IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received February 5, 2020, accepted February 22, 2020, date of publication February 27, 2020, date of current version March 11, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.2976898

Anomalous Telecom Customer Behavior Detection and Clustering Analysis Based on ISP's Operating Data

FENG ZHENG AND QUANYUN LIU

Beijing Key Laboratory of Network System Architecture and Convergence, Beijing University of Posts and Telecommunications, Beijing 100876, China
Beijing Laboratory of Advanced Information Networks, Beijing University of Posts and Telecommunications, Beijing 100876, China

Corresponding author: Feng Zheng (zhengfeng@bupt.edu.cn)

This work was supported by the National Social Science Foundation of China under Grant 17AGL026.

ABSTRACT Mobile networks and smart phones have become ubiquitous in our daily life. Large amount of customer related telecom data from various sources are generated every day, from which diversified behavior patterns can be revealed, including some anomalous behaviors that are vicious. It becomes increasingly important to achieve both efficient and effective customer behavior analysis based on the telecom big data. In this paper, the Multi-faceted Telecom Customer Behavior Analysis (MTCBA) framework for anomalous telecom customer behavior detection and clustering analysis is proposed. In this framework, we further design the *hierarchical Locality Sensitive Hashing-Local Outlier Factor (hierarchical LSH-LOF)* scheme for suspicious customer detection, and the *Autoencoders with Factorization Machines (FM-AE)* structure for dimension reduction to achieve more efficient clustering. Hierarchical LSH-LOF is an improved algorithm of LOF, in which we design a hierarchical LSH process that selects the approximate k nearest neighbors from coarse to fine by gradually narrowing down the scope. Experiments show its superiority over KD-tree w.r.t searching speed. FM-AE exploits Factorization Machines for learning second order feature interactions, which we prove to be useful by designing comparative experiments with five dimension reduction algorithms. With the proposed MTCBA framework, efficient and effective telecom customer behavior analysis including anomalous customer behavior detection and clustering analysis is performed on the real world telecom operating data provided by one of the major Internet service providers (ISPs) in China. Meanwhile, interpretable clustering results of six clusters are obtained to provide valuable information for the precision marketing of telecom operators, criminal combating, and social credit system construction.

INDEX TERMS Anomaly detection, clustering analysis, behavior analysis, telecom operators, dimension reduction.

I. INTRODUCTION

With the development of wireless communication and mobile Internet, the functions of mobile phone are becoming increasingly diversified, no longer limited to making phone calls and sending text messages. Mobile phone has become one of the most important ways for people to obtain information, entertainment and establish connection with the outside world; mobile applications emerge in an endless stream, all of which depend on the wireless communication and data traffic services provided by Internet service providers. In order to meet the diversified customer demand for mobile traffic and communication services, operators must respond to market changes agilely by designing new strategies and package services and carrying out personalized marketing and recommendation scheme, so as to improve customer loyalty, avoid customer churn, and ensure the continuous growth of business revenue. With the scale of users continuously increasing and market environment changing, it is vital to achieve precision marketing by understanding the diversified telecom service consumption patterns of different types of customers. Apart from that, the analysis of telecom customer behavior can also help to find out suspicious telecom fraud, large-scale scalping and other illegal criminal behaviors that are harmful to the society, so as to combat crime and regulate market order.

Specifically, on one hand, because of the increasingly diversified usage of mobile phones, it becomes a more and more complex and imperative task for telecom operators to

The associate editor coordinating the review of this manuscript and approving it for publication was Canbing Li.

provide appropriate packages in order to achieve precision marketing. For example, aged people tend to use mobile phones mainly for making calls, business people for commercial affairs, and teenagers for mobile Internet surfing; meanwhile, it's quite common for one person to own several SIM cards for various purposes. Different user groups have different needs for telecom packages: some want more network traffic flow, while others are satisfied with basic call function. How to achieve precise and concise customer clustering under the background of complex usage pattern in order to increase profit needs to be explored. On the other hand, despite the above normal cases, some SIM cards, i.e. phone numbers, are manipulated by cyber-criminals for illegal business activities. Specifically, some criminals use disguised phone numbers to conduct telecom fraud behavior; some scalpers own large quantities of phone numbers, and when an App begins to send coupons to new users, they register a lot of new accounts with these phone numbers and scalp coupons for economic benefit. How to detect these abnormal phone numbers and combat criminal is also worth studying.

Consequently, it becomes vital to achieve the mining of customer group patterns as well as anomaly detection based on the phone number behavioral data provided by telecom operators. However, existing research of telecom customer clustering and anomaly detection either use Apps' traffic flow data which does not include telecom package consumption information, or use particular scenario-related data, such as bank exchange data and e-commerce consumption data. There do exist some researches about telecom user clustering analysis for precision marketing, but most of them overlook the existence of anomalous users.

In this work, we first finish the anomalous behavior detection task of telecom customers, and then achieve interpretable customer clustering in order to boost precision marketing. Our contributions are as following:

(1) We propose the Multi-faceted Telecom Customer Behavior Analysis (MTCBA) framework for anomalous telecom customer detection and clustering analysis;

(2) We design a hierarchical LSH process for LOF that can achieve faster approximate nearest neighbor searching speed, and thus improve anomaly detection efficiency;

(3) We combine FM with Autoencoders which can achieve better dimension reduction performance, so as to enhance clustering results with interpretability;

(4) We adopt the proposed MTCBA framework and algorithms to solve real world problem of telecom customer behavior analysis based on operating data provided by an ISP.

The rest of this paper is organized as following: In Section II, we review previous work; in Section III, we introduce the MTCBA framework, hierarchical LSH-LOF and FM-AE in detail; in Section IV, experiments are performed and results are presented; in Section V, the paper is concluded.

II. RELATED WORKS

Previous researchers have accumulated extensive studies with different varieties of ISP's telecom data. According to Weiss [1], telecom data can be categorized into call detail data, network data, and customer data, based on which data mining applications may include fraud detection, marketing/customer profiling, and network fraud isolation.

For researches on call detail data mining, Subudhi and Panigrahi [2] adopted quarter-sphere SVM for finding fraudulent mobile phone calls by comparing recent and past usage pattern. An anomaly is detected when the current pattern of a user does not match with any of the individual's normal patterns. Sahin *et al.* [3] systematically explored fraud in telephony networks by constructing their taxonomy for differentiating the root causes, vulnerabilities, exploitation techniques and types of telecom fraud. Their proposed taxonomy helped in better understanding fraud and to mitigate it. Sultan *et al.* [4] firstly used k-means to remove anomalous record in the call detail records dataset, then trained a neural network model with anomaly-free data and respectively observed the MSE performance on anomaly and anomaly-free data, and finally adopted (ARIMA) model for predicting users' future traffic. Their experiments proved that anomaly-free data better generalizes the learning models and performs better on prediction task.

For researches on network data mining, Xu *et al.* [5] designed a time series analysis approach for modeling network traffic patterns of 9,000 cellular towers deployed in a metropolitan city, which was able to decompose large scale mobile traffic into regularity and randomness components. They also revealed that the prediction of regularity component can be achieved but of randomness component impossible. Finkelstein *et al.* [6] used machine learning models to reveal various demographics and technical computer skills of smartphone users based on their Internet traffic records. Their study showed that smartphone users can be classified by their gender, smoking habits, software programming experience, etc. Polpinij and Namee [7] proposed the Generalized Sequential Patter (GSP) algorithm for sequential pattern mining. They used GSP for extracting interesting patterns of inappropriate user behaviors in real event logs from an organization in Thailand, which improved the QoS of the Internet service.

For researches on customer data mining, Cheng *et al.* [8] proposed MQSFLA-k algorithm based on k-means and Multivariable Quantum Shuffled Frog Leaping Algorithm, which can be used for customer segmentation in telecom customer retention and marketing. Their experiments proved MAQFLA's advantages on both convergence rate and accuracy. Idris *et al.* [9] proposed GP-AdaBoost which combines the searching capabilities of genetic programming (GP) with the classification abilities of AdaBoost for predicting telecom customer churn. They also used Particle Swarm Optimization (PSO) undersampling to address the imbalance issue of the dataset. Zhang *et al.* [10] proposed a comprehensive multi-dimensional data user behavior expression method including the time, space and behavior semantics for the mining of user behavior pattern and providing personalized service and management. Their experiments were conducted

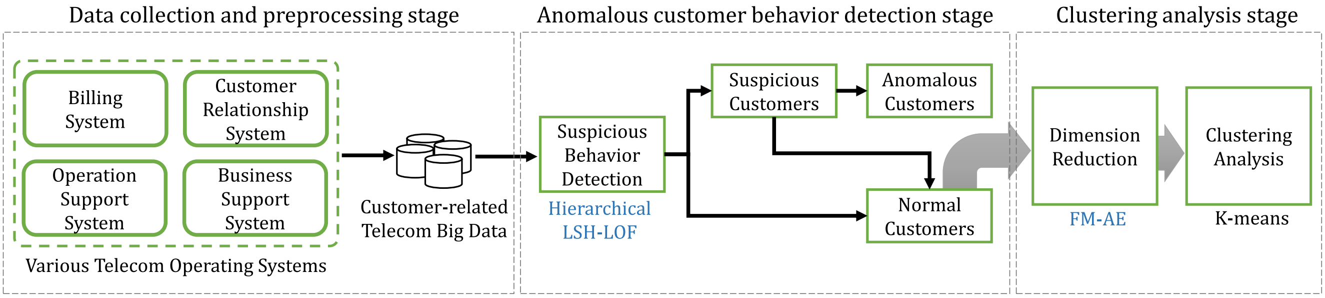

Data collection and preprocessing stage
Anomalous customer behavior detection stage
Clustering analysis stage
Billing System
Customer Relationship System
Operation Support System
Business Support System
Various Telecom Operating Systems
Customer-related Telecom Big Data
Suspicious Behavior Detection
Hierarchical LSH-LOF
Suspicious Customers
Anomalous Customers
Normal Customers
Dimension Reduction
FM-AE
Clustering Analysis
K-means

FIGURE 1. **Multi-faceted telecom customer behavior analysis framework.**

on one-day http connection log of all mobile users of a carrier in a city.

