Analyzing User Patterns via Multi-Level Features for Telecommunications Fraud Detection

Anonymous Authors[1]

Abstract

Telecommunications fraud poses a pervasive and serious threat, creating significant risks for both service providers and consumers. It involves identifying high-risk users, those engaged in fraudulent activities such as scam calls, identity theft, and coordinated fraud rings, by detecting abnormal call patterns and suspicious behaviors. Existing research primarily analyzes call records or manually extracts simplistic user features, often overlooking user relationships, leading to misclassification of users with similar calling patterns, such as salespeople. To address this, we propose a Multi-Level Feature Extraction (MLFE) method that automatically captures intra-user behaviors across multiple call records using a set function and employs graph neural networks to model inter-user relationships. We evaluate our approach on real-world telecommunications data from a network operator, achieving high accuracy in detecting fraudulent users. Additionally, we validate its generalization on a public dataset, confirming its robustness. Our findings reveal that high-risk users exhibit distinct temporal, stability, regional, and call target patterns, offering valuable insights for enhancing fraud detection systems.

1. Introduction

Telecommunications fraud (Rosset et al., 1999; Becker et al., 2010; Zhao et al., 2018) has become a widespread and critical issue, posing significant threats to both service providers and consumers. In the telecom domain, fraud refers to any illegal use of infrastructure and communication services, such as misuse of subscriptions or services, to generate illicit profits and disrupt the legal flow of funds and fees (Becker et al., 2010). Specifically, telecommunications fraud encompasses activities like telephone fraud, billing fraud, phishing, and call hijacking, where individuals exploit telecom systems through deception, misuse, or identity theft. These actions often target individuals or businesses through impersonation and fraudulent calls, leading to substantial financial losses and emotional distress for victims. The impact of fraud goes beyond monetary damage, severely eroding trust in communication systems and service providers.

The scale and complexity of this problem have been increasing at an alarming rate (Chadyšas et al., 2022; Ni & Yu, 2022; Lu et al., 2020). According to the Communications Fraud Control Association (CFCA), global telecommunications fraud losses reached approximately $\$38.95$ billion in 2023, marking a 12% increase compared to 2021 (CFCA, Retrieved August 08, 2024). Furthermore, data from national prosecution agencies indicate that from January to October 2023, over $34,000$ individuals were prosecuted for telecommunications network fraud crimes in China, reflecting a nearly 52% year-on-year increase. These statistics underscore the growing severity of telecommunications fraud and highlight the urgent need for more precise and adaptive detection methods.

With the rapid advancement of technology, electronic communication has become ubiquitous across various industries (Shawe-Taylor et al., 1999; Grabosky & Smith, 2003; Li & Wen, 2022). For instance, a delivery person or a salesperson may make numerous calls in a single day, often to different locations and clients (Anderson, 2022). This increase in electronic communication, while beneficial for business operations, also presents challenges in distinguishing between legitimate and fraudulent activities.

Traditional telecommunications fraud detection methods often focus too much on detailed data (Arafat et al., 2019; Becker et al., 2011), overlooking broader patterns and global information, as shown in Figure 1. Call-level methods primarily focus on analyzing call log data at the call level, often overlooking the relationships between different call records, such as changes in the caller's location (Abidogun, 2005; Tseng et al., 2015; Zhao et al., 2018). While some existing approaches have started to consider user-level analysis (Xu et al., 2008; Lopes et al., 2011; Li et al., 2018), they often rely on simplistic feature extraction techniques that

[1]Anonymous Institution, Anonymous City, Anonymous Region, Anonymous Country. Correspondence to: Anonymous Author <anon.email@domain.com>.

Preliminary work. Under review by the International Conference on Machine Learning (ICML). Do not distribute.

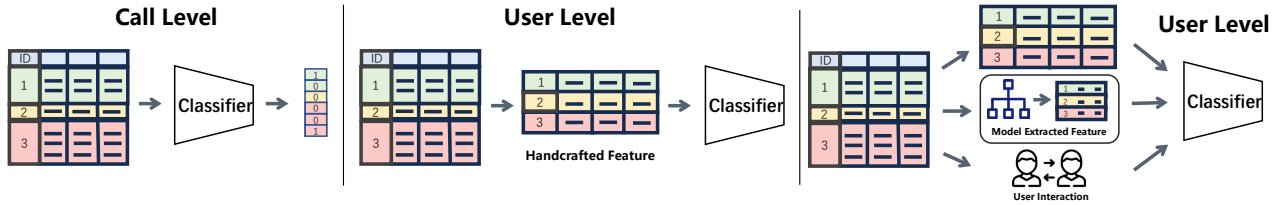

Figure 1. Traditional telecommunications fraud detection methods focus primarily on identifying abnormalities in call records or manually extracting user-level features to categorize users. In contrast, our method not only automatically extracts user features but also considers the correlations between users. One instance is represented by a short line. And the instances in the same color denote that they belong to one user.

fail to capture the complexity of user behavior. This oversight leads to a higher likelihood of misclassifying certain legitimate users, such as delivery personnel and salespeople, whose calling patterns inherently deviate from typical user behavior. Moreover, these methods often fail to account for the relationships between users, such as the frequency of calls between two users, which can indicate that they are likely acquaintances.

To address these challenges, we introduce a novel Multi-Level Feature Extraction (MLFE) method to better differentiate between low-risk and high-risk users. Our approach leverages advanced feature engineering techniques to capture the inner-user and inter-user characteristics, surpassing traditional methods. By employing a set function (Zaheer et al., 2017), we transform call records into detailed user features, offering a more accurate representation of activity patterns. Additionally, we enhance detection accuracy by modeling user relationships in a graph format (Nanavati et al., 2006), where nodes represent users and edges represent call interactions (Scarselli et al., 2008). This graph-based approach enables us to capture complex relationships and interactions, leading to a more precise identification of high-risk individuals through a deeper understanding of user behavior patterns.

We evaluate the effectiveness of our MLFE method using a real-world telecommunications dataset provided by a telecom network operator in China, demonstrating its high efficacy in accurately identifying fraudulent activities. Our findings reveal several patterns among high-risk users. Temporally, high-risk users are most active between 10 AM and 4 PM, suggesting a preference for fraudulent activities during typical working hours. Stability-wise, these users often have high incoming and outgoing call volumes, indicating that frequent call reception can also signal high-risk behavior. Spatially, high-risk users tend to have a broader geographical distribution of calls, reflecting a more dispersed operational range. In terms of call targets, high-risk users exhibit diversity both in the types of target users and on an individual level. Overall, our contributions are as follows:

- We are the first to introduce a multi-dimensional approach to user feature extraction for telecommunications fraud detection. Unlike traditional methods that rely solely on call logs or manually crafted features, our approach incorporates multiple perspectives, significantly improving both accuracy and robustness.
- Our MLFE method provides a comprehensive set of features, both inter and intra users, offering a richer and more detailed characterization of user behavior and interactions. This enables a more effective capture of the complexity of fraudulent activities compared to existing methods.
- We validated our approach using a real-world telecommunications dataset, demonstrating that the insights gained significantly enhance fraud detection for telecom companies. By refining the detection process and better understanding fraudster behavior, service providers are better equipped to protect networks and users, ultimately contributing to a more secure telecom environment.

2. Related Works

Telecommunications fraud detection has been extensively studied using various machine learning and graph-based approaches. Traditional methods primarily rely on analyzing call records or manually extracting user features, often overlooking complex user relationships and behaviors.

2.1. Traditional Machine Learning Methods

Machine learning techniques have been widely used to detect fraudulent activities by identifying patterns in call data (Awoyemi et al., 2017; Perols, 2011; Varmedja et al., 2019). For instance, Ezawa & Schuermann (1996) applied Bayesian network models to detect telecommunications fraud, effectively handling mixed data structures and rare event outcomes. Other studies have explored decision trees, support vector machines, and deep learning models to improve fraud detection accuracy (Ahmed & Mahmoudi, 2016). However, these approaches often rely on handcrafted features, limiting their ability to generalize across diverse fraud scenarios.

2.2. Graph Neural Networks for Fraud Detection

Graph Neural Networks (Scarselli et al., 2008; Villaizán-Vallelado et al., 2023) (GNNs) have been employed to enhance fraud detection capabilities in complex scenarios.

