.36% on the proposed telecommunication fraud dataset. Particularly, our method shows a significant improvement of 13.93% in accuracy on mutation fraud data. We also validate the effectiveness of our method on the publicly available Sichuan Telecommunication Fraud dataset.

1 Introduction

Telecom fraud refers to fabricating stories through messages, phone calls, and other communication methods to acquire a large amount of public and private property illegally. It is a pervasive issue with significant societal and economic implications [Barson *et al.*, 1996], resulting in substantial financial losses and compromised user personal safety.



Phone	Receiver	Imei	Location
#A	#B	Imei_1	Location_1
#B	#A	Imei_1	Location_1
#B	#A	Imei_1	Location_1
#A	#C	Imei_2	Location_2
#A	#D	Imei_3	Location_2
#A	#E	Imei_2	Location_3

Figure 1: A schematic illustration of a mutation fraud. The user communication graph illustrates the connectivity established by a user's calls. VOC records capture call metadata such as start time, duration, and direction (incoming/outgoing). Notably, the conspicuous mutation is observed following the 3rd call. The mutation includes changes in the user communication graph (from friends (F) to strangers (S)) and changes in various call metadata. This indicates that the user has started engaging in telecom fraud after the 3rd call.

Due to the trust granted by communication service providers to users with a history of good communication behavior, fraudsters often resort to buying or renting communication cards from legitimate users to carry out fraudulent activities. Furthermore, fraudsters may also simulate a large volume of normal call activities to conceal a small amount of fraudulent call behavior. This leads to the phenomenon of

[*] Corresponding Author

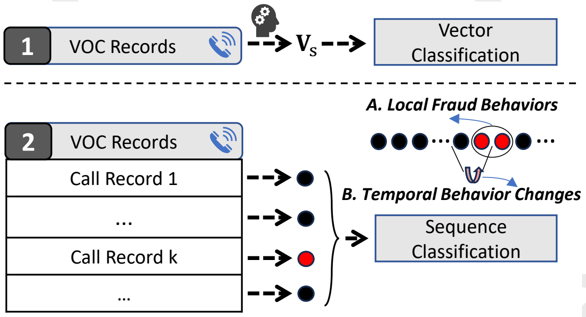


Figure 2: Different modeling methods for VOC. The points represent the raw features, and $\mathbf{V}_S$ represents the statistical feature extracted using manually designed rules. Red dots represent fraudulent behavior, while black dots represent normal behavior. (1) illustrates the current pipeline framework. (2) presents our proposed method. Our method focuses on capturing two key patterns of mutation fraud, considering both static and dynamic perspectives.

mutation fraud. Mutation fraud refers to fraudulent activities interleaved with legitimate behaviors within a short time window. Unlike persistent fraud, mutation fraud exhibits abrupt shifts in communication patterns, such as sudden changes in call recipients (e.g., from known contacts to strangers) or temporal anomalies (e.g., short-duration calls clustered in unusual hours), as illustrated in Figure 1.

Existing methods assume that the users under detection exhibit stable behavioral patterns and directly convert the entire Voice of Call (VOC) records of the user into a hand-designed statistical feature vector (average call duration, total number of calls, etc.) [Hu *et al.*, 2022; Hu *et al.*, 2024; Ji *et al.*, 2020]. Subsequently, they solve the problem by using vector classification approaches. This classification approach based on statistical features is inadequate for adapting to complex and evolving fraud patterns. In the long-term adversarial scenario between fraudsters and regulators, fraud techniques are constantly updated. Once fraud patterns change (e.g., mutation fraud), the existing statistical feature extraction methods are likely to become ineffective.

There are three major challenges in the task of mutation telecommunication fraud detection: (1) Research in this field is currently limited by the absence of datasets created specifically for mutation telecommunication fraud. (2) Mutation fraud has its inherent patterns, but existing methods based on manual statistical features make it difficult to capture these patterns. (3) Telecom fraud detection is a complex task that requires the integration of multiple sources of information. However, existing methods often lack the capability to fuse multi-source information effectively.

To address the first challenge, we have curated a dataset named MutationTeleFraud, specifically containing mutation fraud data. This dataset is obtained from a telecommunication carrier after removing PII (Personally Identifiable Information) and sensitive data and is made publicly accessible to facilitate further research. In addressing the second challenge, we introduce a deep neural method called MutationGuard, specifically designed to handle the intricate pattern detection issues associated with mutation fraud. The main difference between our approach and previous methods for VOC modeling is illustrated in Figure 2. Although mutation fraud is very covert, it still has two significant patterns: 1. From a static perspective, there must be local fraud behaviors; 2. From a dynamic perspective, there are significant temporal behavior changes. Our method processes each call record into a vector, with the entire VOC records forming a list of vectors, and employs a sequence classification method following the above two insights to detect telecom fraud. Existing methods cannot effectively utilize the above two insights because manual features cannot effectively focus on local behaviors and reflect the temporal changes. Regarding the third challenge, we consider users under investigation to be central nodes, with all users communicating with them treated as neighboring nodes. For users engaged in mutation fraud, their communication often undergoes significant changes. For example, an ordinary individual with regular social connections would typically have dense communication with family and friends. However, if their mobile SIM card is lost or sold for fraudulent purposes, their calling users would include both friends and a large number of strangers, as illustrated in Figure 1. Considering that the message-passing mechanism can effectively discover relevant patterns [Zhao *et al.*, 2021; Zhou *et al.*, 2020; Scarselli *et al.*, 2004], we effectively capture the evolving patterns within user communication graphs by employing a graph attention mechanism. In summary, the main contributions are as follows:

