Hindawi
Security and Communication Networks
Volume 2018, Article ID 5680264, 9 pages
https://doi.org/10.1155/2018/5680264

Research Article

A Model Based on Convolutional Neural Network for Online Transaction Fraud Detection

Zhaohui Zhang,[1,2,3] **Xinxin Zhou**,[1] **Xiaobo Zhang**,[1] **Lizhi Wang**,[1] **and Pengwei Wang**[1,2,3]

[1] *School of Computer Science and Technology, Donghua University, Shanghai 201620, China*
[2] *The Key Laboratory of Embedded System and Service Computing, Ministry of Education, Tongji University, Shanghai 201804, China*
[3] *Shanghai Engineering Research Center of Network Information Services, Shanghai 201804, China*

Correspondence should be addressed to Zhaohui Zhang; zhzhang@dhu.edu.cn

Received 20 May 2018; Accepted 16 July 2018; Published 6 August 2018

Academic Editor: Zhaoqing Pan

Using wireless mobile terminals has become the mainstream of Internet transactions, which can verify the identity of users by passwords, fingerprints, sounds, and images. However, once these identity data are stolen, traditional information security methods will not avoid online transaction fraud. The existing convolutional neural network model for fraud detection needs to generate many derivative features. This paper proposes a fraud detection model based on the convolutional neural network in the field of online transactions, which constructs an input feature sequencing layer that implements the reorganization of raw transaction features to form different convolutional patterns. Its significance is that different feature combinations entering the convolution kernel will produce different derivative features. The advantage of this model lies in taking low dimensional and nonderivative online transaction data as the input. The whole network consists of a feature sequencing layer, four convolutional layers and pooling layers, and a fully connected layer. Verifying with online transaction data from a commercial bank, the experimental results show that the model achieves excellent fraud detection performance without derivative features. And its precision can be stabilized at around 91% and recall can be stabilized at around 94%, which increased by 26% and 2%, respectively, comparing with the existing CNN for fraud detection.

1. Introduction

Using wireless mobile terminals has become the mainstream of Internet transactions, which can verify the identity of users by passwords, fingerprints, sounds, and images. The fraudster can collect users' information, such as ID, password, age, occupation, and other information, and logs in various trading systems as normal users to complete fraud. This kind of fraudulent behavior has been very common today in the rapid development of information technology, and it brings great losses to users, businesses, and society. Moreover, once these identity data are stolen, traditional information security methods will not prevent online transaction fraud.

Banks and major financial institutions provide a wide range of services, but the fraud is widespread in many financial transactions provided by these institutions. More services will generate more users' data, which provides a great possibility for fraudsters to steal users' information to complete fraud. How to detect fraudulent transaction accurately and instantly has become an urgent financial security problem for all financial institutions, including banks. The traditional expert rule system is applied to most fraud detection areas. These expert rule systems are based on the existing industry experience rules, which can detect the occurred fraudulent patterns and the existing fraud behaviors. However, online fraud transactions are very different from traditional transactions, so the traditional expert rule system is incapable of detecting and intercepting online fraud transactions effectively. The fraudulent transaction in this paper means that the fraudster embezzles the legitimate user's information and enters the trading system as a normal user.

A variety of machine learning and deep learning models are gradually applied in detecting fraud. Compared with the traditional rule system, the advantage of machine learning is its ability to use a large number of complex data to characterize some financial phenomena that are difficult to be found by traditional methods. Various models used for financial fraud detection include neural networks, deep neural networks, random forests, logistic regression, SVM [1–6], and so on. On the one hand, most of the existing models are applied to credit card fraud detection. On the other hand, credit card transactions differ from online transactions, so these models cannot be entirely suitable for online transactions.

The pros of neural networks and deep learning are that they can fully approximate any complex nonlinear relationships, strong robustness, and fault tolerance and find optimized solutions at high speed. It has an outstanding performance in image recognition [7–9], video processing [10], natural language processing [9], and other fields. But when dealing with the structured data, especially online transaction data, neural networks, and deep learning models have poor performance. Because the available dimensions of transaction data are often very limited, some massive features derived from most existing models and prior knowledge of the industry do not contribute to the learning [2, 11]. Consequently, this paper constructs a CNN model based on the feature sequencing for Internet transaction fraud detection. And compared with the existing CNN models, our model can achieve a better performance only by using the transaction data raw features as training.