Dou et al. (2020) introduced the CARE-GNN model, which addresses camouflage behaviors of fraudsters by refining the GNN aggregation process with specialized modules. This approach has demonstrated significant improvements in detecting fraudulent activities by effectively capturing intricate user interactions. Other studies have leveraged attention mechanisms and message-passing strategies to further improve detection accuracy (Zhang et al., 2021).

2.3. Hybrid Approaches for Fraud Detection

Given the limitations of single-method approaches, recent studies have explored hybrid models that combine machine learning, graph-based techniques, and deep learning. Such methods leverage the strengths of each technique to improve fraud detection performance. For example, some works first extract features using traditional machine learning models and then use graph-based learning to refine fraud detection (Liu et al., 2022). Others integrate recurrent neural networks (RNNs) or transformers to capture temporal fraud patterns while preserving relational information in a graph structure (Chen et al., 2023). These hybrid approaches highlight the need for multi-perspective analysis in combating telecommunications fraud.

Traditional telecommunications fraud detection methods rely heavily on manually crafted features, often misclassifying users with similar calling patterns. Graph-based approaches improve detection by capturing user relationships but face scalability challenges and sensitivity to noisy data. Hybrid models attempt to integrate both techniques but still depend on predefined features, limiting adaptability. To address these issues, our MLFE method automatically extracts intra-user features from multiple call records using a set function and leverages GNNs to model inter-user relationships. This approach enhances fraud detection by effectively distinguishing legitimate users from fraudsters, improving accuracy and generalization across datasets.

3. Problem Statement

In this section, we will first introduce some preliminaries of telecommunications fraud detection. Then we will introduce the Tele dataset, along with some analyses.

3.1. Telecommunications Fraud Detection

Formally, we define the users in telecommunications fraud detection as $\{u_i\}_{i=1}^{N}$, where each u_i represents an individual user. The label space for our classification task is $y_i \in [Y] = \{0, 1\}$, where $y_i = 0$ denotes a low-risk user and $y_i = 1$ denotes a high-risk user.

In our telecommunications fraud detection task, the features of the users u cannot be directly obtained. Instead, we have access only to the call records $c \in \mathbb{R}^d$, which are represented as a d-dimensional vector. Each dimension corresponds to a specific feature of the call records, such as call duration, caller, receiver, and other relevant attributes.

For each user i, there are multiple call records, denoted as $\{c_i^n\}_{n=1}^{cn_i}$, where cn_i denotes the number of records related to the i-th user, which is variable and not fixed. This variability presents a unique challenge in our fraud detection task, as each user may have a different amount of data available for analysis. The irregularity in the number of call records necessitates robust data processing techniques to effectively handle and analyze the information. By considering the entire set of call records for each user, we can extract meaningful patterns and features that contribute to the detection of fraudulent behavior.

Therefore, our goal is to extract user features u from the call records c and minimize the empirical risk of the model f on the training set. Mathematically, this can be formulated as:

$$\min_{\theta} \frac{1}{N} \sum_{i=1}^{N} \mathcal{L}(f(g(c_i); \theta), y_i), \tag{1}$$

where g is the user feature extractor mapping call records c to user features u, θ represents the model parameters, f is the prediction function, and $\mathcal{L}$ is the loss function measuring the discrepancy between the predicted output and the actual label y_i. During the test phase, we evaluate the performance of g and f based on their ability to predict labels given any user's $\{c_j^n\}_{n=1}^{cn_j}$.

3.2. Tele Dataset

The Tele dataset is a comprehensive collection of call records, consisting of 701,815 records from 3,836 users. On average, each user has approximately 183 call records. The dataset also includes labels indicating whether users are classified as high-risk, making it well-suited for multi-instance learning.

Each user in the dataset is labeled as either low-risk or high-risk, with 79.8% (3,064 users) labeled as class-0 (low-risk) and 20.2% (772 users) labeled as class-1 (high-risk). The distribution of the number of call records per user is shown in Figure 5. In this dataset, user features are not directly accessible; we can only extract them through call records. Additionally, the relationships between users are not explicitly shown but can be inferred from the calls' party information. A more detailed description of the dataset can be found in the appendix.

4. Multi-Level Feature Extraction

Building on the challenges identified earlier, this section presents our Multi-Level Feature Extraction (MLFE) method. We extract intra-user and inter-user features us-

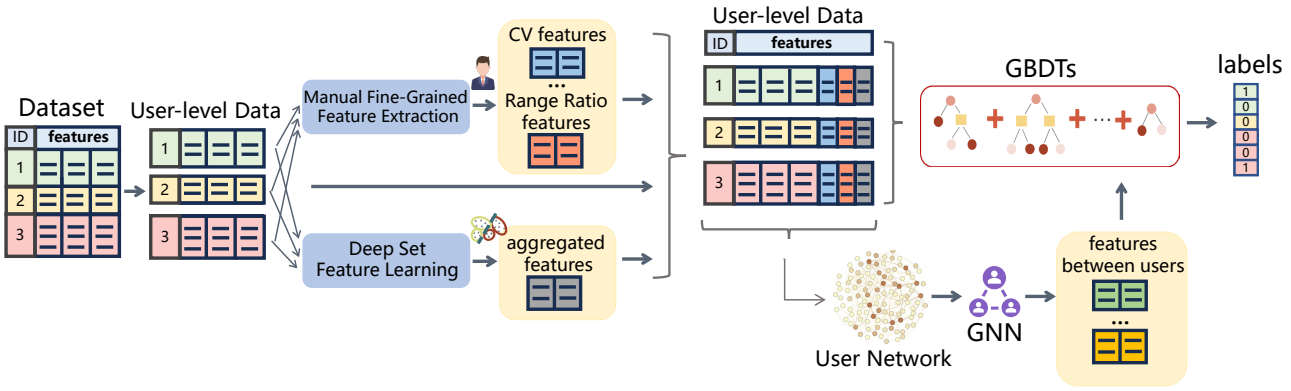

Figure 2. Flowchart of our MLFE method. Specifically, we use Manual Fine-Grained Feature Extraction and DeepSets Feature Learning to mine features from user-level data, then construct a user network and utilize GNN to extract features within the user network, followed by employing GBDTs to predict user labels based on these features.

ing a set function and a graph neural network, respectively, to provide multi-level features for user categorization in telecommunications fraud detection. Our flowchart is shown in Figure 2.

4.1. Deep Intra-User Feature Engineering

Our approach to Deep Intra-User Feature Engineering involves two key components: manual fine-grained feature extraction and automated feature deep learning.

4.1.1. Manual Fine-Grained Feature Extraction

In the manual feature extraction phase, we derive a comprehensive set of statistical metrics from the available data, focusing on temporal and spatial attributes. This includes calculating basic aggregations such as the minimum, maximum, mean, and weight of evidence (WOE) (Edwards, 1954; Smith et al., 2002) for various features like call time and location. These metrics capture the essential characteristics of each user's behavior, providing a foundational understanding of user-specific insights.

We start by extracting basic features, such as counts, maximum, minimum, and mean values, for temporal, spatial, and categorical attributes. These simple statistics provide an initial understanding of user behavior, highlighting key patterns and trends within the data.

To further enhance our analysis, and inspired by (Kirkland et al., 2007), we use Weight of Evidence (WOE) to assess the predictive power of categorical variables in distinguishing between low-risk and high-risk users in telecommunications fraud detection. WOE is calculated by comparing the distribution of low-risk and high-risk outcomes for a given feature, defined as:

$$\text{WOE} = \ln\left(\frac{\text{Proportion of Low-Risk Outcomes}}{\text{Proportion of High-Risk Outcomes}}\right) \quad (2)$$

where the proportion of low-risk outcomes refers to the fraction of users identified as low-risk, while the proportion of high-risk outcomes represents the fraction identified as high-risk. The advantage of using WOE is that it transforms categorical variables into continuous ones, making them more suitable for classification models. Additionally, WOE helps identify significant variables, reduces multicollinearity, and effectively handles missing values, thereby enhancing the interpretability and robustness of the model.