1. Our study represents the first investigation into the phenomenon of mutation telecommunication fraud, offering empirical data from real-world scenarios coupled with comprehensive analytical insights.
2. The proposed MutationGuard facilitates multi-source information fusion and constitutes a significant advancement in detecting mutation fraud. MutationGuard not only captures local fraud behaviors and temporal behavior changes in VOC records but also discerns evolving patterns within user communication graphs.
3. Our MutationGuard shows a significant improvement on the proposed MutationTeleFraud dataset and the open-source dataset. Specifically, experiment results reveal a significant improvement in the AUC score by 1.52% and the F_1 score by 1.36% on the proposed telecommunication fraud dataset.

2 Related Work

Rule-based Methods. Early research on telecom fraud detection primarily employs rule-based approaches, emphasizing the manual design of filtering rules based on expert knowledge. For instance, Hilas [2009] utilizes professional expertise and data mining techniques to design an expert system for telecom fraud detection. Prakash *et al.* [2010] employ the Singular Value Decomposition (SVD) method to analyze anomalous patterns in large-scale telephone communication networks. Liu *et al.* [2017] introduce a metric named 'contrast suspiciousness,' utilizing graph topology information to identify potential fraudsters.

Machine Learning Methods. With the advent of machine learning technology, researchers have shifted towards

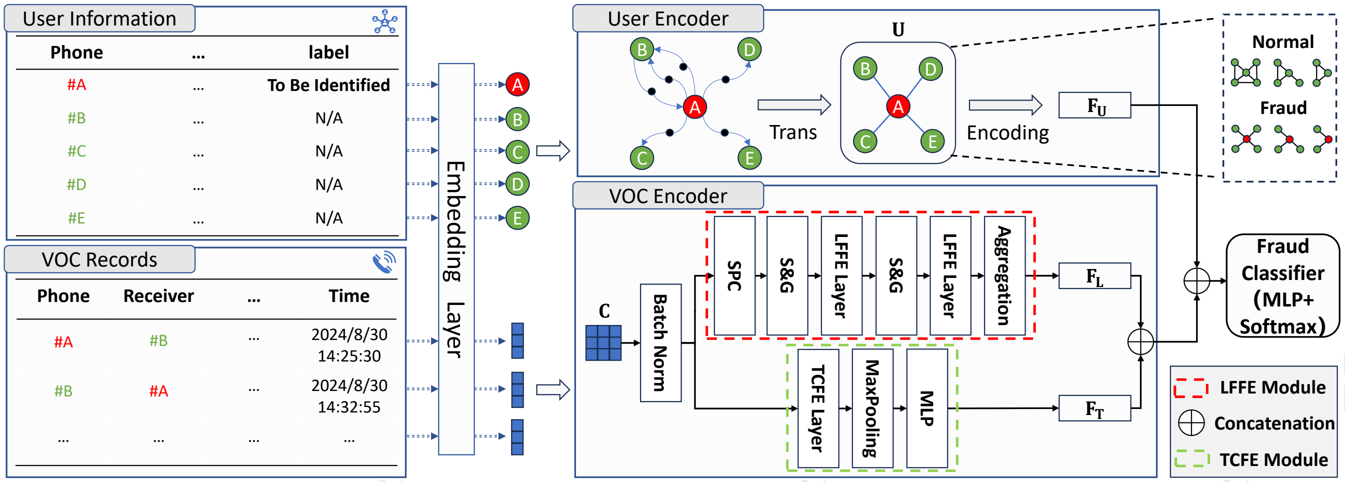


Figure 3: MutationGuard comprises two main components: the User Encoder and the VOC Encoder. We convert a directed graph into an undirected graph by treating multiple directed edges between two nodes as a single undirected edge. The LFFE module first achieves better feature locality through SPC, then extracts multiple regions of interest through S&G, and subsequently captures local fraud features through the LFFE layer. The TCFE module performs time sequence encoding to capture temporal behavior changes.

manually designing features and employing classical machine learning methods to classify fraudulent telephone numbers. Dong *et al.* [2004] extract 55 feature attributes from user call records and utilize a Support Vector Machine (SVM) with a Radial Basis Function (RBF) kernel to predict fraudulent users. Xing *et al.* [2020] employ random forests as a benchmark method, outperforming SVM with an RBF kernel on fraud detection datasets. To overcome the high bias of the dataset, Arafat *et al.* [2019] apply ensemble techniques of classifiers. Despite the computational efficiency of classical machine learning methods, their constrained fitting capability poses challenges in adapting to intricate patterns.