To do behavior analysis on our customer-related telecom big data, unsupervised local anomaly detection and dimensionality reduction algorithms are used, so in this part, we review some researches concerning these fields. For unsupervised local anomaly detection methods, as is reviewed by Goldstein and Uchida [11], they can be divided into two categories: the nearest-neighbor based and the cluster-based methods. In nearest-neighbor based family, Local Outlier Factor [12] firstly introduced the idea of local anomalies, followed with improved algorithms such as Connectivity-based Outlier Factor (COF) [13], Influenced Outlierness (INFLO) [14], Local Outlier Probability (LoOP) [15], Local Correlation Integral (LOCI) and Approximate Local Correlation Integral (aLOCI) [16]. In cluster-based family, Local Density Cluster-based Outlier Factor (LDCOF) [17] and Clustering-based Multivariate Gaussian Outlier Score (CMGOS) are included. For dimension reduction algorithms, it can be divided into linear and non-linear methods. The mostly used linear methods may include Principal Components Analysis (PCA) [18], Linear Discriminant Analysis (LDA) [19], etc.; non-linear methods may include kernel PCA [20], Isometric Feature Mapping (Isomap) [21], Locally Linear Embedding (LLE) [22], Autoencoders (AE) [23], t-distributed Stochastic Neighbor Embedding (t-SNE) [24], etc. Maaten *et al.* [25] classified kernel PCA, Isomap and AE into techniques preserving global properties, and LLE into techniques preserving local properties.

Most former researches on telecom data mining only focus on one specific aspect, such as detecting fraudulent behavior, constructing user models, or user classification. Few studied how to solve multiple mining tasks to achieve more effective exploitation with telecom customer data. Hence, our study will be focused on mining multi-faceted valuable information from customer-related telecommunication data, including anomalous behavior detection and customer clustering.

III. MTCBA: MULTI-FACETED TELECOM CUSTOMER BEHAVIOR ANALYSIS FRAMEWORK

A. MTCBA FRAMEWORK

In this study, we propose a multi-faceted telecom customer behavior analysis framework, which includes three stages: data collection and preprocessing stage, anomalous customer behavior detection stage and clustering analysis stage, as is shown in Fig.1.

In the first stage, customer-related telecom data is extracted from various telecom operating systems, and then preprocessed and stored in the database for later use. In the second stage, our proposed hierarchical LSH-LOF algorithm is used for finding suspicious customers, i.e. outliers, whose behavior patterns greatly deviate from the majority, and with further inspection, we classify the whole group into anomalous customers and normal customers. In the last stage, dimension reduction using the proposed FM-AE structure is adopted to achieve effective customer clustering for the high dimensional telecom data. Details and algorithm will be talked about in the following Section III-B, III-C, and III-D.

B. MULTI-SOURCE TELECOM BIG DATA

The goal of our study is to analyze the behavior of telecom customers from various aspects, so as to provide valuable information for developing effective marketing strategies for telecom operators as well as for fraudulent and criminal detection. Thus, multi-source telecom data need to be collected in order to achieve this goal. Specifically, customer-related telecom data from various operating systems including billing system, customer relationship system, operation support system and business support system are extracted, preprocessed and centrally stored in the database for later use; data cleaning, one-hot encoding and normalization are also included in this step. Then the data is prepared to be used as the input of later modules.

Tab. 1 and Tab. 2 list part of the four types of features extracted from different sources of operating systems. Note that "ARPU" stands for "average revenue per user", and "DOU" stands for "dataflow of usage"; both are metrics commonly used operators to evaluate customer value and activity level; "comm-circle" is an abbreviation of "communication circle".

Data used in this study is provided by one of the major Internet service providers in China. The original data contains operating records of all customers from a south-west province, which are randomly sampled without replacement with a sampling ratio of 0.01 for our research study. To ensure the sampled dataset being representative, non-probability sampling approaches are not considered,

TABLE 1. Customer-related telecom data from the billing system and customer relationship system.

Source	Features
Billing system	account balance
	package price of current month
	activating international call service or not
	activating international roaming service or not
	activating national free flow package in the past year or not
	activating holiday flow package in the past year or not
	activating 4G free time flow package in the past year or not
	average times of package change in the past 3 months
	average times of service activating in the past 3 months
	average times of complaints in the past 3 months
	average times of payment in the past 3 months
	times of payment in the past 3 months
	average ARPU in the past 3 months
	average ARPU
	total shutdown times
	total shutdown days in the past 3 months
	times of arrears in the past 3 months
	amount owed in the past 3 months
	total shutdown times in the past 3 months
Customer relationship system	number of mobile phone numbers in comm-circle
	number of landline phone numbers in comm-circle
	proportion of China Mobile users in comm-circle
	number of mobile phone numbers in strong comm-circle
	number of landline phone numbers in strong comm-circle
	number of mobile phone numbers in outbound comm-circle
	number of landline phone numbers in outbound comm-circle
	average ARPU of mobile users in comm-circle
	average DOU of mobile users in comm-circle
	monthly average call duration of mobile users in comm-circle
	monthly average call duration of landline users in comm-circle
	monthly average call times of mobile users in comm-circle
	monthly average call times of landline users in comm-circle

TABLE 2. Customer-related telecom data from the operation support system and business support system.

Source	Features
Operation support system	number of apps used in the current month
	number of apps used in the last month
	number of apps used the month before last
	DOU in current month
	DOU in last month
	DOU in the month before last
	average DOU
	average DOU in the past 3 months
	average active phone call days per month
	average call times in the past 3 months
	average call duration in the past 3 months
	average call times during idle time in the past 3 months
	average call duration during idle time in the past 3 months
	average number of messages sent in the past 3 months
	times of provincial roaming in the past year
	duration of provincial roaming in the past year
	number of provincial roaming places in the past year
	international roaming times in the past year
	duration of international roaming time in the past year
	International roaming places in the past year
Business support system	number of mobile phone numbers under ID card
	credit score
	length of access
	customer star degree
	key customer degree
	blacklist or not
	age
	gender
	current account points
	points used
	using 4G terminal or not
	enterprise customer or not
	binding family service or not
	contract customer or not

because the sampling results of these approaches cannot represent the original population. Other probability sampling approaches such as stratified sampling or systematic sampling are not chosen either to avoid bias that will make the sampling result not as representative as the result of random sampling. After sampling, we have 488,370 records in total, each with 75 features (before data preprocessing), and every record is uniquely represented by a mobile phone number.

C. HIERARCHICAL LSH-LOF FOR SUSPICIOUS BEHAVIOR DETECTION

As is mentioned in Section I, there exists a group of suspicious customers whose behavior patterns deviate greatly from the majority. Among these customers, some are normal users with a strong preference for certain kinds of telecom services, while other are users related with criminal activity for illegal benefits, such as telecom fraud groups, coupon scalpers, illegal service providers, etc.

Because of the unusual behavior patterns of suspicious customers, they are regarded as outliers of the group. Since the existence of outlier points may disturb the clusters' shape, the performance of customer clustering analysis will be heavily affected if such minorities are not separated from the majority. Hence, we design a anomalous customer behavior detection stage before the later clustering analysis stage, which can help us obtain finer customer clusters with better interpretability, as well as provide clues for criminal combating and social credit system construction.

To achieve the aim of effective and efficient suspicious customer detection, i.e. outlier detection, on our high dimensional and large amount telecom big data, we propose a hierarchical LSH-LOF algorithm, which is an improved algorithm of *local outlier factor(LOF)* with hierarchical *locality sensitive hashing(LSH)*. LSH is adopted here to replace the currently used KD tree search to reduce searching time. In the following subsections, we will make a brief introduction of LOF and LSH, and elaborate hierarchical LSH-LOF.

1) LOCAL OUTLIER FACTOR

The *local outlier factor* algorithm [12] is a nearest-neighbor based local outlier detection algorithm performed in an

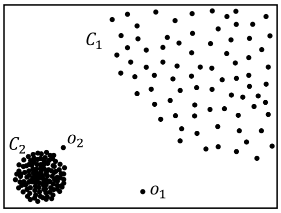


FIGURE 2. An example of the usage scenario of LOF.

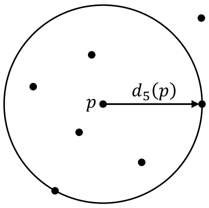


FIGURE 3. The *k*-distance of point *p*.

unsupervised way. LOF is good at detecting outliers in the dataset with clusters of different density. Fig. 2 gives us an intuitive example, where C_1 and C_2 are two clusters of different density, and o_1 and o_2 are two separated points. LOF can perform well on detecting such outliers. This property of LOF satisfies our need for detecting different types of suspicious customers with respect to customer clusters with different density, so we choose LOF as the basic outlier detection algorithm. In following part, the LOF algorithm will be reviewed.

Let $d(p, o)$ be the distance between point p and point o. The k-distance of point p is defined as:

Definition 1: For an integer $k > 0$, the k-distance of p, denoted as $d_k(p)$, is defined as the distance $d(p, o)$ between p and a point $o \in D$ such that:

- for at least k points, $o' \in D \setminus \{p\}, d(p, o') \leq d(p, o)$
- for at most $(k-1)$ points, $o' \in D \setminus \{p\}, d(p, o') < d(p, o)$

In a word, the k-distance of p is the distance between p and its k-th nearest neighbor, excluding p itself, as is shown in Fig. 3. Then the k-distance neighborhood of p, denoted as $N_k(p)$, is defined as all points with distance less than or equal to $d_k(p)$, therefore $\mid N_k(p) \mid \geq k$.

With k-distance defined, the reachability distance of p with respect to o is defined as:

$$reach - dist_k(p, o) = max\{d_k(o), d(p, o)\} \tag{1}$$

which means the reachability distances of those k nearest neighbors of o are all equal to the k-distance of o, i.e. $d_k(o)$. As is shown in Fig. 4, when $k = 5$, the reachability distance of p w.r.t o_1 is their real distance $d(p, o_1)$, and w.r.t o_2, it is $d_5(o_2)$.

Then the definition of *local reachability density* can be given as:

$$lrd_k(p) = 1 / \frac{\sum_{o \in N_k(p)} reach - dist_k(p, o)}{\mid N_k(p) \mid} \tag{2}$$

which is the reciprocal of the average reachability distance of points within $N_k(p)$. LRD can be seen as a density, where points in one cluster tend to have small reachability distance values, and thus high density values. Smaller LRD value means higher probability of the point being an outlier.

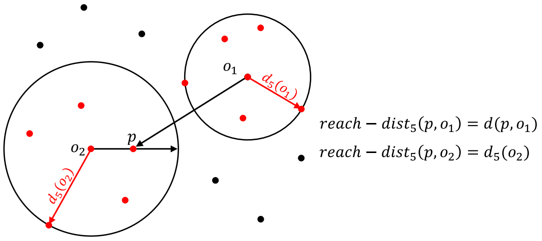

o_1
$d_5(o_1)$
o_2
p
$d_5(o_2)$
$reach - dist_5(p, o_1) = d(p, o_1)$
$reach - dist_5(p, o_2) = d_5(o_2)$

FIGURE 4. Reach distance.