In our method, we aggregate multiple call records for each user based on initial features to derive advanced statistical metrics. These features include the variation coefficient (Abdi, 2010), skewness (Pearson, 1894), kurtosis (DeCarlo, 1997), variance-to-standard-deviation ratio (Dixon & Massey Jr, 1951), and range ratio (Sachs, 2012). These metrics provide deeper insights into user behavior patterns and improve the model's ability to identify high-risk users:

- Variation Coefficient (CV): Measures the relative variability of user behavior compared to the average, helping to identify users with erratic patterns that may indicate fraudulent activity.
- Skewness: Measures the relative variability of user behavior compared to the average, helping to identify users with erratic patterns that may indicate fraudulent activity.
- Kurtosis: Evaluates the "tailedness" of the data distribution, indicating how extreme user behaviors are, which can reveal users with unusually high or low activity levels.
- Variance-to-Standard-Deviation Ratio (Variance/Std Ratio): Indicates the spread of data relative to its standard deviation, identifying users with high variability that might signal irregular activities.
- Range Ratio: Compares the range of user activities to the average activity level, identifying users with broad or narrow activity ranges and providing insights into behavioral diversity.

4.1.2. Automated Feature Learning

Given that manually extracted features often carry inherent biases and may not capture the full extent of user behavior, we implement an automated user feature extraction method. Inspired by the DeepSets (Zaheer et al., 2017; Zhang et al., 2019) approach, our goal is not to classify individual call records but to classify the entire set of call records for each user.

The DeepSets method is particularly well-suited for this scenario. It consists of two main components: the transformation function ϕ and the aggregation function ψ. These are defined as follows:

$$u_i = \psi\left(\sum_{j=1}^{cn_i} \phi(c_j)\right), \tag{3}$$

where ϕ is applied to each call record c_j, transforming it into a certain feature space where individual records contribute meaningful information. This transformation captures the essential characteristics of each call record. The function ψ, which gets the same results regardless of the permutation of the instances, then aggregates the transformed features across all call records, effectively summarizing the user's behavior as a whole. This aggregation ensures that the model considers the collective behavior of the user's call records, capturing patterns that may not be evident from individual records.

4.2. Inter-User Feature Extraction

In addition to the individual user features, we also account for the interactions between users. Call data reflects the connections among different users and can be modeled as a graph $G = (V, E)$, where V represents the set of nodes and E represents the set of edges.

Each node $v_i \in V$ represents a user in the network. The initial features of the nodes may include both the statistical information of the user and those manually extracted from the call records, such as call frequency, average call duration, and user demographics. An edge e_{ij} connects node v_i to node v_j, representing a call made from user u_i to user u_j or the opposite. The edge can be weighted based on various features derived from the call records, such as time, location, and roaming information.

With a Graph Neural Network (GNN) (Scarselli et al., 2008), we can capture the propagation mechanisms along the edges, allowing us to continuously update node features from user information to interactive information. After training, the model learns the relationships between the network topology and telecommunications fraud, thereby incorporating inter-user information into the node features. In simple terms, the purpose of using a GNN is to learn a function h that maps each node to a feature vector, capturing both local and global patterns:

$$h(v_i) = \mathrm{GNN}(v_i, \{v_j \in \mathcal{N}(v_i)\}) \tag{4}$$

where $v_j \in \mathcal{N}$ represents the neighbors of node v_i.

By constructing the graph in this way, telecommunications fraud detection becomes a node classification problem, where the task is to classify each node (user) as either low-risk or high-risk based on its features and its connections with other nodes.

However, relying solely on GNNs for user classification may overlook some nuanced information embedded in individual user attributes. To address this, we use the features extracted by the GNN as additional input for traditional tree models in the classification process. Traditional tree models, such as Gradient Boosting Decision Trees (GBDTs) (Feng et al., 2018), are well-known for their strong performance in capturing complex patterns and interactions within data (Grinsztajn et al., 2022; McElfresh et al., 2024). By combining the strengths of both GNNs and traditional tree models, we can leverage their complementary nature to improve classification accuracy. While GNNs capture the intra-user relationships, GBDTs provide a robust framework for learning from individual features, ultimately resulting in a more comprehensive and accurate classification model.

To summarize, our method not only manually extracts fine-grained user features but also automatically learns user features using set functions. Additionally, it captures inter-user relationships through GNNs. By combining all of these elements, our approach achieves accurate user feature extraction, which will be validated in the following experiments.

The MLFE framework is designed with scalability in mind. By leveraging automated learning components such as DeepSets and GNNs, it avoids the limitations of traditional manual feature engineering and supports efficient processing of large-scale datasets. The modular structure allows for seamless integration with various downstream models. In large-volume scenarios, such as nationwide telecom fraud detection, this design ensures both computational feasibility and representational richness.

5. Experiments

In this section, we present a comparative analysis of our proposed method, MLFE, against traditional tree-based methods, deep tabular learning methods, set methods, and graph neural networks using the Tele dataset. Through a series of experimental evaluations and ablation studies, we demonstrate the effectiveness of our approach. Additionally, we analyze the results of our method to draw meaningful conclusions for telecommunications fraud detection.

5.1. Experiments and Results

5.1.1. Baselines and metrics

Our baseline methods are divided into four categories: traditional tree-based methods, deep tabular learning methods, set methods, and graph neural networks. The metrics used to evaluate are Precision, Recall, F1 Score, Accuracy (ACC), and Area Under the Receiver Operating Characteristic Curve (AUC). More details can be found in Appendix B.

5.1.2. Results

To demonstrate the superiority of MLFE, we compare it with other popular methods in the tele dataset as shown in Table 1. Our experimental results demonstrate that our method outperforms all other methods in this dataset. For clarity, we present a concise version of Table 1 here, while the complete version is provided in the Appendix.

In addition to the experiments on the tele dataset, we further validated the generalization ability of our MLFE method on a new dataset provided by the Digital Sichuan Innovation Competition. This competition dataset focuses on telecommunications fraud detection and contains rich intra-user and inter-user feature representations, making it an excellent choice for evaluating our method's robustness across diverse scenarios. Since this dataset lacks significant graph structure features, graph-based methods were not included in the evaluation. The results are presented in Table 2. Similarly, we provide the full version of Table 2 in the Appendix.

The traditional tree-based models (XGB (Chen & Guestrin, 2016), CAB (Prokhorenkova et al., 2018), and LGB (Ke et al., 2017)) demonstrate strong performance across most metrics, indicating their robustness in handling complex datasets. The deep tabular learning methods (MLP, ResNet (Gorishniy et al., 2021), and TabR (Gorishniy et al., 2023)) also perform well, particularly in identifying positive cases, as evidenced by their higher Recall scores. However, the DeepSets (Zaheer et al., 2017) and Graph Neural Network (GNN) (Chiang et al., 2019) methods exhibit lower performance. This suggests that focusing solely on intra-user features or inter-user features is insufficient for accurately detecting fraud.

To address these limitations, we propose our MLFE method that integrates both intra-user and inter-user features. By using three traditional tree-based models as backbones, our MLFE method significantly enhances the performance of these methods. In Table 1, it is evident that the MLFE-enhanced models (MLFE-XGB, MLFE-CAB, and MLFE-LGB) show substantial improvements across all metrics, demonstrating the effectiveness of combining user-specific and user-interaction features. Notably, MLFE-LGB achieves the hig-hest overall performance, confirming the superiority of MLFE features in enhancing model accuracy and robustness. These results validate the efficacy of the MLFE approach in improving telecommunications fraud detection.

Table 1. Performance comparison of MLFE and existing methods in Tele Dataset.

Model	Precision	Recall	F1
XGB	0.8955	0.8224	0.7613
CAB	0.8984	0.8186	0.7578
LGB	0.8842	0.8254	0.7597
MLP	0.8851	0.8229	0.7571
ResNet	0.8880	0.8354	0.7722
TabR	0.9056	0.8336	0.7792
DeepSets	0.6111	0.6188	0.6149
GNN	0.8911	0.5960	0.7143
MLFE-XGB	<u>0.9201</u>	<u>0.8679</u>	<u>0.8196</u>
MLFE-CAB	0.9163	0.8615	0.8166
MLFE-LGB	**0.9215**	**0.8699**	**0.8286**

Table 2. Performance comparison of MLFE and existing methods in the Digital Sichuan dataset.