The rest of this paper has been structured as follows. Section 2 introduces the application and effectiveness of existing machine learning and deep learning methods in financial fraud detection. Then, Section 3 describes the process of constructing a convolutional neural network model based on feature rearrangement. Later, we set up experiments and evaluate its performance in Section 4. Finally, Section 5 summarizes this paper and discusses future work.

2. Related Work

Antifraud system is the first line of defense for financial institutions. So, the antifraud system is widely adopted in banking, insurance, law, business administration, and other fields.

With the maturity of many machine learning algorithms and deep learning algorithms, they have been successfully used in image detection, text processing, and other fields. Gurusamy, R. and Subramaniam, V. [8] proposed a new method for the denoising, extraction and tumor detection on MRI images. They used a variety of machine learning algorithms to build brain image recognition systems to aid in medical diagnosis and medical evaluation. These methods include CNN and SVM, and these algorithms had a good result when they were applied in this scenario. Chengsheng Yuan [9] used a combination of CNN and SVM to build a live fingerprinting model and achieved good results. They use CNN for feature extraction and SVM for classification.

Machine learning and deep learning models are also used in the field of financial fraud detection. S.Ghosh and D.L.reilly [12] used the neural network algorithm to construct a transaction fraud detection system, which was verified in the credit card transaction data of Mellon Bank and applied to the actual transaction system in 1994. The model built by Bayesian belief network was also used for fraud detection. Sam Maes et al. [1] built a transaction fraud model using Bayesian belief networks. When applied to experimental data sets, it was found that Bayesian belief networks had higher recognition accuracy than neural networks. In some scenarios, constructing the recognition models with a single algorithm is inferior to combinatorial algorithms. V. Hanagandi et al. [13] constructed a credit card fraud scoring system by combining a radial basis function network with a density-based clustering algorithm. According to users' historical records, this system would generate fraud scores for decreasing credit fraud. It is hard to determine the network topology when using neural networks to build this model. Raghavendra Patidar et al. [14] made it easier to build a fraud detection model, by using a genetic algorithm to calculate this neural network topology, including the numbers of hidden layers and the numbers of nodes.

Existing neural networks often require high-dimensional data as input, which means that it is hard to obtain high dimensionality and strong availability transaction. The common solution is making derivative features for consumer behavior patterns based on industry experience, which reflects the users' behavior habits. The exploration of different legal consumer behavior patterns and fraudster behavior patterns is critical in fraud detection. A. I. Kokkinaki et al. [15] described legal consumer's transaction habit with a decision tree and Boolean logic method. Besides, they used clustering methods to analyze the distinction between normal or abnormal transaction. Kang Fu et al. [2] proposed using trading entropy and other derivative features based on industry experience to characterize user trading action. The average transaction amount, total amount, the difference between the current transaction amount and the average transaction amount, trading entropy, and other features generated from the raw data of the fixed time window were used as model input data. The trading entropy is a novel feature used to describe the user's transaction behavior. It can describe the relationship between the user's transaction amount and the total transaction amount over a period of time. These derived features can better reflect the characteristics of the user's transaction behavior under certain conditions.

Besides the above method of analyzing user's behavior from a data perspective, some scholars analyze user's abnormal behavior from the perspective of system behavior. Zhang and J. Cui proposed a method to discover user's abnormal behavior from a system perspective. Zhang Z., Ge L. [16] et al. proposed an effective way to solve user behavior anomalies through system behavior reconstruction.

In addition to neural network algorithms, logistic regression, support vector machines, random forest algorithms [3, 4], hidden Markov models [17, 18], and adversarial learning methods [19] are also widely applied in constructing credit fraud detection model. Most of these existing model algorithms are based on credit card transactions. Credit card transaction and online transaction are different in terms of transaction methods, transaction characteristics, trader behaviors, etc. [20]. The models based on credit transaction are not fully applied to the online transaction.