Deep Learning Methods. Deep learning methods have been widely used in an adversarial scenario nowadays [Zhou *et al.*, 2024], including in the field of telecommunication fraud detection [Ravi *et al.*, 2022; Hu *et al.*, 2024]. Wahid *et al.* [2023] introduce a real-time fraud detection method utilizing a Neural Factorization Autoencoder (NFA). Ji *et al.* [2020] employ a Multi-Range Gated Graph Neural Network for telecom fraud detection. Hu *et al.* [2022] utilize subscriber synergy behavior to reconstruct connectivity, thereby bridging the gap between sparse connectivity data and graph machine learning. These methods commonly require manually extracting statistical features from telecommunication users' VOC information and various behavioral records. To fully utilize the temporal features in VOC records, Zhen and Gao [2023] introduce CDR2IMG that leverages temporal periodicity to enhance fraud detection. Specifically, they compress the call duration feature into a 2D image and employ Convolutional Neural Networks (CNN) to extract latent fraud features embedded in the image. These existing approaches heavily depend on predefined feature engineering methods, making existing methods challenging to adapt to complex mutation fraud scenarios. In contrast, our proposed MutationGuard directly utilizes the raw data from call records as input, allowing the method to better capture complex mutation patterns.

3 Our Method

3.1 Overview

Figure 3 shows the architecture of our MutationGuard, which is based on multi-source information joint learning. MutationGuard follows the design principles presented in the Introduction section, using the VOC Encoder to capture local fraud behaviors and temporal behavior changes, and then leveraging the User Encoder to incorporate user interactions.

3.2 Embedding Layer

User Feature Initialization. User features are encoded into a vector encompassing: static attributes (e.g., SMS usage, app preferences) and temporal distribution features (e.g., call time distribution). Ultimately, we obtain the user feature matrix $\mathbf{U} \in \mathbb{R}^{M \times d_{\mathrm{u}}}$, where M denotes the number of nodes in the graph, and d_{u} denotes the dimensionality of features.

VOC Feature Initialization. VOC refers to detailed records about communication activities generated by the telephone system. These records encompass details such as the call start time, call duration, call type (e.g., incoming or outgoing), etc. Assuming a target user to be identified generates a set of records with a length of N over a period, we obtain the VOC feature representation matrix $\mathbf{C} \in \mathbb{R}^{N \times d_{\mathrm{c}}}$, where d_{c} represents the dimensionality of each VOC record feature.

3.3 VOC Encoder

To capture fraud behaviors, we design two corresponding modules inspired by Pointnet++ [Qi *et al.*, 2017] and BiLSTM: the **Local Fraud Feature Extraction (LFFE) Module** and the **Temporal Change Feature Extraction (TCFE) Module**. The components are shown in Figure 3.

Locality. To achieve better local fraud feature representation, it's essential to reconsider the locality within human activity data. Organizing data based on the principle of local similarity enables more effective extraction of local features. If data

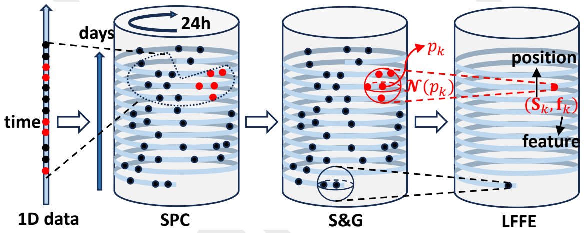


Figure 4: The schematic diagram illustrates the process of SPC, S&G, and LFFE. Black dots represent normal behavior, and red dots represent fraudulent behavior. The fraudulent behaviors in VOC records can be better aggregated in a local area after the SPC step.

isn't organized according to this principle, during local aggregation, the model might blend distinct fraud patterns with normal behavior, resulting in higher false positives or overlooking critical fraud features. Image data naturally lends itself to the concept of locality through geometric relations, where neighboring pixels often exhibit similar values due to spatial proximity. VOC data is 1D time-series data that exhibits local behavioral similarity within adjacent time windows. However, humans also engage in similar activities at the same time on different days, which cannot be reflected in the 1D time series, thus losing some local information. Therefore, we need to address the question: How can we reorganize VOC records so that reorganized human activity behaviors exhibit better local similarity, facilitating the detection of local fraudulent activities? Zhen and Gao [2023] uses a 2D time grid to solve this problem, but the sparsity of features is caused by fixed intervals. Unlike them, we conceptualize time as a spiral structure to solve this problem.

Spatial Position Calculation (SPC). Spatial Position Calculation provides spatial position inputs for the LFFE Layer in the VOC encoder. Specifically, we transform each timestamp in VOC records into coordinates on a cylindrical surface to ensure a reasonable distance measurement between these embeddings. The projection of human behavior data onto a cylindrical surface is a deliberate design choice that leverages the natural periodicity of time to provide better locality. In further detail, time can be conceptualized as a spiral structure when viewed in a cylindrical projection. The cylindrical surface allows for the representation of time as progressing both in the vertical direction and along the horizontal axis of the cylindrical surface. As we move through time, the data forms a spiral that encircles the cylinder. Each complete rotation represents a full day, and the height of the spiral indicates the temporal progression of the days themselves, as shown in Figure 4. We not only ensure that feature vectors corresponding to temporally adjacent times on the same day (along the horizontal axis of the cylindrical surface: e.g., 2024-10-1 20:00:00 and 2024-10-1 20:01:00) are spatially adjacent but also that feature vectors corresponding to adjacent times on different days are spatially adjacent (in the vertical direction: e.g., 2024-10-1 20:00:00 and 2024-10-2 20:00:00). Clearly, by selecting the same number of feature points within any local neighborhood, our proposed data organization method exhibits higher local similarity compared to the simple 1D sequence feature similarity, thereby achieving better locality. For any VOC record (at time t on the x-th day), where $0 \leq t{<}24$ and $x \in \mathbb{N}$ (we set the earliest VOC record time at x = 0.), we obtain its cylindrical coordinates:

$$r = 1 \tag{1}$$

$$\theta = (t/24) * 2\pi \tag{2}$$

$$z = \text{Normalize}(x + t/24) \tag{3}$$

We normalize $x + t/24$ to the range $[-1, 1]$ and convert the cylindrical coordinates to Cartesian coordinates for subsequent Euclidean distance calculation in the S&G step. Then we get the spatial position matrix $\mathbf{S} \in \mathbb{R}^{N\times3}$ for VOC records after SPC, where N represents the number of calls.

Sampling and Grouping (S&G). The goal of the Sampling and Grouping (S&G) layer is to extract the most informative and relevant local fraud features from VOC data to detect patterns of telecom fraud. This is achieved by first selecting a representative subset of points and then grouping them based on their proximity in time, as shown in Figure 4.

Sampling. Given the VOC feature representation matrix $\mathbf{C} \in \mathbb{R}^{N\times d_c}$ and the spatial position matrix $\mathbf{S} \in \mathbb{R}^{N\times3}$, where N is the number of calls. Each point $p_i = (\mathbf{S}_i, \mathbf{C}_i)$ represents a moment of activity. Thus, the complete set of points can be represented as: $\mathcal{P} = \{p_1, p_2, \ldots, p_N\}$. We apply the farthest point sampling (FPS) method to sample a subset of the points. We aim to sample a subset $\mathcal{P}_{\text{sample}}$ of points from $\mathcal{P}$, where $|\mathcal{P}_{\text{sample}}| = K$ and K represents the number of selected samples. Using FPS, we iteratively select points that are maximally distant from already selected ones. The sampling process can be mathematically formulated as:

$$p_k = \arg \max_{p_i \in \mathcal{P}\setminus\mathcal{P}_{\text{sample}}} \left(\min_{p_j \in \mathcal{P}_{\text{sample}}} \|p_i - p_j\|_2 \right) \tag{4}$$

where $\|p_i - p_j\|_2$ is the Euclidean distance between points p_i and p_j, and $\mathcal{P}_{\text{sample}}$ is the selected points. This procedure ensures that the selected points $\mathcal{P}_{\text{sample}}$ are representative.

Grouping. After selecting the representative points, we group the sampled points based on their temporal proximity. For each sampled point p_k, we define a neighborhood $\mathcal{N}(p_k)$ consisting of points within a radius r.

$$\mathcal{N}(p_k) = \{p_j \in \mathcal{P} \mid \|p_j - p_k\|_2 \leq r\} \tag{5}$$

This step ensures that each point is grouped with others that exhibit similar behaviors, allowing the model to capture localized patterns of user activity.

We empirically determined optimal sampling ratios (K/N) and grouped radius through a grid search on the validation set.

LFFE Layer. Once the neighborhoods are formed, we aggregate the features within each group, as shown in Figure 4. A common approach is to apply max pooling to capture the most significant features within each neighborhood. The aggregated feature $\mathbf{f}_k$ for each neighborhood $\mathcal{N}(p_k)$ is:

$$\mathbf{f}_k = \varphi(\{\text{MLP}(\mathbf{C}_j)) \mid p_j = (\mathbf{S}_j, \mathbf{C}_j), p_j \in \mathcal{N}(p_k)\}) \tag{6}$$

where $\varphi(\cdot)$ performs the element-wise maximum operation over the features in the neighborhood $\mathcal{N}(p_k)$. Each input feature point p_k is updated as $p_k^{new} = (\mathbf{S}_k, \text{MLP}(\mathbf{f}_k))$.

Aggregation. The aggregation Layer aggregates features directly across all points to obtain a comprehensive feature representation of local fraud behaviors in VOC records:

$$\mathbf{F}_{\mathrm{L}} = \psi\left(\mathrm{MaxPooling}(\{\mathrm{MLP}(\mathbf{f}_{k'}) \mid k' = 1, \ldots, K'\})\right) \quad (7)$$

Where K' represents the total number of feature points input to the Aggregation layer, and $\psi(\cdot)$ represents the feature transformation function, which is a Multi-Layer Perceptron.

TCFE Layer. Due to the inability of the LFFE module to perceive the order of data, the TCFE layer employs a bidirectional Long Short-Term Memory (BiLSTM) to incorporate temporal dynamics perception. It allows the model to learn how current actions might relate to future fraudulent tendencies. For instance, a fraudster may engage in a series of small, unremarkable actions that gradually escalate into a full-blown scam. By analyzing from a dynamic perspective, the TCFE Layer can detect pattern changes in behavior. The final output feature representation of the TCFE module is $\mathbf{F}_{\mathrm{T}}$.

3.4 User Encoder

Generally, there are significant differences between the communication graphs of normal users and those of fraudulent users. **Differences in user features** include variations in data usage expenses and app usage patterns. **Differences in graph features** are evident as the user groups contacted by normal users and fraudulent users often differ significantly, leading to substantial differences in the overall features of the graph.