Model	Precision	Recall	F1
XGB	0.8490	0.8359	0.8424
CAB	0.8450	0.8385	0.8417
LGB	0.8575	0.8333	0.8453
MLP	0.8705	0.8103	0.8393
ResNet	0.8325	0.8154	0.8238
TabR	0.8863	0.8258	0.7651
MLFE-XGB	0.8909	<u>0.8954</u>	<u>0.8931</u>
MLFE-CAB	<u>0.9175</u>	**0.9082**	**0.9128**
MLFE-LGB	**0.9632**	0.8673	**0.9128**

Overall, the experimental results give strong support that the combination of deep intra- and inter-user features with tree-based models significantly boosts model performance, offering a more nuanced understanding of user behavior and improving the detection of high-risk users in telecommunications fraud scenarios.

5.2. Experiments Analysis

5.2.1. Ablation Studies

The ablation study presented in Table 3 evaluates the impact of different feature combinations across both traditional and modern tabular classification models. We include three tree-based models (XGB, CAB, LGB) and four state-of-the-art neural architectures (MLP-PLR, RealMLP, TabM, TabPFN). (The full version of Table 3 is in the appendix.)

The models are assessed under four feature settings: - Vanilla: Utilizes only the original features without any additional processing. - +Inter: Incorporates inter-user features

extracted via Graph Neural Networks, capturing the relationships between users. - +Intra: Incorporates intra-user features extracted by DeepSets, focusing on individual user characteristics. - MLFE: Our proposed method, which combines both inter-user and intra-user features to leverage the strengths of both perspectives.

Table 3. Ablation study on our MLFE method.

Model	Feature	Precision	Recall	F1
XGB	Vanilla	0.896	0.822	0.761
	+ Inter	0.911	0.858	0.810
	+ Intra	0.909	0.865	0.817
	MLFE	**0.920**	**0.868**	**0.820**
CAB	Vanilla	0.898	0.819	0.758
	+ Inter	0.906	**0.869**	**0.819**
	+ Intra	0.911	0.866	0.818
	MLFE	**0.916**	0.861	0.817
LGB	Vanilla	0.884	0.825	0.760
	+ Inter	0.906	0.868	0.818
	+ Intra	0.912	0.868	0.822
	MLFE	**0.921**	**0.870**	**0.829**
MLP-PLR	Vanilla	0.871	0.845	0.769
	+ Inter	0.865	0.856	0.776
	+ Intra	0.889	0.874	0.809
	MLFE	**0.894**	**0.878**	**0.816**
RealMLP	Vanilla	0.862	0.830	0.748
	+ Inter	0.864	0.846	0.765
	+ Intra	0.892	0.874	0.809
	MLFE	**0.896**	**0.879**	**0.818**
TabM	Vanilla	0.879	0.834	0.762
	+ Inter	0.869	0.848	0.771
	+ Intra	**0.897**	**0.879**	**0.819**
	MLFE	0.892	0.872	0.809
TabPFN	Vanilla	0.874	0.823	0.746
	+ Inter	0.885	0.842	0.774
	+ Intra	**0.898**	**0.890**	**0.830**
	MLFE	0.896	0.883	0.823

The results demonstrate that incorporating either inter- or intra-user features individually significantly improves the performance of all the models compared to original features. Specifically, in the case of tree-based methods, adding inter-user features yields notable improvements in Recall and F1 scores, highlighting the importance of capturing user relationships. Similarly, adding intra-user features enhances Precision and AUC, emphasizing the value of detailed user characteristics.

For neural models, the impact of additional features is more nuanced but still clearly beneficial. In particular, intra-user features often lead to the largest gains in F1, achieving the best results in models like TabM and TabPFN. Inter-user features, while slightly less impactful, still yield noticeable improvements across all neural architectures.Although MLFE does not always outperform the best individual feature variant, it delivers consistently strong and competitive results across all neural models.

The superior performance of MLFE can be attributed to its ability to provide a more comprehensive view of user behavior by integrating both intra-user and inter-user features. This integration allows the model to capture complex patterns and dependencies that would otherwise be missed when using only a single type of feature.

Importantly, the MLFE framework is model-agnostic: the extracted features can be seamlessly integrated into a wide range of downstream classifiers, from tree-based ensembles to deep neural architectures. This highlights not only the effectiveness but also the flexibility and robustness of our approach for real-world fraud detection scenarios.

5.2.2. Visualization

The T-SNE (Van der Maaten & Hinton, 2008) visualizations of user features extracted by deepsets and graphs are shown in Figure 3. The plot reveals that high-risk and low-risk users are intricately interwoven throughout the space, indicating a complex relationship between user interactions and risk levels.

From the visualization, it is clear that in the graph feature space, high-risk users do not form distinct, separate clusters but are instead dispersed among low-risk users. This distribution suggests that high-risk behavior is not solely determined by isolated characteristics but is influenced by the network of interactions and relationships with other users. In contrast, in the set feature space, users can be roughly distinguished, with only a few challenging samples. This indicates that deep sets are more effective at extracting the inherent characteristics of different users.

The overlap between high-risk and low-risk users highlights the importance of considering inter-user relationships in fraud detection. By capturing these relationships, Graph Neural Networks (GNNs) can extract features that reveal subtle patterns and dependencies that may not be apparent when examining users in isolation. This underscores the need to leverage both intra-user and inter-user features to enhance the accuracy of risk classification.

5.2.3. User Patterns Analysis

We have already discussed in section A.1.3 some temporal differences between high-risk and low-risk users. Specifically, high-risk users typically conduct their activities between 10 AM and 4 PM. Here, we will conduct a more detailed analysis based on feature importance to uncover finer distinctions between these user groups. In the process of conducting a deep analysis of user characteristics, we

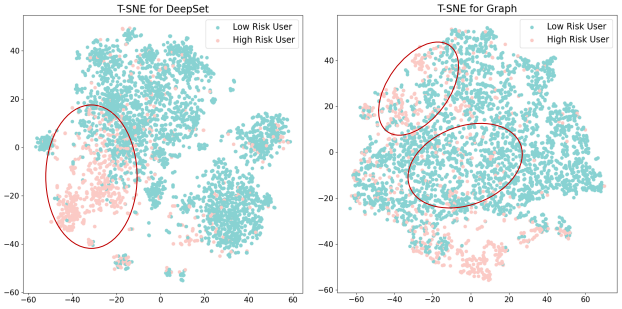

Figure 3. T-SNE visualization for features extracted by deepsets and graph.

first visualized the distribution of features based on their importance, as shown in Figure 4.

Call Event Stability. First, regarding stability, we can examine the top two charts in Figure 4. The left chart displays the distribution of variance indicators for call events (whether the user is the caller or the callee). High-risk users have significantly lower variance in this measure, indicating that they seldom switch between making and receiving calls. The right chart presents the distribution of making-call ratios in call events. This distribution shows that users who make an exceptionally high number of calls tend to be high-risk. Interestingly, users who frequently receive calls and rarely make them are also likely to be high-risk. Conversely, users who have a balanced pattern of making and receiving calls are often low-risk. This analysis highlights the importance for telecom operators to monitor users who make or receive a large volume of calls, as they might be involved in fraudulent activities.

Spatiality. Secondly, concerning target distribution, the middle two images in Figure 4 visualize the distribution of key target distribution features. These images compare the variance_std_ratio of the WOE (Weight of Evidence) for two characteristics: long_type1(the type of the long distance call) and called_home_code (the area code of the called user's address) for high-risk and low-risk users. It was observed that high-risk users make significantly more long-distance calls than low-risk users, and low-risk users rarely make long-distance calls. Meanwhile, high-risk users have higher variance in the called_home_code variance feature, indicating a wider geographical range and variety of call recipients. In contrast, roles like salespeople or delivery personnel often make numerous calls within a single region. This differentiation helps distinguish high-risk users from individuals in professions such as delivery or sales, where making and receiving many calls is common.