With LRD defined, the *local outlier factor* can be defined as:

$$LOF_k(p) = \frac{\sum_{o \in N_k(p)} \frac{lrd_k(o)}{lrd_k(p)}}{\mid N_k(p) \mid} = \frac{\sum_{o \in N_k(p)} lrd_k(o)}{\mid N_k(p) \mid} / lrd_k(p) \tag{3}$$

which is an average value of the ratio of LRDs of $o \in N_k(p)$ to LRD of p. A point with LOF value close to 1 means it has similar density with its neighbors. The greater $LOF_k(p)$ value is than 1, the smaller relative density of p is, and the more likely point p being an outlier.

2) LOCALITY SENSITIVE HASHING

Nearest neighbor searching is the most time-consuming process in LOF, therefore we consider improving this process by designing new nearest neighbor searching method. *Locality sensitive hashing* [26] may be the most well-known approximate nearest neighbor searching scheme, which has been successfully applied to information retrieval, data mining and recommendation system. In this study, we apply LSH to improve the efficiency of LOF, and design a hierarchical structure to further speed up nearest neighbor searching. In following part, the LSH algorithm will be reviewed.

Assume we have n records stored in the database as a vector set $X = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n\}$, where $\mathbf{x}_i \in \mathbb{R}^m$. Given a query vector $\mathbf{q} \in \mathbb{R}^m$, we want to find the most similar vectors to $\mathbf{q}$ in X. The idea of LSH is to project the data into a lower dimensional binary space, i.e. a Hamming space $\mathbb{B}^p$, where each record of the original data is represented by a p-bit binary vector, a *hash key*. In another word, p is the length of a hash key, and a hash key is represented by a p-dimensional binary vector. If we can find an appropriate projection method, $\mathbf{q}$'s approximate nearest neighbors will be found in time sub-linear to n.

The hash key is obtained by applying p hash functions $h_1, h_2, \ldots, h_p$ to the record in the database. We want to find a family of hash functions satisfying this property: the closer two records are in $\mathbb{R}^m$, the higher probability of the two records sharing the same hash key in $\mathbb{B}^p$, and vice versa. Such property is called *locality sensitive*:

Definition 2: A family $\mathcal{H}$ of functions from $\mathbb{R}^m$ to $\mathbb{B}^p$ is called (r_1, r_2, P_1, P_2)-sensitive for $D(\cdot, \cdot)$ if for any $\mathbf{x}_i, \mathbf{x}_j \in \mathbb{R}^d$

- $Pr_{h \in \mathcal{H}}(h(\mathbf{x}_i) = h(\mathbf{x}_j)) \geq P_1, \quad \text{if } d(\mathbf{x}_i, \mathbf{x}_j) \leq r_1$
- $Pr_{h \in \mathcal{H}}(h(\mathbf{x}_i) = h(\mathbf{x}_j)) \leq P_2, \quad \text{if } d(\mathbf{x}_i, \mathbf{x}_j) \geq r_2$

where $D(\cdot, \cdot)$ is a distance function in the original space $\mathbb{R}^m$. Obviously, a family $\mathcal{H}$ is valid only when $r_2 > r_1$, and $P_1 > P_2$. Gionis *et al.* proved that given valid LSH functions, the query time for retrieving $(1+\epsilon)$-near neighbors is bounded by $O(n^{\frac{1}{1+\epsilon}})$for the Hamming distance [27].

The intuition of LSH is that we manage to let similar records collide (i.e. sharing the same hash key), then as an approximate nearest neighbor searching scheme, at query time, only those vectors with the same hash key need further calculation in the original space.

For the widely used inner product similarity $sim(\mathbf{x}_i, \mathbf{x}_j) = \mathbf{x}_i^T \mathbf{x}_j$, Charikar [28] proposed a hash function by rounding the inner product value with a random vector $\mathbf{r}$:

$$h_r(\mathbf{x}) = \begin{cases} 1, & \text{if } \mathbf{r}^T\mathbf{x} \geq 0 \\ 0, & \text{if } \mathbf{r}^T\mathbf{x} < 0 \end{cases} \tag{4}$$

where $\mathbf{r}$ can be seen as a random hyperplane from an m-dimensional Gaussian distribution $\mathcal{N}(0, I)$. Goemans and Williamson [29] proved that for such an $\mathbf{r}$,

$$Pr(h_r(\mathbf{x}_i) = h_r(\mathbf{x}_j)) = 1 - \frac{\theta(\mathbf{x}_i, \mathbf{x}_j)}{\pi} \tag{5}$$

where $\theta(\mathbf{x}_i, \mathbf{x}_j) = cos^{-1}(\frac{\mathbf{x}_i^T \mathbf{x}_j}{\|\mathbf{x}_i\| \|\mathbf{x}_j\|})$ is the angle between $\mathbf{x}_i$ and $\mathbf{x}_j$. Let $D(\mathbf{x}_i, \mathbf{x}_j) = \frac{\theta(\mathbf{x}_i, \mathbf{x}_j)}{\pi}$, it is obvious that if $D(\mathbf{x}_i, \mathbf{x}_j) \leq r_1$, then $Pr(h(\mathbf{x}_i) = h(\mathbf{x}_j)) \geq 1 - r_1$; if $D(\mathbf{x}_i, \mathbf{x}_j) \geq r_2$, then $Pr(h(\mathbf{x}_i) = h(\mathbf{x}_j)) \leq 1 - r_2$. Thus $P_1 = 1 - r_1$ and $P_2 = 1 - r_2$, where we have $P_1 > P_2$ provided $r_2 > r_1$, which satisfies the locality sensitive property.

In real cases, P_1 and P_2 can be very close to each other, so an amplification process of concatenating the output of several hash functions is often adopted. Apart from the above distance measure, the p-norm distance [30], Mahalanobis distance [31] and kernel similarity [32] also have been explored, which will not be introduced here.

Algorithm 1 Hierarchical LSH-LOF

```
Input: Dataset, $X = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n\}$; Outlier ratio, $\beta$;
    Number of nearest neighbors, $k$; Number of hash tables,
    $T$; Length of the hash key, $p$; Length of the remapping
    hash key, $m'$; Remapping candidate parameter, $\alpha$;
Output: Outlierness label vector, $O$;
1: initialize $T$ hash function sets for mapping $\mathbf{x}_i$ into $T$ buckets;
2: initialize $m'$ hash functions for remapping;
3: for each $\mathbf{x}_i \in X$ do
4:     calculate $\mathbf{x}_i$'s $p$-dimensional hash keys and map $\mathbf{x}_i$ into
       $T$ buckets using $T$ hash function sets;
5:     calculate $\mathbf{x}_i$'s $m'$-dimensional hash key using $m'$
       remapping hash functions;
6: end for
7: for each $\mathbf{x}_i \in X$ do
8:     calculate the union set of points in those $T$ buckets with
       $\mathbf{x}_i$ in them as $candidate_i$;
9: end for
10: for each $\mathbf{x}_i \in X$ do
11:     for each $\mathbf{x}_j \in candidate_i$ do
12:         calculate the Hamming distance between $\mathbf{x}_i$ and $\mathbf{x}_j$
            in the remapped space;
13:         sort $\mathbf{x}_j$ by the Hamming distance, and select the top
            $\alpha k$ points as final $candidate'_i$;
14:         calculate the Euclidean distance between $\mathbf{x}_i$ and
            points in $candidate'_i$, and select the top $k$ points as
            final $k$ nearest neighbors;
15:     end for
16: end for
17: run $LOF$ with $k$ nearest neighbors of $\mathbf{x}_i$, and decide
    outlier threshold according to $\beta$;
18: return Outlierness label vector, $O$
```

3) HIERARCHICAL LSH-LOF

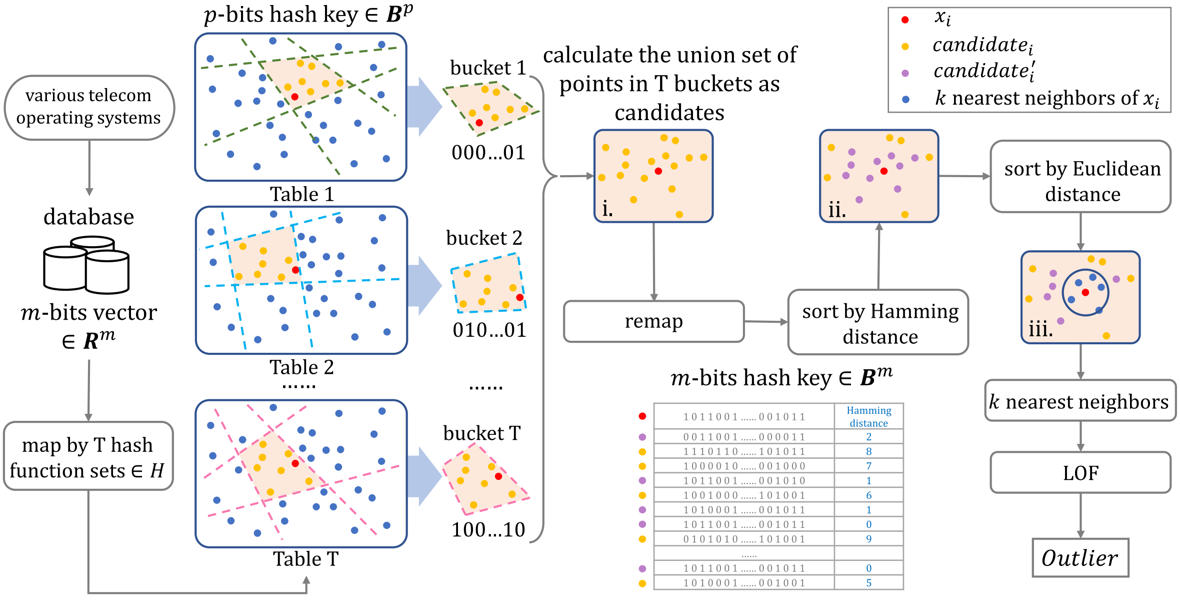
With the knowledge of LOF and LSH, we now introduce our proposed hierarchical LSH-LOF method for detecting suspicious telecom customers. Flowchart of the whole process is shown in Fig. 5, and the pseudo code is shown in Alg. 1.

The hierarchical LSH-LOF method firstly adopts LSH for finding k nearest neighbors, which is then used as the input of LOF for finding outliers. The LOF algorithm has been reviewed in Sec. III-C.1, so we leave it out in Alg. 1 (line 17). In following part, we will describe the hierarchical LSH in detail.

Our proposed LSH for approximate k nearest neighbor searching process can be hierarchically divided into three steps: the candidate finding step (line 1-9), the final candidate selecting step (line 10-13), and the k nearest neighbors selecting step (line 14).