Our approach is capable of directly encoding any available features into the node attributes of users and utilizes the User Encoder to capture various latent features, highlighting the importance of incorporating multi-source information in telecom fraud detection. We conduct L-layer graph attention mechanism to obtain user interaction pattern features $\mathbf{F}_{\mathrm{U}}$.

$$\mathbf{e}_{ij} = \mathrm{LeakyReLU}\left(\mathbf{a}^{\top}[\mathbf{W}\mathbf{u}_i^l || \mathbf{W}\mathbf{u}_j^l]\right), \quad (8)$$

$$\alpha_{ij} = \frac{\exp(\mathbf{e}_{ij})}{\sum_{k \in \mathcal{N}_i} \exp(\mathbf{e}_{ik})}, \quad (9)$$

$$\mathbf{u}_i^{l+1} = \sigma\left(\sum_{j \in \mathcal{N}_i} \alpha_{ij} \mathbf{W}\mathbf{u}_j^l\right), \quad (10)$$

where $\mathbf{u}_i^l$ represents the input feature vector for node i in layer l, $\mathbf{W}$ is the weight matrix, $\mathcal{N}_i$ represents the neighborhood of node i in the graph, $\mathbf{a}$ is the attention weight vector, σ is the activation function. Then we get the graph pattern feature:

$$\mathbf{F}_{\mathrm{U}} = \mathbf{u}_0^L. \quad (11)$$

4 Experiments

In this section, we evaluate the performance of MutationGuard. We make all of our source code and datasets publicly available to facilitate future study[1]. The research data were provided by China Mobile Communications Group Shaanxi Co., Ltd. The identification of fraudsters was achieved through various measures after long-term efforts by China Mobile Communications Group Shaanxi Co., Ltd.

Differences	Sichuan Dataset	MutationTeleFraud
CDR	ALL	VOC&CNS
Mutation Fraud	Missing	Included
User Graph	Missing	Included

Table 1: Comparison of the Sichuan Telecom Fraud Dataset and the proposed Dataset. 'ALL' represents the aggregation of APP, SMS (short messages), VOC, and CNS (user consumption).

4.1 Datasets

We first conduct experiments on the Sichuan telecom fraud dataset[2] including 4587 subscribers with complete features from the 2020 Digital Sichuan Innovation Competition, organized by the Sichuan Provincial Big Data Center in China. Some users in this dataset have missing attributes. To ensure fair comparison, we test all baseline methods using users with complete attributes. We also conduct experiments on the introduced dataset MutationTeleFraud. This dataset comprises 9874 subscribers, documenting their VOC records and other relevant user information generated between August 1, 2023, and September 14, 2023. The dataset encompasses 1849 normal fraudulent users, 388 users exhibiting mutation fraud, and 7637 normal users. The mutation fraudsters are defined as users who: (1) maintained normal communication behaviors for at least 12 consecutive months; (2) were flagged by both telecom operators and public security agencies for participating in fraud incidents during the latest month. Following the previous work [Hu *et al.*, 2024], the ratio of the training set, validation set, and testing set in both datasets is 60%, 20%, and 20%, respectively. The main differences between the proposed dataset and the publicly available Sichuan dataset are illustrated in Table 1.

4.2 Evaluation Metrics and Baselines

Following the previous work [Zhen and Gao, 2023], we evaluate the performance with AUC, macro Recall, macro Precision, and macro F_1. The default threshold for prediction is 0.5. To gain a comprehensive understanding of the performance of regular fraud and mutation fraud data, we also provide results on Accuracy for the two types of fraudulent activities. The baselines for the Sichuan telecom fraud dataset we compared include (1) machine learning methods: Support Vector Machine (SVM) [Cortes and Vapnik, 1995], Logistic Regression (LR) [Cox, 1972], Random Forest (RF) [Breiman, 2001]; (2) deep learning methods: Multilayer Perceptron (MLP), GCN [Kipf and Welling, 2016], GAT [Veličković *et al.*, 2017], CDR2IMG [Zhen and Gao, 2023], GAT-COBO [Hu *et al.*, 2024]. We follow the hyperparameter settings in the provided official implementations [Zhen and Gao, 2023; Hu *et al.*, 2024].

For the construction of manual features of VOC, we employ three approaches: '-B', '-W', and '-D', representing basic feature engineering [Hu *et al.*, 2022], basic features combined with weekly-level features, and basic features combined with daily-level features, respectively. Since previous work does not consider the presence of mutation fraud,