Target Distribution. Finally, concerning target distribution, we examined user stability features based on their importance, as shown in the bottom two charts of Figure 4. The left image shows that high-risk users exhibit either high

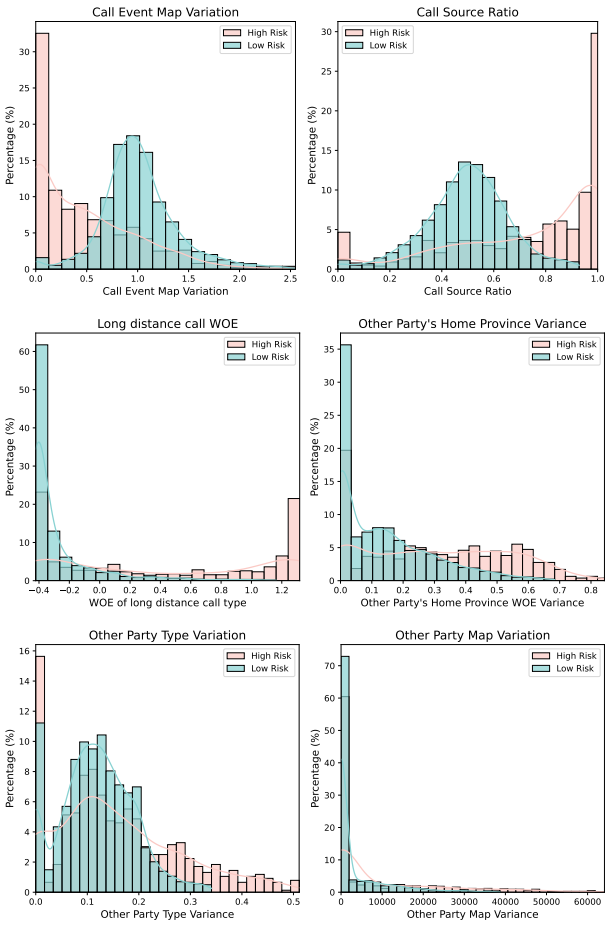

Figure 4. Comparison of call proportions between low-risk and high-risk users by hour and day of the week.

variance or zero variance in the types of called parties, suggesting that some high-risk users frequently deceive specific types of people, while others target multiple groups to expand their scope of fraud. In contrast, low-risk users typically call a more moderate number of target types. The right chart shows the variance in the number of target users called. High-risk users tend to call a larger number of people to commit fraud, while low-risk users generally do not engage in such behavior.

To sum up, our research findings reveal key characteristics of high-risk users from four aspects. Temporally, high-risk users are most active between 10 AM and 4 PM, suggesting that they tend to conducting fraud during working hours. Stability-wise, these users often have high volumes of incoming or outgoing calls, suggesting that frequent answering of calls could also indicate risk. Spatially, they contact a wide geographical range of targets, showing a dispersed activity area. Regarding call targets, most high-risk users exhibit diversity in both the types of target users and on an individual level, however, some high-risk users focus

exclusively on a specific type of user.

6. Conclusion

In this paper, we propose the Multi-Level Feature Extraction (MLFE) method for telecommunications fraud detection that captures multi-dimensional user features. Our approach considers both fine-grained intra-user characteristics and inter-user relationships, offering a comprehensive view of user behavior. We validated the superiority of our method through experiments on the real-world Tele dataset provided by telecom operators. The results demonstrate significant improvements over existing approaches. Additionally, we conducted an in-depth analysis of the features extracted by the MLFE, uncovering insightful conclusions that provide practical anti-fraud recommendations for telecom companies and individual users alike.

Accessibility

Due to privacy concerns regarding the tele dataset, we have only made the code related to the SiChuan dataset publicly available. The full implementation of our method is accessible in an anonymous GitHub repository: https://anonymous.4open.science/r/tele-7D6B/

Impact Statement

This paper presents work whose goal is to advance the field of Machine Learning. There are some potential social implications in our work. We believe researchers should maintain a certain level of privacy regarding the open-source data and code, restricting their use to research purposes only. At the same time, our method has been provided to telecommunications companies and has proven to hold significant value by the experts.

References

Abdi, H. Coefficient of variation. *Encyclopedia of research design*, 1(5):169–171, 2010.

Abidogun, O. A. *Data mining, fraud detection and mobile telecommunications: call pattern analysis with unsupervised neural networks*. PhD thesis, University of the Western Cape, 2005.

Ahmed, M. B. and Mahmoudi, S. Survey on credit card fraud detection using machine learning techniques. *International Journal of Computer Applications*, 172(7): 22–25, 2016.

Anderson, K. B. Mass-market consumer frauds:: What the statistical data show. In *A Fresh Look at Fraud*, pp. 15–41. Routledge, 2022.

Arafat, M., Qusef, A., and Sammour, G. Detection of wangiri telecommunication fraud using ensemble learning. In *2019 IEEE Jordan International Joint Conference on Electrical Engineering and Information Technology (JEEIT)*, pp. 330–335. IEEE, 2019.

Awoyemi, J. O., Adetunmbi, A. O., and Oluwadare, S. A. Credit card fraud detection using machine learning techniques: A comparative analysis. In *2017 international conference on computing networking and informatics (ICCNI)*, 2017.

Becker, R. A., Volinsky, C., and Wilks, A. R. Fraud detection in telecommunications: History and lessons learned. *Technometrics*, 52(1):20–33, 2010.

Becker, R. A., Volinsky, C., and Wilks, A. R. Fraud detection in telecommunications: History and lessons learned. *Quality control and applied statistics*, 56(1):143–144, 2011.

CFCA. Global fraud loss survey 2023. *New Jersey: Communications Fraud Control Association.*, Retrieved August 08, 2024.

Chadyšas, V., Bugajev, A., Kriauzienė, R., and Vasilecas, O. Outlier analysis for telecom fraud detection. In *International Baltic Conference on Digital Business and Intelligent Systems*, pp. 219–231. Springer, 2022.

Chen, T. and Guestrin, C. Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 785–794, 2016.

Chen, Y., Xu, M., Wang, W., Fang, S., and Guo, R. Fraudtrans: Transformer-based hybrid approach for financial fraud detection. *IEEE Transactions on Neural Networks and Learning Systems*, 2023.

Chiang, W.-L., Liu, X., Si, S., Li, Y., Bengio, S., and Hsieh, C.-J. Cluster-gcn: An efficient algorithm for training deep and large graph convolutional networks. In *Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining*, pp. 257–266, 2019.

DeCarlo, L. T. On the meaning and use of kurtosis. *Psychological methods*, 2(3):292, 1997.

Dixon, W. J. and Massey Jr, F. J. Introduction to statistical analysis. 1951.

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

Edwards, W. The theory of decision making. *Psychological bulletin*, 51(4):380, 1954.

Ezawa, K. J. and Schuermann, T. Fraud/uncollectible debt detection using a bayesian network based learning system: A rare binary outcome with mixed data structures. *International Journal of Intelligent Systems in Accounting, Finance & Management*, 7(4):233–247, 1996.

Feng, J., Yu, Y., and Zhou, Z.-H. Multi-layered gradient boosting decision trees. *Advances in neural information processing systems*, 31, 2018.

Gorishniy, Y., Rubachev, I., Khrulkov, V., and Babenko, A. Revisiting deep learning models for tabular data. *Advances in Neural Information Processing Systems*, 34: 18932–18943, 2021.

Gorishniy, Y., Rubachev, I., Kartashev, N., Shlenskii, D., Kotelnikov, A., and Babenko, A. Tabr: Unlocking the power of retrieval-augmented tabular deep learning. *arXiv preprint arXiv:2307.14338*, 2023.

Grabosky, P. and Smith, R. Telecommunication fraud in the digital age: The convergence of technologies. In *Crime and the Internet*, pp. 41–55. Routledge, 2003.

Grinsztajn, L., Oyallon, E., and Varoquaux, G. Why do tree-based models still outperform deep learning on typical tabular data? In *Advances in Neural Information Processing Systems 35 (NeurIPS)*, 2022.

Ke, G., Meng, Q., Finley, T., Wang, T., Chen, W., Ma, W., Ye, Q., and Liu, T.-Y. Lightgbm: A highly efficient gradient boosting decision tree. *Advances in Neural Information Processing Systems*, 30, 2017.