(i) For the candidate finding step, firstly, T hash function sets $S_t(\cdot) = \{h_{1,t}(\cdot), h_{2,t}(\cdot), \cdots, h_{p,t}(\cdot)\}$ are initialized, where $h_{s,t}(\cdot)(1 \leq s \leq p, 1 \leq t \leq T) \in \mathbb{R}^m$ are uniformly and independently generated from an m-dimensional Gaussian distribution $\mathcal{N}(0, I)$ at random. Secondly, for each data point $\mathbf{x}_i$ in the database, it is mapped into T "buckets" by calculating its inner product with $h_{s,t}(\cdot)$ following with a rounding process mentioned in Equation 4. Each bucket here is represented by a p-bits hash key, which is actually a sub-space of the original data space separated by p hyperplanes. Data points with the same hash key are mapped into the same bucket. T hash function sets result in T kinds of space separating ways, and thus T hash tables. Thirdly, a union set of points in T buckets with $\mathbf{x}_i$ in them is calculated as the candidate neighbors of $\mathbf{x}_i$, denoted as $candidate_i$.

(ii) For the final candidate selecting step, firstly, m' hash functions $h_1(\cdot), h_2(\cdot), \ldots, h_{m'}(\cdot)$ are initialized, where $h_M(\cdot)(1 \leq M \leq m') \in \mathbb{R}^m$ are uniformly and independently generated from an m-dimensional Gaussian distribution $\mathcal{N}(0, I)$ at random. Secondly, for each data point $\mathbf{x}_j$ in $candidate_i$, it is remapped into the Hamming space $\mathbb{B}^{m'}$ by calculating its inner product with $h_M(\cdot)$ following with a rounding process mentioned in Equation 4. Thirdly, the Hamming distance of the projected vectors between $\mathbf{x}_i$ and $\mathbf{x}_j$


various telecom operating systems
database
m-bits vector $\in \boldsymbol{R}^m$
map by T hash function sets $\in H$
p-bits hash key $\in \boldsymbol{B}^p$
Table 1
Table 2
......
Table T
bucket 1
000...01
bucket 2
010...01
......
bucket T
100...10
calculate the union set of points in T buckets as candidates
i.
remap
sort by Hamming distance
ii.
sort by Euclidean distance
iii.
k nearest neighbors
LOF
Outlier
x_i
$candidate_i$
$candidate'_i$
k nearest neighbors of x_i
m-bits hash key $\in \boldsymbol{B}^m$

● 1 0 1 1 0 0 1 ...... 0 0 1 0 1 1	Hamming distance
● 0 0 1 1 0 0 1 ...... 0 0 0 0 1 1	2
● 1 1 1 0 1 1 0 ...... 1 0 1 0 1 1	8
● 1 0 0 0 0 1 0 ...... 0 0 1 0 0 0	7
● 1 0 1 1 0 0 1 ...... 0 0 1 0 1 0	1
● 1 0 0 1 0 0 0 ...... 1 0 1 0 0 1	6
● 1 0 1 0 0 0 1 ...... 0 0 1 0 1 1	1
● 1 0 1 1 0 0 1 ...... 0 0 1 0 1 1	0
● 0 1 0 1 0 1 0 ...... 1 0 1 0 0 1	9
......	
● 1 0 1 1 0 0 1 ...... 0 0 1 0 1 1	0
● 1 0 1 0 0 0 1 ...... 0 0 1 0 0 1	5

FIGURE 5. The flowchart of our proposed hierarchical LSH-LOF.

in the remapping space is calculated. Fourthly, points in $candidate_i$ are sorted by this Hamming distance, and the top αk points are selected to form the final candidate neighbors of $\mathbf{x}_i$, denoted as $candidate'_i$.

(iii) For the k nearest neighbors selecting step, the Euclidean distances between $\mathbf{x}_i$ and points in $candidate'_i$ are calculated, and the top k points are selected as the final k nearest neighbors of $\mathbf{x}_i$.

Additionally, we would like to talk about the intuition of designing such a hierarchical structure. In general, the whole process is to gradually narrow down the scope of approximate nearest neighbors by adopting different strategies from coarse to fine.

The first candidate finding step aims for fast search with constant time complexity, while the existence of the real neighbors of $\mathbf{x}_i$ in $candidate_i$ needs to be ensured. If we only use one single hash table with long hash key bits, the Hamming space will be separated into too many buckets, and very few data points will fall into the same bucket. To address this problem, we design the strategy of using several (T) hash tables each with relatively short hash keys (p-bits). A larger p, on the one hand, can be more selective for finding the real nearest neighbors; on the other hand, T has to be set larger to ensure that the real neighbors collide at least once with $\mathbf{x}_i$. In practice, we can find a tradeoff between the two parameters p and T based on the application.

The second final candidate selecting step aims for further narrowing down the scope of nearest neighbors with relatively fast speed and acceptable accuracy. To achieve this goal, it would be a good idea to remap the data into another Hamming space with dimensionality larger than p, and perform sorting by Hamming distance. This step can be seen as a fine searching process compared with the coarse searching in the first step. We do not apply Euclidean distance calculation in $candidate_i$ because the number of points in $candidate_i$ is still very large, where the time consumption will not be acceptable.

The third k nearest neighbors selecting step aims for a relatively accurate search of k nearest neighbors within αk candidates, which is done by calculating Euclidean distance in the original data space. Note that although this step is an accurate searching process, the hierarchical LSH structure, as a whole, is still an approximate k nearest neighbors searching strategy.

In the proposed hierarchical LSH-LOF framework, LOF actually can be replaced by any other k nearest neighbor based anomaly detection method. We will not compare LOF with other methods under our framework, because the telecom data we use doesn't have ground-truth label, which makes it impossible for us to evaluation anomaly detection performance of different models. For reference, Goldstein and Uchida [11] not only gave a thorough overview of existing unsupervised anomaly detection methods, but also did extensive experiments on 10 public datasets to evaluate their advantages and weaknesses. It would be unnecessary to redo their experiments, so we will just quote their conclusion here: in most cases, nearest neighbor based algorithms perform better than cluster-based algorithms; LoOP achieves best performance on four datasets, but performs poorly in global anomaly detection cases; LOF is recommended if we previously know the dataset contains local anomalies.

D. DIMENSION REDUCTION FOR EFFECTIVE CUSTOMER CLUSTERING

After the anomalous customer detection stage, anomalous data records that have negative effects for clustering

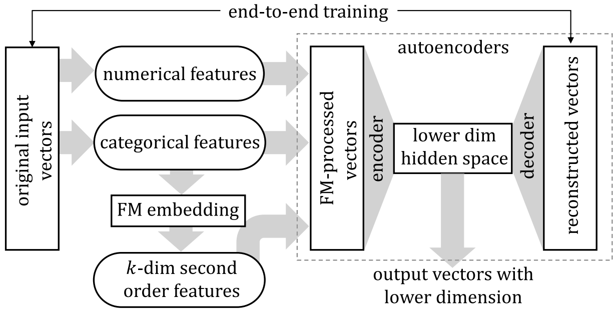


FIGURE 6. FM-AE: an end-to-end structure of FM embedding for learning second order feature interactions and autoencoders for dimension reduction.

performances have been removed. Before we do clustering analysis on the normal part, we first consider a dimension reduction process. Here, an end-to-end structure of FM embedding for learning second order feature interactions and autoencoders for dimension reduction is designed, as is shown in Fig. 6.

Factorization Machines (FM) [33] are machine learning models that can model all interactions between variables using factorized parameters, and thus they can be used for estimating interactions even in problems with huge sparsity (like recommender systems). There are 20 categorical features and 55 numerical features out of the total 75 features in our dataset. The 55 numerical features are normalized and then used as part of the input vector of the following autoencoders, but the 20 categorical features firstly need a one-hot encoding process which transforms them to numbers, and thus they can be taken as input of the autoencoders. The dimensionality and sparsity of the 20 categorical features will be even larger after one-hot encoding. Meanwhile, many of our categorical features are telecom service choices and customer-related properties, which inherently have strong correlations between each other. In order to learn the interactions between one-hot encoded sparse categorical features, we use the idea of FM here to learn the k-dimensional interaction feature vectors, where $k < 20$.

Autoencoders (AE) [34] are bottleneck-shaped symmetric neural networks with a lower dimensional hidden space layer in the middle. Once the autoencoders are trained, the decoder networks are usually discarded and the encoder networks are used for dimension reduction. Because of the nonlinearity of neural networks, autoencoders can achieve better dimension reduction performance than linear methods such as principal component analysis. What's more, it is also very convenient for autoencoders to be concatenated with other components to achieve end-to-end training.

As is depicted is Fig. 6, with sparse categorical features as its input, the FM embedding module learns their k-dimensional second order interaction features, which are concatenated to the original data features. The concatenated feature vectors are used as the input of the autoencoders, and then encoded and decoded to become the reconstructed output vectors of the autoencoders. Here, in order to achieve end-to-end training, we adjust the symmetric autoencoders to asymmetric. Specifically, the dimension of the autoencoders' output vectors is different from the autoencoders' input vectors, but is the same as the input vectors of the whole FM-AE structure.

The loss function of our FM-AE structure is

$$loss = \sum_{\mathbf{x}\in X} L\left(\mathbf{x}, g\left(f\left(concat\left(\mathbf{x}, FM(\mathbf{x}^{cat})\right)\right)\right)\right) \tag{6}$$

where $f(\cdot)$ and $g(\cdot)$ are the encoder and decoder networks correspondingly, $FM(\cdot)$ is the FM embedding module, $\mathbf{x}^{cat}$ is the data record with only categorical features, $concat(\cdot,\cdot)$ means the concatenation of features, and $L(\cdot,\cdot)$ is a distance function and l_2-norm is adopted here. Note that the dimension of the output vector of $g(\cdot)$ is the same as $\mathbf{x}$, and the dimension of the output vector of $f(\cdot)$ is much lower than $\mathbf{x}$.

Assume that $\mathbf{x}^{cat}$ is n-dimensional, and x_ix_j is the pairwise interaction of features in $\mathbf{x}^{cat}$, then $FM(\mathbf{x}^{cat})$ can be formulated as:

$$FM(\mathbf{x}^{cat}) = \sum_{i=1}^{n-1}\sum_{j=i+1}^{n} \omega_{ij}x_ix_j \tag{7}$$

where $\omega_{ij} = \langle \mathbf{v}_i, \mathbf{v}_j\rangle$ models the interaction between the i-th and j-th variable. Instead of using an own model parameter ω_{ij} for each interaction, the FM models the interaction by factorizing it. Rendle [33] proved that the pairwise interaction of features can be reformulated, i.e. factorized as:

$$\sum_{i=1}^{n}\sum_{j=i+1}^{n}\langle \mathbf{v}_i, \mathbf{v}_j\rangle x_ix_j = \frac{1}{2}\sum_{f=1}^{k}\left(\left(\sum_{i=1}^{n} v_{i,f}x_i\right)^2 - \sum_{i=1}^{n} v_{i,f}^2x_i^2\right) \tag{8}$$

Note that FM is used here merely to factorize those second order pairwise interaction features and get k-dimensional vectors that embody the correlations between features. For those first order numerical and categorical features, they are concatenated with the k-dimensional output vectors of FM, and then the concatenated vectors are used as the input of the autoencoders. Once the end-to-end training is finished, we discard the decoder networks and other components of FM-AE are retained for dimension reduction.