[1] https://github.com/nlgandnlu/MutationGuard

[2] https://aistudio.baidu.com/aistudio/datasetdetail/40690

Methods	AUC	Recall	Precision	F_1	NorFraud-A	MutFraud-A
SVM (Linear)	0.9493	0.9083	0.8762	0.8908	0.8652	0.4051
SVM (Poly)	0.9520	0.9207	0.8738	0.8943	0.8491	0.4051
SVM (Rbf)	0.9503	0.9113	0.8712	0.8890	0.8544	0.3797
SVM (Sigmoid)	0.8273	0.8362	0.7736	0.7979	0.7116	0.0506
LR	0.9461	0.9002	0.8660	0.8814	0.8544	0.3544
RF	0.9562	0.9313	0.8685	0.8948	0.8248	0.4177
MLP-B	0.9498	0.8697	0.9097	0.8875	0.8464	0.4051
GCN-B	0.9589	0.8868	0.9145	0.8996	0.8625	0.5316
GAT-B	0.9608	0.9000	0.9160	0.9076	0.8733	0.6456
GAT-W	0.9732	0.9090	0.9328	0.9202	0.8679	0.7215
GAT-D	0.9746	0.9140	0.9396	0.9260	0.8706	0.7468
CDR2IMG	0.9656	0.8870	0.9025	0.8944	0.8329	0.7215
GAT-COBO	0.9576	0.8816	0.9134	0.8961	0.8652	0.4557
Ours	**0.9898**†	**0.9344**†	**0.9450**†	**0.9396**†	**0.8922**†	**0.8861**†
Ours w/o LFFE Module	0.9790	0.9201	0.9369	0.9282	0.8733	0.8228
Ours w/o TCFE Module	0.9855	0.9205	**0.9450**	0.9320	0.8652	0.7975
Ours w/o VOC Encoder	0.9444	0.8193	0.8914	0.8478	0.7655	0.2025
Ours w/o User Encoder	0.9861	0.9298	0.9396	0.9346	**0.8922**	0.8228

Table 2: Results on the proposed MutationTeleFraud dataset. The best performance is highlighted in bold. ”NorFraud-A” represents the Accuracy score on normal fraud data. ”MutFraud-A” represents the Accuracy score on mutation fraud data. † denotes our method achieves significant improvements over all existing baselines in a paired t-test with p-value <0.05. The statistical values (mean ± standard deviation of 5 experiments) of our method on the main metrics are: AUC is 0.9899 ± 0.0003 and F_1 is 0.9364 ± 0.0071.

the basic feature engineering method directly extracts stable long-term fraud-related features. However, for mutation fraud, these features are less effective because the presence of fraud is short-term, and the average call duration within a week or a day may be abnormal but appears normal in the long term. Therefore, we further supplement weekly-level and daily-level features for comparison. Specifically, the ‘-B’ method extracts 42-dimensional features from VOC without considering temporal granularity, and ‘-W’ and ‘-D’ additionally extract 27-dimensional features respectively.

4.3 Experiment Settings

We use the Adam optimizer with the learning rate being 0.001 for the User Encoder, and 0.0001 for the other learnable modules. The dropout rate is 0.1 and the batch_size is 16. The first-level sampling retains 70% of points (ratio=0.7) with grouping radius r=0.5, while the second-level sampling uses 30% of remaining points (ratio=0.3) with r=0.2. Other detailed hyperparameters can be found in the open-source code.

4.4 Experiment on MutationTeleFraud Dataset

The evaluation results on the MutationTeleFraud Dataset are presented in Table 2.

Firstly, we see that the GCN-B method outperforms the MLP-B method, achieving an improvement of 1.61% points on normal fraud data and 12.65% points on mutation fraud data. The difference between the two methods lies in the introduction of the user communication graph, indicating that the user communication graph can significantly aid in fraud detection, particularly in the detection of mutation fraud.

Secondly, we see our method outperforms all baselines in all metrics. Specifically, MutationGuard shows a 1.36% improvement in the F_1 score and a 1.52% improvement in the AUC score. Particularly, on mutation fraud data, our method shows a significant improvement of 13.93%.

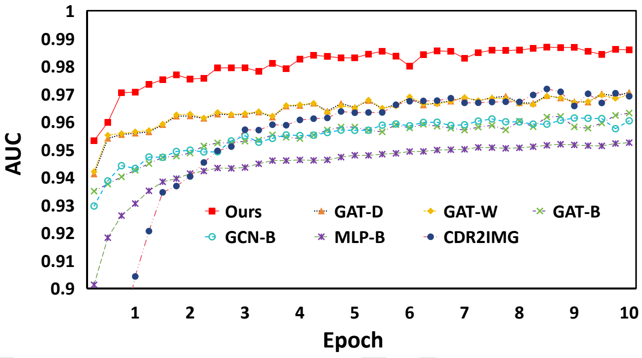


Figure 5: The AUC curves on the validation set during the training process of the baseline methods. During the training process, we measure the AUC score every 100 steps.

Finally, comparing the three baseline methods GAT-B, GAT-W, and GAT-D, we see that a more fine-grained feature engineering approach GAT-D can effectively improve the detection results of mutation fraud by 10.12%. This further indicates the necessity to effectively utilize mutation features in mutation fraud. Our method leverages these features by effective neural modeling, resulting in a further improvement of 13.93% compared to GAT-D’s manual feature engineering.

To intuitively analyze the performance differences among these methods, we further provide the AUC curves of deep learning-based baseline methods during the training process on the validation set, as shown in Figure 5. The GAT-COBO, being a graph node classification method, is trained for 2000 epochs, so it is not included in the figure. We observe a high

Methods	AUC	Recall	Precision	F_1	NorFraud-A	MutFraud-A
1-D CNN	0.9597	0.8860	0.8935	0.8896	0.8437	0.6962
TCFE Module	0.9673	0.8938	0.9182	0.9052	0.8302	0.7595
LFFE Module	0.9821	0.9224	0.9285	0.9254	0.8868	0.8101
TCFE&LFFE	**0.9861**	**0.9298**	**0.9396**	**0.9346**	**0.8922**	**0.8228**

Table 3: VOC modeling test results on the proposed MutationTeleFraud dataset. The best performance is highlighted in bold. ”NorFraud-A” represents the Accuracy score on normal fraud. ”MutFraud-A” represents the Accuracy score on mutation fraud.