Kirkland, D., Aardema, M., Banduhn, N., Carmichael, P., Fautz, R., Meunier, J., and Pfuhler, S. In vitro approaches to develop weight of evidence (woe) and mode of action (moa) discussions with positive in vitro genotoxicity results. *Mutagenesis*, 22(3):161–175, 2007.

Li, G. and Wen, Y. [retracted] research on the detection countermeasures of telecommunication network fraud based on big data for killing pigs and plates. *Journal of Robotics*, 2022(1):4761230, 2022.

Li, R., Zhang, Y., Tuo, Y., and Chang, P. A novel method for detecting telecom fraud user. In *2018 3rd International Conference on Information Systems Engineering (ICISE)*, pp. 46–50. IEEE, 2018.

Liu, Y., Wang, S., Jiang, X., and Tang, X. Multi-view graph learning for financial fraud detection. *Expert Systems with Applications*, 198:116809, 2022.

Lopes, J., Belo, O., and Vieira, C. Applying user signatures on fraud detection in telecommunications networks. In *Advances in Data Mining. Applications and Theoretical Aspects: 11th Industrial Conference, ICDM 2011, New York, NY, USA, August 30–September 3, 2011. Proceedings 11*, pp. 286–299. Springer, 2011.

Loshchilov, I. and Hutter, F. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017.

Lu, C., Lin, S., Liu, X., and Shi, H. Telecom fraud identification based on adasyn and random forest. In *2020 5th International Conference on Computer and Communication Systems (ICCCS)*, pp. 447–452. IEEE, 2020.

McElfresh, D., Khandagale, S., Valverde, J., Prasad C, V., Ramakrishnan, G., Goldblum, M., and White, C. When do neural nets outperform boosted trees on tabular data? *Advances in Neural Information Processing Systems (NeurIPS)*, 2024.

Nanavati, A. A., Gurumurthy, S., Das, G., Chakraborty, D., Dasgupta, K., Mukherjea, S., and Joshi, A. On the structural properties of massive telecom call graphs: findings and implications. In *Proceedings of the 15th ACM international conference on Information and knowledge management*, pp. 435–444, 2006.

Ni, P. and Yu, W. A victim-based framework for telecom fraud analysis: A bayesian network model. *Computational Intelligence and Neuroscience*, 2022(1):7937355, 2022.

Pearson, K. Contributions to the mathematical theory of evolution. *Philosophical Transactions of the Royal Society of London. A*, 185:71–110, 1894.

Perols, J. Financial statement fraud detection: An analysis of statistical and machine learning algorithms. *Auditing: A Journal of Practice & Theory*, 30(2):19–50, 2011.

Prokhorenkova, L., Gusev, G., Vorobev, A., Dorogush, A. V., and Gulin, A. Catboost: unbiased boosting with categorical features. *Advances in Neural Information Processing Systems*, 31, 2018.

Rosset, S., Murad, U., Neumann, E., Idan, Y., and Pinkas, G. Discovery of fraud rules for telecommunications—challenges and solutions. In *Proceedings of the fifth ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 409–413, 1999.

Sachs, L. *Applied statistics: a handbook of techniques*. Springer Science & Business Media, 2012.

Scarselli, F., Gori, M., Tsoi, A. C., Hagenbuchner, M., and Monfardini, G. The graph neural network model. *IEEE transactions on neural networks*, 20(1):61–80, 2008.

Shawe-Taylor, J., Howker, K., and Burge, P. Detection of fraud in mobile telecommunications. *Information Security Technical Report*, 4(1):16–28, 1999.

Smith, E. P., Lipkovich, I., and Ye, K. Weight-of-evidence (woe): quantitative estimation of probability of impairment for individual and multiple lines of evidence. *Human and Ecological Risk Assessment*, 8(7):1585–1596, 2002.

Tseng, V. S., Ying, J.-C., Huang, C.-W., Kao, Y., and Chen, K.-T. Fraudetector: A graph-mining-based framework for fraudulent phone call detection. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 2157–2166, 2015.

Van der Maaten, L. and Hinton, G. Visualizing data using t-sne. *Journal of machine learning research*, 2008.

Varmedja, D., Karanovic, M., Sladojevic, S., Arsenovic, M., and Anderla, A. Credit card fraud detection-machine learning methods. In *2019 18th International Symposium INFOTEH-JAHORINA (INFOTEH)*, pp. 1–5. IEEE, 2019.

Villaizán-Vallelado, M., Salvatori, M., Martínez, B. C., and Sánchez-Esguevillas, A. J. Graph neural network contextual embedding for deep learning on tabular data. *CoRR*, abs/2303.06455, 2023.

Xu, W., Pang, Y., Ma, J., Wang, S.-Y., Hao, G., Zeng, S., and Qian, Y.-H. Fraud detection in telecommunication: a rough fuzzy set based approach. In *2008 International Conference on Machine Learning and Cybernetics*, volume 3, pp. 1249–1253. IEEE, 2008.

Zaheer, M., Kottur, S., Ravanbakhsh, S., Poczos, B., Salakhutdinov, R. R., and Smola, A. J. Deep sets. *Advances in neural information processing systems*, 30, 2017.

Zhang, J., He, Y., Luo, D., and Li, Q. Learning from dynamic user interaction graphs for identity fraud detection. *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pp. 634–644, 2021.

Zhang, Y., Hare, J., and Prugel-Bennett, A. Deep set prediction networks. *Advances in Neural Information Processing Systems*, 32, 2019.

Zhao, L., Song, Y., Zhang, C., Liu, Y., Wang, P., Lin, T., Deng, M., and Li, H. T-gcn: A temporal graph convolutional network for traffic prediction. *IEEE transactions on intelligent transportation systems*, 21(9):3848–3858, 2019.

Zhao, Q., Chen, K., Li, T., Yang, Y., and Wang, X. Detecting telecommunication fraud by understanding the contents of a call. *Cybersecurity*, 1:1–12, 2018.

A. Tele Dataset

The Tele dataset is a comprehensive collection of call records, containing a total of 701,815 call records from 3,836 users. On average, each user has approximately 183 call records. This dataset includes labels indicating whether the users are high-risk, making it suitable for multi-instance learning. The dataset is composed of three main files, `features.csv`, `labels.csv`, and `id_description.xlsx`, which are described in Table 4.

Table 4. Data structure for call records from the Tele dataset: (1) *features*, row of all 28 feature values for `msisdn`, (2) *Identification number description*, introducing the identification number types that appear frequently in features, and (3) *is_sa*, the class label for `msisdn`.

`features.csv`

msisdn	start_time	end_time	call_event	other_party	ismultimedia	···	date_c
1000176	20231223095118	20231223095129	call_dst	1069460	0	···	20240106
···	···	···	···	···	···	···	···
1000176	20231230122627	20231230122806	call_src	1196442	0	···	20240113
1000184	20240113202822	20240113203058	call_src	1242046	0	···	20240116
···	···	···	···	···	···	···	···

`id_description.xlsx`

Type	Explanation
0	Unknown
9	Virtual Operator (numbers starting with 170, 171, etc.)
···	···

`labels.csv`

msisdn	is_sa
1000176	0
1000184	1
···	···

Each user in the dataset is labeled as either low-risk or high-risk, with 79.8% (3,064 users) labeled as class-0 (low-risk) and 20.2% (772 users) labeled as class-1 (high-risk). The distribution of the number of call records per user is shown in Figure 5. The figure shows that the number of call records among users in the Tele dataset exhibits diversity. In this dataset, user features are not directly accessible; we can only extract them through call records. Additionally, the relationships between users are not explicitly shown but can be inferred from the calls' party information.

A.1. Dataset Analysis

In this section, we will further analyze and visualize the features of the dataset to motivate our method's introduction. By examining the characteristics of the data, we aim to uncover underlying patterns and relationships that justify the need for our method.

The dataset includes a variety of features that provide detailed information about each call record. These features are categorized into five main groups: User Information, Call Information, Party Information, Geographical Information, and Metadata. Table 5 provides a comprehensive description of these features.

To better understand the characteristics of the Tele dataset, we conducted a statistical analysis and visualized the distributions of key features.