After the dimension reduction process, the high dimensional data has been embedded into a lower dimensional space, where clustering analysis can be carried out more effectively. We use k-means to cluster the data, and the detailed analysis of clustering results will be talked about in Section IV-B.3.

IV. EXPERIMENTAL ANALYSIS

A. EXPERIMENTAL DATASETS

The customer data provided by telecom operators does not have a clear label that indicates which kinds a customer belongs to and whether a customer is normal or not, and thus cannot be used to verify the performance of our proposed methods. Accordingly, in this part we firstly use the KDDCUP99 dataset [35] to do some experiments to verify the efficiency and effectiveness of hierarchical LSH-LOF; then we use the telecom customer data and design prediction experiments to prove the effectiveness of FM-AE; we finally apply the proposed MTCBA framework to solve real world customer behavior analysis task and get interpretable anomalous customer detection and customer clustering results.

The original KDDCUP99 dataset contains 4 million data records and 42 attributes, with 3.67% of records being network intrusion. We randomly sample the intrusion and non-intrusion part and construct a new dataset with 2% of records being network intrusion records, i.e. outliers.

As is mentioned in Section III-B, our telecom customer data contains 488,370 data records and 75 attributes. The attributes can be divided into four categories according to their sources: the billing system, customer relationship system, operation support system and business support system.

B. EXPERIMENTAL RESULTS

1) EFFICIENCY AND EFFECTIVENESS VERIFICATION OF HIERARCHICAL LSH-LOF

For efficiency verification, experiments of time consumption for finding nearest neighbors are done on 100k to 500k KDDCUP99 dataset sizes, and the efficiency of our proposed hierarchical LSH is compared with KD tree [36], a well-known nearest neighbor searching scheme. Experimental results are shown in Fig. 7. We can see from the line chart that the time consumption of both algorithms increases as dataset size becomes larger, and hierarchical LSH always consumes less time than KD tree for finding nearest neighbors. Note that hierarchical LSH searches for the approximate nearest neighbors, while KD tree searches for the exact nearest neighbors, which means adopting hierarchical LSH sacrifices accuracy to some degree for higher efficiency.

The time complexity of using brute force to find nearest neighbor is $O(T_E \cdot N^2)$, where T_E stands for time consumption of calculating the Euclidean distances. Using KD tree algorithm, the single search complexity is $O(\log N)$ and the total time cost is $O(T_E \cdot N \log N)$. Using hierarchical LSH, there are averagely $T \cdot N/2^p$ points in the $candidate_i$ set. For each query, it takes $O(T_H \cdot T \cdot N/2^p)$ to calculate the Hamming distance for these points. The final k nearest neighbors selecting step takes $O(T_E \cdot \alpha k)$. The time consumption for initializing T hash function sets is $O(mpTN)$ and for initializing m' hash funstions for remapping is $O(mm'T \cdot N/2^p)$. Practically, the hash function calculation is done offline. Therefore, the total time cost of hierarchical LSH is $O(T_H \cdot T \cdot N/2^p + N \cdot T_E \cdot \alpha k + mpTN + mm'T \cdot N/2^p)$. Theoretically, larger p will result in faster searching but lower precision. According to our experiments, setting $p = 8$ and $T = 6$ can lead to satisfying speed and acceptable performance.

For effectiveness verification, we firstly use the nearest neighbor recall as the evaluation metric to verify the effectiveness of hierarchical LSH for finding nearest neighbors, and experiments of recall performance with different α values and remapping hash key lengths are done on KDDCUP99 dataset.

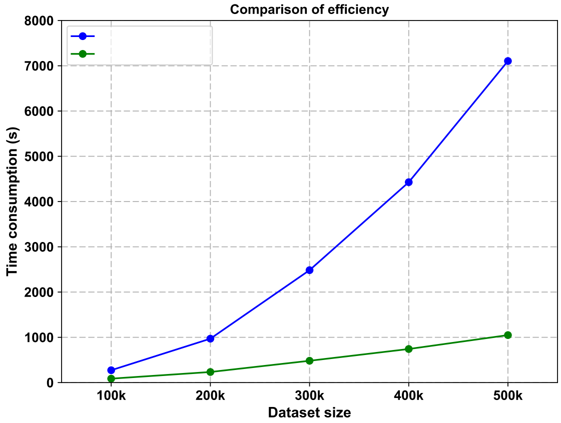


FIGURE 7. Comparison of the efficiency of KD tree and hierarchical LSH for neighbor searching.

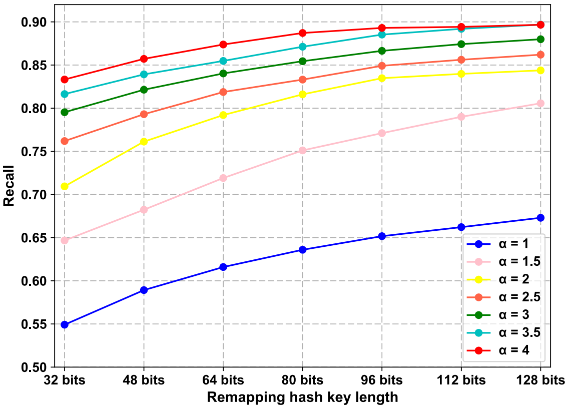


FIGURE 8. Recall performance w.r.t different α values and remapping hash key lengths.

Note that here the exact nearest neighbors are firstly calculated and stored, and then compared with the approximate nearest neighbors returned by hierarchical LSH, and a recall value is calculated based on these two results. Experimental results are shown in Fig. 8. We can see from the line chart that the best recall can be up to 0.9, which is a quite satisfying result for an approximate nearest neighbor searching scheme. Longer remapping hash key lengths bring about higher recall; Larger $candidate_i'$ sets (i.e. larger α values) result in better performance, but the performance improvement gradually becomes less apparent with α becoming larger. In practice, we can get satisfactory outcome with $\alpha = 3.5$ and $m' = 128$.

We then use the AUC as evaluation metric to verify the effectiveness of hierarchical LSH-LOF for finding outlier points. Based on previous experiments, we set $p = 8$, $T = 6$, $\alpha = 3.5$ and $m' = 128$. Experiments are done on KDDCUP99. The AUCs of hierarchical LSH-LOF and KD tree LOF reaches 0.9023 and 0.9160 respectively. Thus, conclusion can be drawn that hierarchical LSH-LOF can reach comparable performance while much faster speed than KD tree LOF. For the LOF outlier detection algorithm, substituting exact nearest neighbor searching scheme with approximate scheme will result in much higher efficiency along with very little degradation of detection performance.

2) EFFECTIVENESS VERIFICATION OF THE FM-AE DIMENSION REDUCTION PROCESS

Dimension reduction is a commonly used method for overcoming the curse of dimensionality. However, dimension reduction unavoidably leads to information loss, which degrades the performance of later tasks. A good dimension reduction method should be able to preserve the most valuable information as much as possible, and thus bring about less side-effect for accomplishing later tasks.

Based on this idea, in order to compare the dimension reduction performance of our proposed FM-AE with other dimension reduction algorithms, the ''later task'' is designed as using Xgboost [37] to predict three attributes in our telecom customer data: the ''credit score'', ''blacklist or not'', and ''customer star degree'' attributes. The input data of Xgboost are processed by different dimension reduction methods. In theory, Xgboost will achieve better prediction performance if its input data are processed by a better dimension reduction method. In our experiments, we compare FM-AE with PCA, kernel PCA, LLE, t-SNE, plain autoencoders, and the condition without dimension reduction, i.e. using the original high dimensional data as input. Because ''credit score'' and ''customer star degree'' are numerical features and ''blacklist or not'' is a categorical feature, we use the credit socre prediction RMSE, customer star degree prediction RMSE and blacklist or not prediction AUC as evaluation metrics. For the setting of parameters and network structure, we set the hidden space dimensionality as 10 for all methods compared; The autoencoders' structure in the plain autoencoders and FM-AE are [75-64-32-10-32-64-75] and [(75+k)-64-32-10-32-64-75] respectively. Both autoencoders are trained with the learning rate of 1e-2, batch size of 20,000 and run for 700 epochs. RBF kernel is adopted for kernel PCA, the number of neighbors in LLE is set to 10, and the exact method of gradient calculation is used in t-SNE. Experimental results are shown in Tab. 3.

We can see from the table that the best prediction performance is achieved without using any dimension reduction method because no information is lost, but this will cause much more computational cost for later clustering analysis process. The FM-AE method also achieves comparable performance with the no dimension reduction method while greatly reduces computational cost. The performance of FM-AE is better than the other five models, which proves the effectiveness of the FM module for learning feature interactions. Also, FM-AE can achieve much better dimension reduction performance than the linear methods PCA, owing to the non-linearity of neural networks. In a word, using FM-AE for dimension reduction can achieve a good tradeoff between performances and computational complexity.

3) ANOMALOUS TELECOM CUSTOMER BEHAVIOR DETECTION AND CLUSTERING RESULTS ANALYSIS

In this part, firstly, we elaborate how we classify the suspicious customers detected by hierarchical LSH-LOF depending on JS divergence and the defined w coefficient. Secondly, the clustering with k-means is performed and characteristics of the obtained 6 clusters are analyzed.

TABLE 3. Experimental results for the effectiveness verification of the FM-AE dimension reduction process.

Dimension reduction method	Credit score prediction RMSE	Blacklist or not prediction AUC	Customer star degree RMSE
PCA	20.5023	0.951976	0.514517
Kernel PCA	18.2691	0.992357	0.503402
LLE	21.3232	0.983778	0.618755
t-SNE	16.2769	0.977893	0.647565
Autoencoders	15.8421	0.973765	0.490291
FM-AE	15.3692	0.981412	0.488524
Not applied	13.9018	0.997707	0.277062

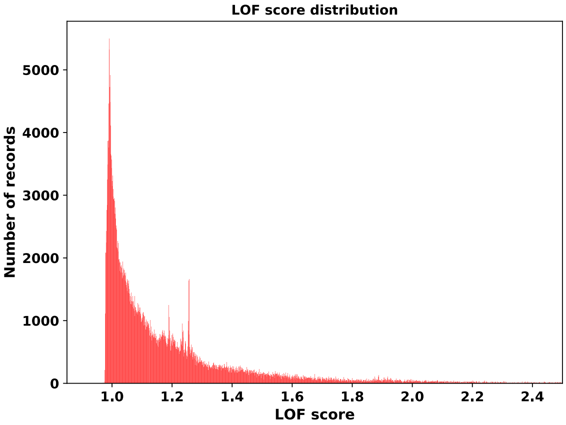


FIGURE 9. The LOF score distribution.