Methods	AUC	Recall	Precision	F_1
SVM (Linear)	0.9022	**0.8418**	0.7833	0.8059
SVM (Poly)	0.8655	0.7913	0.7651	0.7768
SVM (Rbf)	0.8998	0.8340	0.8051	0.8181
SVM (Sigmoid)	0.8987	0.8141	0.7734	0.7904
LR	0.9026	0.8036	0.8862	0.8361
RF	0.9125	0.7713	**0.9312**	0.8220
CDR2IMG	0.9012	0.8120	0.8896	0.8432
GAT-COBO	0.9058	0.8149	0.8907	0.8455
Ours	**0.9208**†	0.8189	0.9110	**0.8549**†

Table 4: Results on the open Sichuan telecom fraud dataset. The best performance is highlighted in bold. † denotes our method achieves significant improvements over all existing baselines in a paired t-test with p-value <0.05.

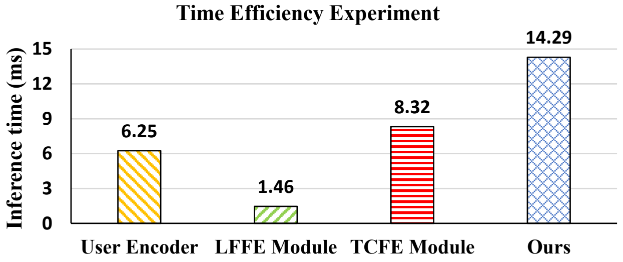


Figure 6: Results of the computational efficiency evaluation. The inference time represents the average inference time of a single sample (average on more than 10,000 samples). The methods are executed in a Python 3.8.0 environment, with hardware specifications including 1 Tesla V100 GPU.

consistency between the performance results reported on the testing set and those on the validation set, further confirming the effectiveness of our proposed method.

4.5 Ablation Study

Table 2 presents the evaluation results of the ablation study. We see some important conclusions from the results.

- **The method's ability to capture local fraud features and temporal change features is crucial for the recognition of mutation fraud.** Table 2 shows that both modules for individual ablation and combined ablation exhibit significant performance degradation.
- **The GAT trained on the user graph can enhance the method's ability to detect mutation fraud.** We see that the score given by the ablated version of ”without User Encoder” has decreased by 0.5% and 0.37% in the F_1 score and AUC score. The decline is most severe in mutation data, amounting to a decrease of 6.33%.

4.6 VOC Modeling Test

To explore the neural modeling method of VOC records, we test the following deep learning-based methods in Table 3: 1-D CNN, TCFE Module, LFFE Module and TCFE&LFFE. We see TCFE&LFFE achieves the best results.Both 1-D CNN and LFFE structures are adept at discovering local fraud features in data. Due to the reasonable utilization of temporal periodicity, our proposed Time-aware LFFE Module performs better. We achieve a more effective neural method by combining the Time-aware LFFE Module with TCFE Module.

4.7 Experiment on Sichuan Fraud Dataset

The evaluation results on the open Sichuan telecom fraud dataset are shown in Table 4. It is important to note that the publicly available dataset does not provide communication feature information for the receivers, and we cannot utilize information from the user communication graph. Instead, we directly employ a Multilayer Perceptron (MLP) to extract user features. Our approach demonstrates a pronounced advantage compared to other classical and benchmark methods. Results show that our MutationGuard achieves the best performance in terms of AUC and F_1 metrics.

4.8 Computational Efficiency Evaluation

The proposed method consists of three main components: the User Encoder, LFFE Module, and TCFE Module. We test the average time consumption of our method and each component, as shown in Figure 6. The majority inference cost is from two modules: the TCFE Module and the User Encoder. Our method shows remarkable scalability and efficiency, enabling inference of more than 250,000 users in one hour on a single GPU, making it suitable for applications.

5 Conclusion

This paper addresses the issue of mutation telecommunication fraud, a complex form of fraudulent activity characterized by intermittent and abrupt patterns. We present MutationTeleFraud enriched with mutation fraud instances, which fills a critical absence in the available resources for researching this specific type of fraud. We also contribute by introducing MutationGuard, a deep neural network designed explicitly for the detection of mutation telecommunication fraud. Future work includes (1) integrating real-time streaming data for instant fraud alerts, (2) extending MutationGuard to cross-domain fraud detection (e.g., financial scams), and (3) addressing ethical challenges in user privacy preservation.

References

[Arafat *et al.*, 2019] Mais Arafat, Abdallah Qusef, and George Sammour. Detection of wangiri telecommunication fraud using ensemble learning. In *2019 IEEE Jordan International Joint Conference on Electrical Engineering and Information Technology (JEEIT)*, pages 330–335. IEEE, 2019.

[Barson *et al.*, 1996] P Barson, S Field, N Davey, G McAskie, and R Frank. The detection of fraud in mobile phone networks. *Neural Network World*, 6(4):477–484, 1996.