A.1.1. From Call Records to Users

As shown in the Figure 5, the number of call records for each user is uncertain and follows a long-tailed distribution. Therefore, it is not reasonable to use manually designed user features directly for users with a small number of call records, as these features may not be representative. Additionally, when visualizing the distribution of some manually extracted features in Figure 6, such as statistics on call time and location, we find that, except for `dayofweek`, the differentiation of all features after normalization is particularly small.

Table 5. Tele Dataset Features Description

Feature	Description
User Information:	
msisdn	Unique identifier for the user (Integer)
a_product_id	Product code corresponding to different products (Integer)
open_datetime	Account opening time (YYYYMMDDHHMMSS)
Call Information:	
start_time	Timestamp indicating the start of the call (YYYYMMDDHHMMSS)
end_time	Timestamp indicating the end of the call (YYYYMMDDHHMMSS)
call_event	Type of call event, e.g., call_src: outgoing, call_dst: incoming (Categorical)
call_duration	Call duration in seconds (Integer)
hour	Call hour of the day (Integer)
dayofweek	Day of the week of the call (Integer)
ismultimedia	Flag indicating whether the call was a video call (Binary)
cfee	Basic call fee (Float)
lfee	Long distance call fee (Float)
Party Information:	
other_party	Unique identifier for the called party in the call (Integer)
a_serv_type	User service type, e.g., 01: outgoing, 02: incoming, 03: call forwarding (Categorical)
long_type1	Long distance type, e.g., international, inter-provincial, intra-provincial (Categorical)
roam_type	Roaming type, e.g., none, intra-provincial roaming, inter-provincial roaming, international roaming (Categorical)
Geographical Information:	
home_area_code	Area code of the caller's home location (Integer)
visit_area_code	Area code of the caller's location (Integer)
called_home_code	Area code of the called party's home location (Integer)
called_code	Area code of the called party's location (Integer)
phone1_type	Type of the caller's phone, e.g., landline, mobile (Categorical)
phone2_type	Type of the called phone (Categorical)
phone1_loc_city	City name of the caller's location (Categorical)
phone1_loc_prov	Province name of the caller's location (Categorical)
phone2_loc_city	City name of the called person's location (Categorical)
phone2_loc_prov	Province name of the called person's location (Categorical)
Metadata:	
update_time	The time when the call record was stored
date	Date of the call
date_c	Data acquisition time

A.1.2. Geographical Distribution of Calls

We analyzed the geographical distribution of call records. Using the city-level map of China, we created a heatmap to visualize the distribution of calls across different cities. Figure 7 shows the geographical distribution of call records, with darker colors indicating higher call volumes. The distribution of these call records indicates that our dataset is highly representative. At the same time, the differences between locations highlight the importance of focusing more on location-related features.

A.1.3. Risk Category Comparisons

To compare the behaviors of low-risk and high-risk users, we analyzed the call distribution by hour and day of the week. Figure 8 shows these comparisons using line plots. The plots show that high-risk users have a more oscillatory call frequency, meaning their call variance is greater than that of low-risk users, both within weeks and within days. Additionally, these users tend to make most of their calls between 10 AM and 4 PM, from Wednesday to Friday.

To conclude, the Tele dataset offers a rich set of features that enable detailed analysis and modeling. Its combination of call details and user labels makes it an invaluable resource for research in telecommunications and risk assessment. We should delve deeper into the relationships within each user's data, including temporal and spatial information, while also paying attention to the interaction between users.

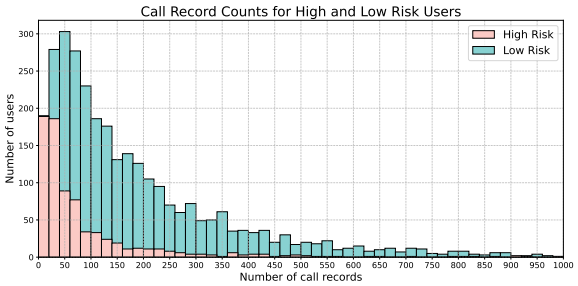

Figure 5. Distribution of the number of call records per user.

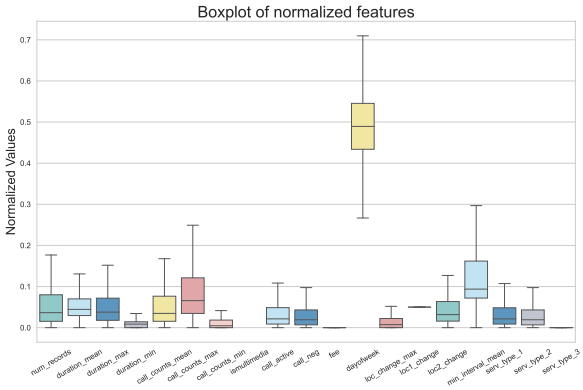

Figure 6. Distribution of the manually extracted user characteristics.

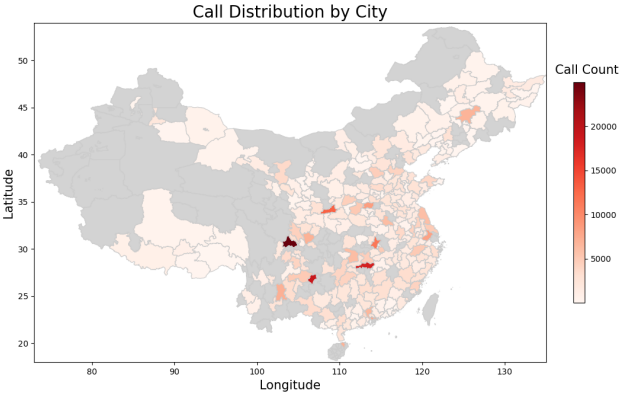

Figure 7. Geographical distribution of call records in the Tele dataset.

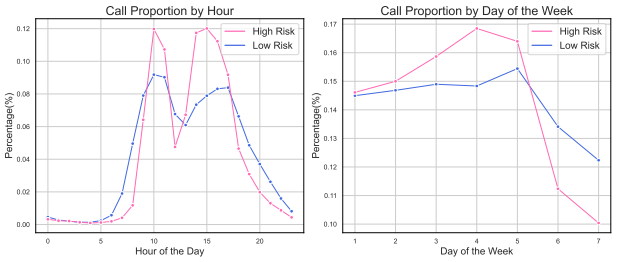

Figure 8. Comparison of call proportions between low-risk and high-risk users by hour and day of the week.

B. Baselines and Evaluation Metrics

For traditional tree-based methods, we compared XGBoost (Chen & Guestrin, 2016) (XGB), LightGBM (Ke et al., 2017) (LGB), and CatBoost (Prokhorenkova et al., 2018) (CAB). XGB is an ensemble method that uses boosting techniques to improve prediction accuracy by combining the predictions of several weak learners. LGB is a gradient-boosting framework designed for efficiency and scalability, while CAB is particularly effective with categorical features due to its innovative handling of categorical features. In the category of deep tabular learning methods, we included multilayer perceptron (MLP), ResNet (Gorishniy et al., 2021), and TabR (Gorishniy et al., 2023). MLP is an artificial neural network with multiple layers that maps input data to output labels using backpropagation. ResNet is a deep residual network that uses shortcut connections to facilitate the training of very deep networks. TabR is a recent architecture for tabular data, leveraging its specific neighbor information for improved performance. For set methods, we used DeepSets, a neural network architecture designed for sets, capable of handling unordered inputs with permutation-invariant operations. Lastly, we evaluated Graph Neural Networks, specifically GCNs (Zhao et al., 2019; Chiang et al., 2019), which are designed to work directly on graph-structured data, capturing dependencies and relationships between nodes through message passing. For every traditional tree-based model and deep tabular learning method, we search the hyperparameters for 50 iterations and select the hyperparameters that perform the best on the validation set. The hyperparameter space will be shown in the Appendix C. While for the deepsets methods and GNNs, we only train them for 200 epochs as we only use them as feature extractors to do the data argumentation.