(i) Anomalous telecom customer behavior detection.

As is introduced in Section III-C.1, the outlierness of a point can be quantitatively described by its LOF score. The larger LOF score is than 1, the more likely the point being an outlier. By running hierarchical LSH-LOF, we get the LOF scores' distribution of the telecom customer data, as is displayed in Fig. 9. By setting the outlier ratio to be 3%, we can get the outlier threshold 2.0756. In another word, 97% of the data are deemed to be inliers, whose LOF scores are smaller than 2.0756. It can also be understood as a point is deemed to be an outlier if its neighbors' average LRD values are 2.0756 times larger than its own LRD value, and the LRD value can be understood as the ''density''. The largest LOF value reaches 65.6178, the second largest 34.0511 and the smallest 0.9717. The average LOF score is 1.2187. The LOF scores of 16.76% of data are less than 1, 79.78% within [1,2), and 2.57% within [2,3); only 0.20% of data have LOF scores larger than 5, and 0.03% larger than 10.

With the LOF scores obtained and threshold defined, the data are separated into two categories: inliers and outliers. In order to figure out what kinds of customer behavior deviate from the majority, i.e. the constituents of outliers, we hope to get enlightenment by comparing the distribution of their features.

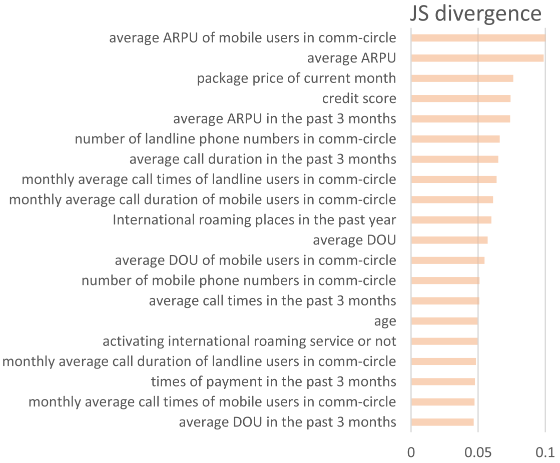


FIGURE 10. Features with high JS devergence values.

The Jensen-Shannon divergence (JS divergence) can measure the similarity of two distributions:

$$JS(P \parallel Q) = \frac{1}{2}KL(P(x) \parallel \frac{P(x)+Q(x)}{2}) + \frac{1}{2}KL(Q(x) \parallel \frac{P(x)+Q(x)}{2}) \qquad (9)$$

where $KL(\cdot)$ is the Kullback-Leibler divergence $D(P \parallel Q) = \sum P(x)log\frac{P(x)}{Q(x)}$; $P(x)$ and $Q(x)$ are two distributions. The smaller JS divergence is, the more similar two distributions are, and vice versa. Hence, we calculate the JS divergence of the outlier distribution and inlier distribution w.r.t. all the features to find the most different features.

Fig. 10 displays the top 20 features with the highest JS divergence values. Among these 20 features, 8 are from the customer relationship system, 5 from the billing system, 5 from the operation support system and 2 from the business support system. To get more intuitive understanding, we visualize the distribution by plotting their histograms, and 8 of them are displayed in Fig. 11. The mean and variance values for these 8 features are listed in Tab. 4. Except for the age feature, the mean and variance values of the outlier group are always larger than those of the inlier group, which means that the distribution of outliers are less concentrated. Average age of the inlier group is smaller, partly because a lot of age values in this group are missing and recorded as 0, and partly because there are indeed more young people in this group.

By observing these distributions, we can find that the inliers are more likely to have normal distributions, power-law distributions or Rayleigh distributions, which are common distributions of events happened in nature. However, the distributions of outliers are more likely to be long tailed, more even or having higher spikes. Thus, we may infer that there are still several types of users with different behavior patterns in the outlier category.

TABLE 4. Mean and variance values of 8 features with the highest JS divergence w.r.t inliers and outliers.

Feature	in/outlier	Mean	Variance
Credic score	inlier	588.26	3205.74
	outlier	602.91	6076.90
Average ARPU	inlier	57.97	2649.68
	outlier	139.19	19522.04
Average ARPU of mobile users in comm-circle	inlier	73.36	912.18
	outlier	83.32	3235.15
Average DOU	inlier	1996.51	1.545e7
	outlier	5470.04	6.267e7
Average call duration in the past 3 months	inlier	1.621e4	3.406e8
	outlier	3.693e4	2.204e9
Number of mobile phone numbers in comm-circle	inlier	89.55	1.046e4
	outlier	208.94	1.322e5
Age	inlier	42.67	237.19
	outlier	34.49	264.87
Times of payment in the past 3 months	inlier	2.88	8.60
	outlier	4.85	43.73

There are great differences in communication patterns of outliers and inliers considering that 40% of the top 20 features are related with customer's communication circle. Therefore, starting from the communication circle related features, we find two interesting features: "number of mobile phone numbers in comm-circle" and "number of mobile phone numbers in strong comm-circle". Using these two features, we define a "weak communication" coefficient w as the ratio of the former to the later feature. Classification and analysis of outliers can be done based on this coefficient:

A large w means that a customer has a very long contact list, but only very few are his/her close friends, which is the usual pattern of mobile phone numbers used for telecom fraud, making harassing calls or sales calls.

For small w, there are two situations: both communication circle and strong communication circle are small, or both are large. The first situation may be phone numbers used for conducting scalping behaviors for illegitimate benefit. Scalpers usually have hundreds of mobile phone numbers using the lowest price package. Once a newly emerged App begins to send high-value coupon to newly registered users, the scalper will register many new accounts using these phone numbers, and sell these coupons with prices lower than their worth. The first situation may also be some very inactive users, who use this phone number as a spare. The second situation may include high-value customers, who frequently contact with friends by making phone calls and texting. To distinguish these two situation, the "average ARPU" features is added to help decision making: the large ARPU group are classified as high-value customers, and the rest as suspicious scalpers or inactive customers. The above conclusion drawn by analyzing w is shown in Fig. 12.

Despite from the above outlier types, there also exists a kind of customers with very large "account balance" absolute value or very large "amount owed in the past 3 months" value, reaching 100,000 or even more. The account balance of normal individual customers is often smaller than 1,000, most smaller than 100. By looking into the data, we find that

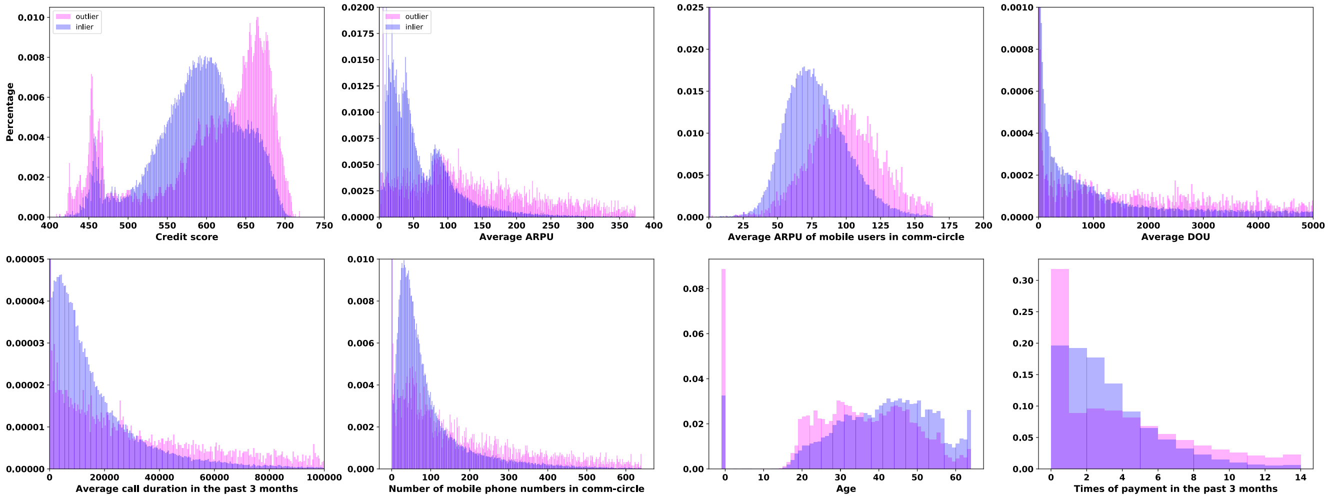


FIGURE 11. Distribution comparison of outliers and inliers w.r.t. various features.

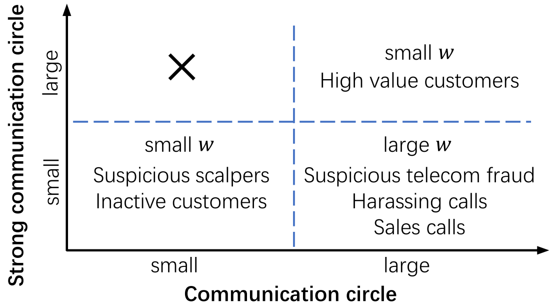


FIGURE 12. The weak communication coefficent.

such customers are enterprise customers, whose communication bills are usually later and quarterly paid. What's more, individuals can only apply for at most 5 phone numbers with one ID card, but enterprise customers are not restricted by this rule. Such different treatment results in different behaviors, and thus the deviation from the majority.

Still, there are some very obscure record in the outlier group that cannot be classified into any above categories. Their behavior randomly deviates from the majority with very random high value or low value features, which cannot be intuitively defined. We label these customers as "others", whose behaviors need further expert inspection.

Based on classification strategies mentioned above, the outliers are classified into 5 types. The corresponding pie chart is displayed in Fig. 13.

(ii) Clustering results analysis.

K-means is adopted here to perform the customer clustering, and the 10-dimensional hidden space vectors of FM-AE is used as its input data. The "elbow method" is used here to decide the clustering coefficient k. From Fig. 14 we can see that the "elbow" is at $k = 6$, and thus the number of clusters is set to 6. After running k-means, we obtain 6 customer clusters, and the percentages of them are 37.63%, 33.11%, 12.54%, 12.18%, 3.94% and 0.59% respectively.

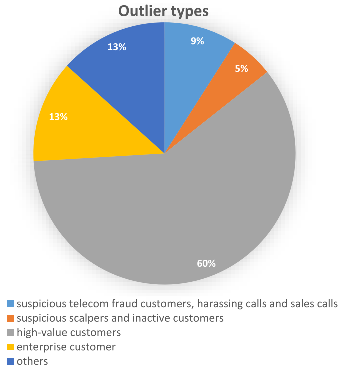


FIGURE 13. Outlier types.