[Breiman, 2001] Leo Breiman. Random forests. *Machine learning*, 45:5–32, 2001.

[Cortes and Vapnik, 1995] Corinna Cortes and Vladimir Vapnik. Support-vector networks. *Machine learning*, 20:273–297, 1995.

[Cox, 1972] David R Cox. Regression models and life-tables. *Journal of the Royal Statistical Society: Series B (Methodological)*, 34(2):187–202, 1972.

[Dong *et al.*, 2004] Wang Dong, Wang Quan-yu, Zhan Shou-yi, Li Feng-xia, and Wang Da-zhen. A feature extraction method for fraud detection in mobile communication networks. In *Fifth World Congress on Intelligent Control and Automation (IEEE Cat. No. 04EX788)*, volume 2, pages 1853–1856. IEEE, 2004.

[Hilas, 2009] Constantinos S Hilas. Designing an expert system for fraud detection in private telecommunications networks. *Expert Systems with applications*, 36(9):11559–11569, 2009.

[Hu *et al.*, 2022] Xinxin Hu, Hongchang Chen, Shuxin Liu, Haocong Jiang, Guanghan Chu, and Ran Li. Btg: A bridge to graph machine learning in telecommunications fraud detection. *Future Generation Computer Systems*, 137:274–287, 2022.

[Hu *et al.*, 2024] Xinxin Hu, Haotian Chen, Junjie Zhang, Hongchang Chen, Shuxin Liu, Xing Li, Yahui Wang, and Xiangyang Xue. Gat-cobo: Cost-sensitive graph neural network for telecom fraud detection. *IEEE Transactions on Big Data*, 2024.

[Ji *et al.*, 2020] Shuyun Ji, Jinglin Li, Quan Yuan, and Jiawei Lu. Multi-range gated graph neural network for telecommunication fraud detection. In *2020 International Joint Conference on Neural Networks (IJCNN)*, pages 1–6. IEEE, 2020.

[Kipf and Welling, 2016] Thomas N Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907*, 2016.

[Liu *et al.*, 2017] Shenghua Liu, Bryan Hooi, and Christos Faloutsos. Holoscope: Topology-and-spike aware fraud detection. In *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*, pages 1539–1548, 2017.

[Prakash *et al.*, 2010] B Aditya Prakash, Ashwin Sridharan, Mukund Seshadri, Sridhar Machiraju, and Christos Faloutsos. Eigenspokes: Surprising patterns and scalable community chipping in large graphs. In *Pacific-Asia conference on knowledge discovery and data mining*, pages 435–448. Springer, 2010.

[Qi *et al.*, 2017] Charles Ruizhongtai Qi, Li Yi, Hao Su, and Leonidas J Guibas. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. *Advances in neural information processing systems*, 30, 2017.

[Ravi *et al.*, 2022] Akshaya Ravi, Mounira Msahli, Han Qiu, Gerard Memmi, Albert Bifet, and Meikang Qiu. Wangiri fraud: Pattern analysis and machine-learning-based detection. *IEEE Internet of Things Journal*, 10(8):6794–6802, 2022.

[Scarselli *et al.*, 2004] Franco Scarselli, Ah Chung Tsoi, Marco Gori, and Markus Hagenbuchner. Graphical-based learning environments for pattern recognition. In *Structural, Syntactic, and Statistical Pattern Recognition: Joint IAPR International Workshops, SSPR 2004 and SPR 2004, Lisbon, Portugal, August 18-20, 2004. Proceedings*, pages 42–56. Springer, 2004.

[Veličković *et al.*, 2017] Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Lio, and Yoshua Bengio. Graph attention networks. *arXiv preprint arXiv:1710.10903*, 2017.

[Wahid *et al.*, 2023] Abdul Wahid, Mounira Msahli, Albert Bifet, and Gerard Memmi. Nfa: A neural factorization autoencoder based online telephony fraud detection. *Digital Communications and Networks*, 2023.

[Xing *et al.*, 2020] Jian Xing, Miao Yu, Shupeng Wang, Yaru Zhang, and Yu Ding. Automated fraudulent phone call recognition through deep learning. *Wireless Communications and Mobile Computing*, 2020:1–9, 2020.

[Zhao *et al.*, 2021] Lingxiao Zhao, Wei Jin, Leman Akoglu, and Neil Shah. From stars to subgraphs: Uplifting any gnn with local structure awareness. *arXiv preprint arXiv:2110.03753*, 2021.

[Zhen and Gao, 2023] Zhen Zhen and Jian Gao. Cdr2img: A bridge from text to image in telecommunication fraud detection. *Computer Systems Science & Engineering*, 47(1), 2023.

[Zhou *et al.*, 2020] Jie Zhou, Ganqu Cui, Shengding Hu, Zhengyan Zhang, Cheng Yang, Zhiyuan Liu, Lifeng Wang, Changcheng Li, and Maosong Sun. Graph neural networks: A review of methods and applications. *AI open*, 1:57–81, 2020.

[Zhou *et al.*, 2024] Ziyang Zhou, Pinghui Wang, Zi Liang, Ruofei Zhang, and Haitao Bai. Pair: Pre-denosing augmented image retrieval model for defending adversarial patches. In *Proceedings of the 32nd ACM International Conference on Multimedia*, pages 5771–5779, 2024.