As to evaluation metrics, precision is the ratio of correct classifications to the total predicted positives, measuring the model's ability to accurately identify positive cases. Recall is the proportion of actual positive labels correctly classified, assessing the model's sensitivity. The F1 Score, which is the harmonic mean of Precision and Recall, provides a balanced evaluation

by considering both false positives and false negatives. Accuracy (ACC) measures the overall correctness of the model by calculating the ratio of correctly classified users to the total users. AUC evaluates the model's ability to distinguish between classes, with a higher AUC indicating better performance in ranking positive instances higher than negative ones.

C. Hyperparameters Space

In this section, we will give the hyperparameters space of the parameters of all models.Here we give the hyperparameters space of used optimizers, AdamW (Loshchilov & Hutter, 2017) in Table 6.

Table 6. Hyperparameters space of optimizers.

Hyperparameters	Space
Learning rate	$\mathrm{LogUniform}(1e\text{-}5, 1e\text{-}2)$
Weight decay	$\{0, \mathrm{LogUniform}(1e\text{-}6, 1e\text{-}3)\}$

Subsequently, we will provide the hyperparameter space for each corresponding method based on its category, including tree-based models and deep learning models. First, we give the hyperparameters space of tree-based models, including XGBoost, CatBoost, and LightGBM, in Table 7. Next, we will give the hyperparameters space of classic deep learning models, including MLP, ResNet, and TabR, in Table 8. We follow the style in (Gorishniy et al., 2021) to describe the hyperparameter space.

D. Experiment result

This appendix provides the complete versions of Tables 1, 2 and 3. These tables contain all details of the experimental results for a comprehensive comparison.

Table 9 presents the full experimental results on the tele dataset, comparing MLFE with other popular methods. The results demonstrate that our method consistently achieves superior performance across all evaluation metrics.

Table 10 provides the complete results of our experiments on the Digital Sichuan Innovation Competition dataset. The dataset focuses on telecommunications fraud detection and contains rich intra-user and inter-user feature representations. As shown in the table, MLFE outperforms all competing methods, further validating its robustness and generalization ability.

Table 11 presents the complete ablation study results on the tele dataset. This study evaluates the impact of different feature configurations using three traditional tree-based models: XGBoost (XGB), CatBoost (CAB), and LightGBM (LGB). The results demonstrate that adding either inter-user or intra-user features significantly improves performance compared to the vanilla setting, which only uses the original features. Specifically, incorporating inter-user features leads to notable improvements in Recall and F1-score, emphasizing the importance of capturing user relationships. Meanwhile, adding intra-user features enhances Precision and AUC, highlighting the value of individual user characteristics. Our proposed MLFE method, which integrates both inter- and intra-user features, achieves the best performance across all models, validating its effectiveness in leveraging complementary feature perspectives.

Table 7. Hyperparameters space of tree-based models.

Hyperparameter	Space
XGBoost	
Alpha	$\{0, \text{LogUniform}(1e\text{-}8, 1e2)\}$
Col sample by level	$\text{Uniform}(0.5, 1)$
Col sample by tree	$\text{Uniform}(0.5, 1)$
Gamma	$\{0, \text{LogUniform}(1e\text{-}8, 1e2)\}$
Lambda	$\{0, \text{LogUniform}(1e\text{-}8, 1e2)\}$
Learning rate	$\text{LogUniform}(1e\text{-}5, 1)$
Max depth	$\text{UniformInt}(3, 10)$
Min child weight	$\text{LogUniform}(1e\text{-}8, 1e5)$
Subsample	$\text{Uniform}(0.5, 1)$
CatBoost	
Bagging temperature	$\text{Uniform}(0, 1)$
Depth	$\text{UniformInt}(3, 10)$
L2 leaf reg	$\text{LogUniform}(1, 10)$
Leaf estimation iterations	$\text{UniformInt}(1, 10)$
Learning rate	$\text{LogUniform}(1e\text{-}5, 1)$
LightGBM	
Num Leaves	$\text{UniformInt}(10, 100)$
Max Depth	$\text{UniformInt}(3, 10)$
min child weight	$\{0, \text{LogUniform}(1e\text{-}3, 1)\}$
min child sample	$\text{UniformInt}(2, 100)$
Subsample	$\text{Uniform}(0.5, 1)$
Col sample bytree	$\text{Uniform}(0.5, 1)$
Reg lambda	$\{0, \text{LogUniform}(1e\text{-}5, 1)\}$

Table 8. Hyperparameters space of classic deep learning models.

Hyperparameter	Space
MLP	
N_layers	$\text{UniformInt}(1, 8)$
Hidden dimension	$\text{UniformInt}(64, 512)$
Dropout	$\{0, \text{LogUniform}(1e\text{-}6, 1e\text{-}3)\}$
ResNet	
N_layers	$\text{UniformInt}(1, 8)$
dimensions(d)	$\text{UniformInt}(64, 512)$
d hidden factor	$\text{Uniform}(1, 4)$
Residual dropout	$\{0, \text{LogUniform}(0, 0.5)\}$
hidden dropout	$\text{Uniform}(0, 0.5)$
TabR	
D main	$\text{UniformInt}(96, 384)$
Context dropout	$\text{Uniform}(0, 0.6)$
Encoder numbers blocks	$\text{UniformInt}(0, 1)$
Predictor numumbers block	$\text{UniformInt}(1, 2)$
Dropout0	$\text{Uniform}(0, 0.6)$
N frequencies	$\text{UniformInt}(16, 96)$
Frequency scale	$\text{LogUniform}(1e\text{-}2, 100)$
D embedding	$\text{UniformInt}(16, 64)$

Table 9. Performance comparison of MLFE and existing methods in Tele Dataset.

Model	Precision	Recall	F1	ACC	AUC
XGB	0.8955	0.8224	0.7613	0.9063	0.9279
CAB	0.8984	0.8186	0.7578	0.9059	0.9302
LGB	0.8842	0.8254	0.7597	0.9036	0.9078
MLP	0.8851	0.8229	0.7571	0.9031	0.9138
ResNet	0.8880	0.8354	0.7722	0.9076	0.9221
TabR	0.9056	0.8336	0.7792	0.9129	0.9278
DeepSets	0.6111	0.6188	0.6149	0.8385	0.7576
GNN	0.8911	0.5960	0.7143	0.9061	0.7891
MLFE-XGB	0.9201	0.8679	0.8196	0.9254	0.9617
MLFE-CAB	0.9163	0.8615	0.8166	0.9254	0.9612
MLFE-LGB	**0.9215**	**0.8699**	**0.8286**	**0.9298**	**0.9631**

Table 10. Performance comparison of MLFE and existing methods in the Digital Sichuan dataset.

Model	Precision	Recall	F1	ACC	AUC
XGB	0.8490	0.8359	0.8424	0.9000	0.8830
CAB	0.8450	0.8385	0.8417	0.8992	0.8831
LGB	0.8575	0.8333	0.8453	0.9025	0.8841
MLP	0.8705	0.8103	0.8393	0.9008	0.8768
ResNet	0.8325	0.8154	0.8238	0.8885	0.8691
TabR	0.8863	0.8258	0.7651	0.8973	0.9152
MLFE-XGB	0.8909	0.8954	0.8931	0.9311	0.9217
MLFE-CAB	0.9175	**0.9082**	**0.9128**	0.9443	**0.9348**
MLFE-LGB	**0.9632**	0.8673	**0.9128**	**0.9467**	0.9258

Table 11. Ablation study on our MLFE method.

Model	Feature	Precision	Recall	F1	ACC	AUC
XGB	Vanilla	0.896	0.822	0.761	0.906	0.928
	+ Inter	0.911	0.858	0.810	0.922	0.960
	+ Intra	0.909	0.865	0.817	0.924	0.959
	MLFE	**0.920**	**0.868**	**0.820**	**0.925**	**0.962**
CAB	Vanilla	0.898	0.819	0.758	0.906	0.930
	+ Inter	0.906	**0.869**	**0.819**	0.924	0.959
	+ Intra	0.911	0.866	0.818	**0.925**	**0.962**
	MLFE	**0.916**	0.861	0.817	**0.925**	0.961
LGB	Vanilla	0.884	0.825	0.760	0.904	0.908
	+ Inter	0.906	0.868	0.818	0.924	**0.964**
	+ Intra	0.912	0.868	0.822	0.926	0.962
	MLFE	**0.921**	**0.870**	**0.829**	**0.930**	0.963