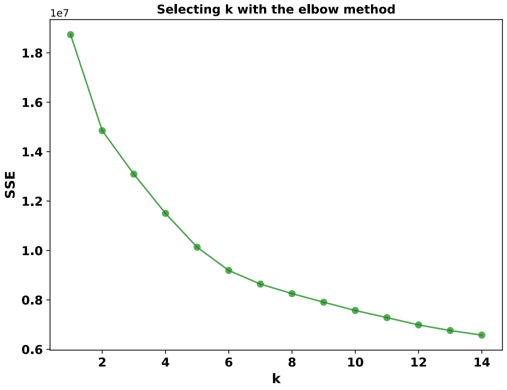


FIGURE 14. Selecting *k* with the elbow method.

In order to intuitively understand the behavior of these six types of users, we select 8 representative features and draw the histogram, which is shown is Fig, 15. The 8 features are: credit score, average ARPU, average DOU, age, length of access, average call times in the past 3 months, current account points, and points used. These features are selected because they are able to describe the communication behavior, Internet traffic usage behavior, loyalty, customer value,

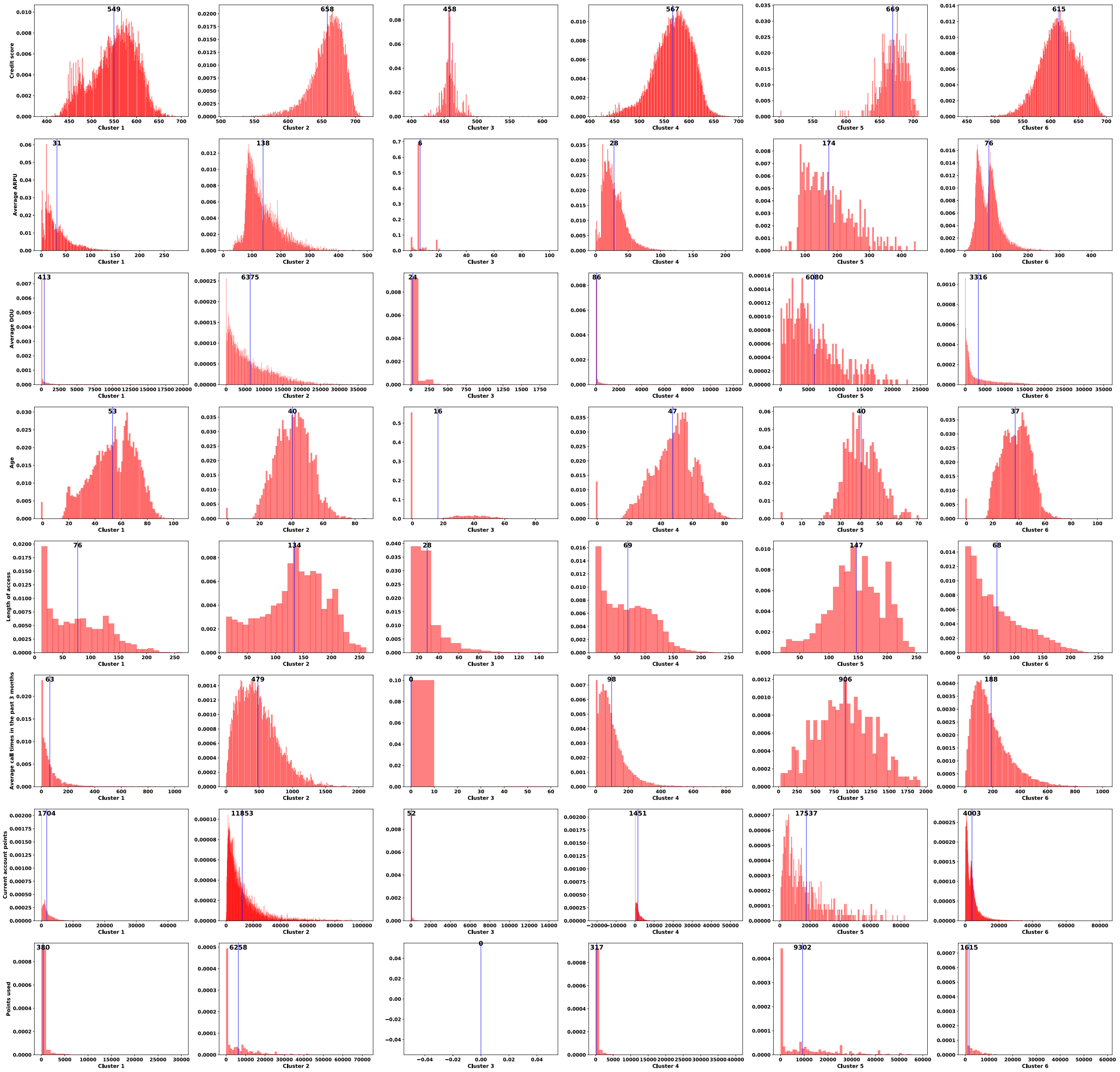


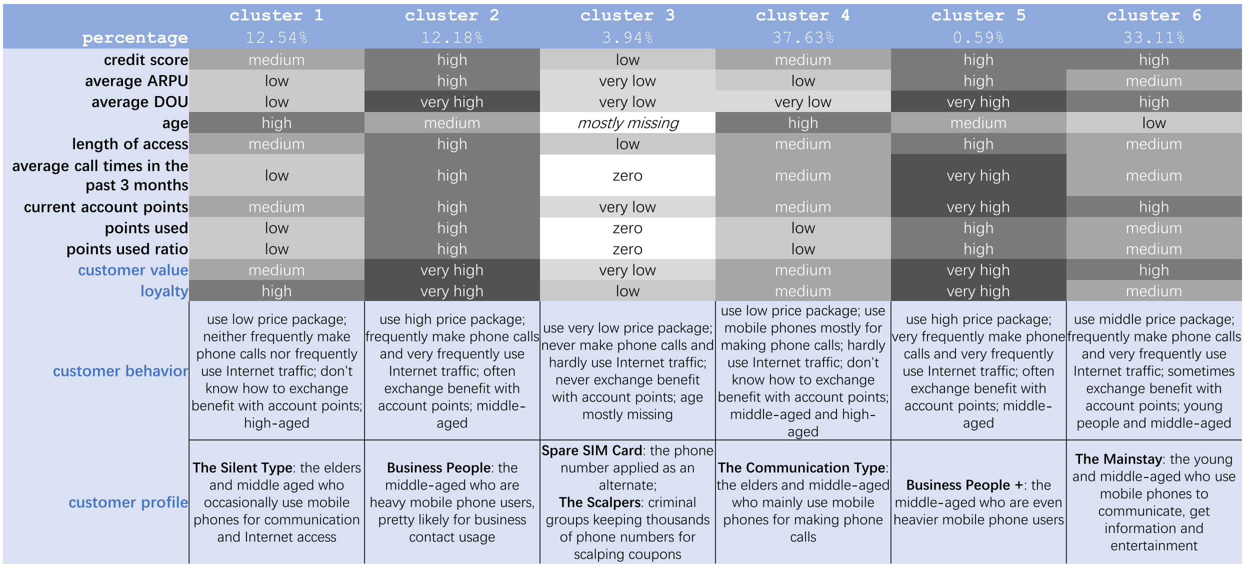
FIGURE 15. Distributions of 8 features of 6 clusters: credit score, average ARPU, average DOU, age, length of access, average call times in the past 3 months, current account points, and points used.

and activity level. The number displayed around the blue vertical line in each histogram is the mean of that distribution.

The behavioral differences of customers belonging to different clusters can be seen from the histograms. A detailed summary and analysis is shown in Fig. 16.

Cluster 1: we name this cluster as ‘‘The Silent Type’’, because neither mobile communication nor mobile Internet surfing attracts their interest. They rarely speak on social media, their communication circle is very small, and they are mostly the forties to the seventies. They may be the grandmothers occasionally call or send a WeChat message to their son and daughter, and except for this, their demand for communication service is very low.

Cluster 2: ‘‘Business People’’ may best describe this group of people. Their daily life and business activity heavily rely on mobile communication and mobile Internet accessing. They are the busiest thirties and forties who cannot afford missing any important message from their cell phone. In order to reduce the burden of monthly communication cost, they would like to participate in operator's promotional events and exchange traffic flow and communication bill with their account points.



percentage	cluster 1 12.54%	cluster 2 12.18%	cluster 3 3.94%	cluster 4 37.63%	cluster 5 0.59%	cluster 6 33.11%
credit score	medium	high	low	medium	high	high
average ARPU	low	high	very low	low	high	medium
average DOU	low	very high	very low	very low	very high	high
age	high	medium	*mostly missing*	high	medium	low
length of access	medium	high	low	medium	high	medium
average call times in the past 3 months	low	high	zero	medium	very high	medium
current account points	medium	high	very low	medium	very high	high
points used	low	high	zero	low	high	medium
points used ratio	low	high	zero	low	high	medium
customer value	medium	very high	very low	medium	very high	high
loyalty	high	very high	low	medium	very high	medium
customer behavior	use low price package; neither frequently make phone calls nor frequently use Internet traffic; don't know how to exchange benefit with account points; high-aged	use high price package; frequently make phone calls and very frequently use Internet traffic; often exchange benefit with account points; middle-aged	use very low price package; never make phone calls and hardly use Internet traffic; never exchange benefit with account points; age mostly missing	use low price package; use mobile phones mostly for making phone calls; hardly use Internet traffic; don't know how to exchange benefit with account points; middle-aged and high-aged	use high price package; very frequently make phone calls and very frequently use Internet traffic; often exchange benefit with account points; middle-aged	use middle price package; frequently make phone calls and very frequently use Internet traffic; sometimes exchange benefit with account points; young people and middle-aged
customer profile	**The Silent Type**: the elders and middle aged who occasionally use mobile phones for communication and Internet access	**Business People**: the middle-aged who are heavy mobile phone users, pretty likely for business contact usage	**Spare SIM Card**: the phone number applied as an alternate; **The Scalpers**: criminal groups keeping thousands of phone numbers for scalping coupons	**The Communication Type**: the elders and middle-aged who mainly use mobile phones for making phone calls	**Business People +**: the middle-aged who are even heavier mobile phone users	**The Mainstay**: the young and middle-aged who use mobile phones to communicate, get information and entertainment

FIGURE 16. Analysis of the 6 clusters.

Cluster 3: there may be two possible types of customers in this cluster: the "Spare SIM Card" type, i.e. people who use this phone number as an alternate number, and "The Scalpers", i.e. criminal groups conducting coupon scalping behavior. This is the most inactive and most anomalous group of customers. Their length of access is extremely low because those criminal groups are periodically destroyed and thus they have to apply for new phone numbers very often.

Cluster 4: this is the typical "Communication Type" of people who hardly use Internet traffic but only use mobile phones for making calls. Many people in this cluster choose the 18RMB monthly package, which can satisfy their basic communication need. The forties to the seventies are the majority.

Cluster 5: we call this cluster of people as "Business People +" because their behavior is very similar to cluster 2, only that they are even more loyal, more valuable and more active.

Cluster 6: this is "The Mainstay" cluster that contributes the most revenue to the telecom operator. They are quite active in both mobile communication and mobile Internet surfing. The average ARPU distribution of this cluster has two apparent spikes, because these people mostly choose the 48RMB and 88RMB monthly package. The majority of this group of people are the twenties, thirties and forties, who have strong need for the latest news and entertainment by accessing mobile Internet, and strong need for communicating with family and friends.

For customers in cluster 1 and cluster 4, they have loyalty, but very low willingness to use mobile traffic, so educating them to use mobile phones for Internet surfing would bring about more revenue. Still, the hardest task is teaching aged people to embrace the Internet. For customers in cluster 2 and 5, periodically sending traffic package to them would win their favor. It is a good idea to recommend package with more favorable price to them. For customers in cluster 3, expert inspection is needed to detect criminal acts in time. For customers in cluster 6, by tracing their package price change and DOU change, agile and fast marketing strategies are needed to avoid customer churn.

V. CONCLUSION

In this paper, the Multi-faceted Telecom Customer Behavior Analysis (MTCBA) framework for anomalous telecom customer behavior detection and clustering analysis is proposed. In this framework, we further design the hierarchical LSH-LOF scheme for suspicious customer detection and FM-AE structure for dimension reduction, which boost the performance with respect to computing speed and clustering results. With our proposed framework, efficient and effective telecom customer behavior analysis is performed and interpretable clustering results are obtained, which provide valuable information for precision marketing of telecom operators, criminal combating, and social credit system construction.

REFERENCES

[1] G. M. Weiss, "Data mining in telecommunications," in *Data Mining and Knowledge Discovery Handbook*. Boston, MA, USA: Springer, 2005, pp. 1189–1201.

[2] S. Subudhi and S. Panigrahi, "Quarter-sphere support vector machine for fraud detection in mobile telecommunication networks," *Procedia Comput. Sci.*, vol. 48, pp. 353–359, Jan. 2015.

[3] M. Sahin, A. Francillon, P. Gupta, and M. Ahamad, "SoK: Fraud in telephony networks," in *Proc. IEEE Eur. Symp. Secur. Privacy (EuroS&P)*, Apr. 2017, pp. 235–250.

[4] K. Sultan, H. Ali, and Z. Zhang, "Call detail records driven anomaly detection and traffic prediction in mobile cellular networks," *IEEE Access*, vol. 6, pp. 41728–41737, 2018.

[5] F. Xu, Y. Lin, J. Huang, D. Wu, H. Shi, J. Song, and Y. Li, "Big data driven mobile traffic understanding and forecasting: A time series approach," *IEEE Trans. Services Comput.*, vol. 9, no. 5, pp. 796–805, Sep. 2016.

[6] A. Finkelstein, R. Biton, R. Puzis, and A. Shabtai, "Classification of smartphone users using Internet traffic," 2017, *arXiv:1701.00220*. [Online]. Available: http://arxiv.org/abs/1701.00220

[7] J. Polpinij and K. Namee, "Internet usage patterns mining from firewall event logs," in *Proc. Int. Conf. Big Data Edu. (ICBDE)*, 2019, pp. 93–97.

[8] C. Cheng, X. Cheng, M. Yuan, C. Song, L. Xu, H. Ye, and T. Zhang, "A novel cluster algorithm for telecom customer segmentation," in *Proc. 16th Int. Symp. Commun. Inf. Technol. (ISCIT)*, Qingdao, China, Sep. 2016, pp. 324–329.

[9] A. Idris, A. Iftikhar, and Z. U. Rehman, "Intelligent churn prediction for telecom using GP-AdaBoost learning and PSO undersampling," *Cluster Comput.*, vol. 22, no. S3, pp. 7241–7255, Sep. 2017.

[10] W. Zhang, W. Zhou, and J. Luo, "Mining and application of user behavior pattern based on operation and maintenance data," in *Proc. IFIP/IEEE Symp. Integr. Netw. Service Manage. (IM)*, Washington DC, USA, Apr. 2019, pp. 614–618.

[11] M. Goldstein and S. Uchida, "A comparative evaluation of unsupervised anomaly detection algorithms for multivariate data," *PLoS ONE*, vol. 11, no. 4, Apr. 2016, Art. no. e0152173.

[12] M. Breunig, H. Kriegel, and R. Ng, "LOF: Identifying density-based local outliers," *ACM SIGMOD Rec.*, vol. 29, no. 2, pp. 93–104, 2000.

[13] J. Tang, Z. Z. Chen, A. Fu, and D. Cheung, "Enhancing effectiveness of outlier detections for low density patterns," in *Proc. 6th Pacific-Asia Conf. Knowl. Discovery Data Mining*. Berlin, Germany: Springer, 2002, pp. 535–548.

[14] W. Jin, A. Tung, J. Han, and W. Wang, "Ranking outliers using symmetric neighborhood relationship," in *Advances in Knowledge Discovery and Data Mining* (Lecture Notes in Computer Science), vol. 3918. Berlin, Germany: Springer, 2006, pp. 577–593.

[15] H. P. Kriegel, P. Kröger, E. Schubert, and A. Zimek, "LoOP: Local outlier probabilities," in *Proc. 18th ACM Conf. Inf. Knowl. Manage.*, New York, NY, USA, 2009, pp. 1649–1652.

[16] S. Papadimitriou, H. Kitagawa, P. B. Gibbons, and C. Faloutsos, "LOCI: Fast outlier detection using the local correlation integral," in *Proc. 19th Int. Conf. Data Eng.*, Alamitos, CA, USA, Mar. 2003, pp. 315–326.

[17] M. Amer and M. Goldstein, "Nearest-neighbor and clustering based anomaly detection algorithms for rapidminer," in *Proc. 3rd RapidMiner Community Meeting Conf.*, 2012, pp. 1–12.

[18] H. Hotelling, "Analysis of a complex of statistical variables into principal components," *J. Educ. Psychol.*, vol. 24, no. 6, p. 417, 1933.

[19] R. A. Fisher, "The use of multiple measurements in taxonomic problems," *Ann. Eugenics*, vol. 7, no. 2, pp. 179–188, Aug. 2012.

[20] B. Schölkopf, A. Smola, and K.-R. Müller, "Nonlinear component analysis as a kernel eigenvalue problem," *Neural Comput.*, vol. 10, no. 5, pp. 1299–1319, Jul. 1998.

[21] J. B. Tenenbaum, "Mapping a manifold of perceptual observations," in *Proc. Adv. Neural Inf. Process. Syst.*, 1998, pp. 682–688.

[22] S. T. Roweis, "Nonlinear dimensionality reduction by locally linear embedding," *Science*, vol. 290, no. 5500, pp. 2323–2326, Dec. 2000.

[23] G. E. Hinton, "Reducing the dimensionality of data with neural networks," *Science*, vol. 313, no. 5786, pp. 504–507, Jul. 2006.

[24] L. van der Maaten and G. Hinton, "Visualizing data using t-SNE," *J. Mach. Learn. Res.*, vol. 9, pp. 2579–2605, Nov. 2008.

[25] L. Van Der Maaten, E. Postma, and J. Van den Herik, "Dimensionality reduction: A comparative," *J. Mach. Learn. Res.*, vol. 10, pp. 66–71, Oct. 2009.

[26] P. Indyk and R. Motwani, "Approximate nearest neighbors: Towards removing the curse of dimensionality," in *Proc. 13th Annu. ACM Symp. Theory Comput.*, 1998, pp. 604–613.

[27] A. Gionis, P. Indyk, and R. Motwani, "Similarity search in high dimensions via hashing," *VLDB*, vol. 99, no. 6, pp. 518–529, 1999.

[28] M. S. Charikar, "Similarity estimation techniques from rounding algorithms," in *Proc. 34th Annu. ACM Symp. Theory Comput. (STOC)*, 2002, pp. 380–388.

[29] M. X. Goemans and D. P. Williamson, "Improved approximation algorithms for maximum cut and satisfiability problems using semidefinite programming," *J. ACM*, vol. 42, no. 6, pp. 1115–1145, Nov. 1995.

[30] M. Datar, N. Immorlica, P. Indyk, and V. S. Mirrokni, "Locality-sensitive hashing scheme based on p-stable distributions," in *Proc. 20th Annu. Symp. Comput. Geometry (SCG)*, 2004, pp. 253–262.

[31] P. Jain, B. Kulis, and K. Grauman, "Fast image search for learned metrics," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2008, pp. 1–8.

[32] B. Kulis and K. Grauman, "Kernelized locality-sensitive hashing for scalable image search," in *Proc. IEEE 12th Int. Conf. Comput. Vis.*, Sep. 2009, pp. 2130–2137.

[33] S. Rendle, "Factorization machines," in *Proc. IEEE Int. Conf. Data Mining*, Dec. 2010, pp. 995–1000.

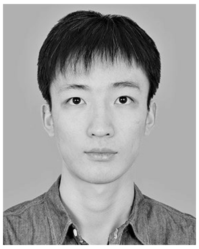
[34] M. A. Kramer, "Nonlinear principal component analysis using autoassociative neural networks," *AIChE J.*, vol. 37, no. 2, pp. 233–243, Feb. 1991.

[35] *KDDCUP99 Dataset*. [Online]. Available: https://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html

[36] J. L. Bentley, "Multidimensional binary search trees used for associative searching," *Commun. ACM*, vol. 18, no. 9, pp. 509–517, Sep. 1975.

[37] T. Chen and C. Guestrin, "Xgboost: A scalable tree boosting system," in *Proc. 22nd ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2016, pp. 785–794.

FENG ZHENG received the Ph.D. degree from the Beijing University of Posts and Telecommunications (BUPT), China, in 2012. She is currently an Associate Professor with the School of Information and Communication Engineering, BUPT. She has authored 31 articles and published 13 articles as the first author. She has participated in 20 national and enterprise-level scientific research projects. Her research interests are wireless communication and data mining.



QUANYUN LIU received the B.E. degree from the Xi'an University of Posts and Telecommunications, in 2017. He is currently pursuing the master's degree with the Beijing University of Posts and Telecommunications (BUPT), China. His research areas are data mining, machine learning, and telecom service provisioning.