Most of the existing fraud detection models are constructed for the credit card transaction, which is not fully

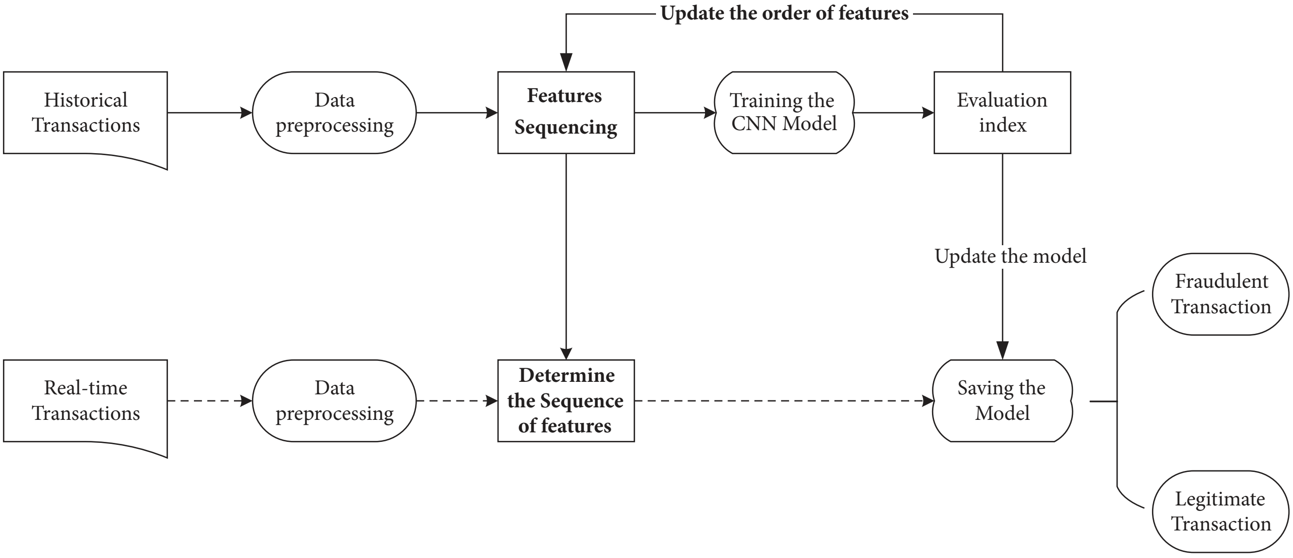

FIGURE 1: The structure of the model.

applicable to online transaction fraud detection. Most of the neural network models will need a large number of derivative variables in feature engineering; thus, these models cannot be applied to low dimensional transaction data. Therefore, the CNN based on feature sequencing is proposed to solve the problem of online transaction fraud detection.

3. CNN Based on Feature Sequencing

3.1. Framework. This paper builds the model based on the CNN to directly use the low dimensional raw features as the input of the model. The feature sequencing layer is added to automatically optimize the sequence of features. This approach can save variable derived time, take advantage of the CNN, learn the derivative features that are beneficial to the classification results, and reduce the interference of human experience with the model. In the fraudulent transaction, there are a lot of features and trading patterns which are not found, and the purpose of reducing the interference is to let the CNN learn the transaction characteristics and trading mode as much as possible.

The overall structure of the model is divided into two parts: the model training part and the transaction detection part. The training part of the model is divided into two parts: the feature sequencing layer and the CNN. The increased feature sequencing layer is used to optimize the sequence of transaction features. First, the historical data is cleaned and so on, then put data into the feature sequencing layer, and the model effect is tested by training the CNN model, and the feature sequence order is modified by the effect feedback. In update time, we can find out the optimal sequence mode by fixed feature permutation times. When the real-time data enters the model, the data features are sorted by the order of the feature, and then the training model is judged (Figure 1).

3.2. Feature Sequencing Layer. Transaction data consists of multidimensional features, and there is no direct connection between multidimensional features, so multidimensional attributes can be arranged randomly. If transaction data put into various model algorithms in the form of one-dimensional variables, the arrangement, and combination of different attributes will not affect the physical meaning of the record. But different permutations and combinations will affect the results of the model. This is essentially the same as image, speech, text, and other data. Take image data as an example: although the image can be translated, rotated, flipped, etc., it remains invariant during the conversion process, but the essence of the image is composed of ordered pixels. The position of these pixels is not allowed to change; otherwise, the inherent information carried by the image will change.

We use a 5-tuple to describe transaction data.

Definition 1. A transaction data M is a five-tuple composed of transaction features, feature arrangement state, position exchange operation, feature initial arrangement state, and feature final arrangement.

Formally as follows: $M = (Q, \Sigma, \delta, q_0, F)$

Q: a finite set representing transaction features

Σ: a finite set representing the different arrangements of transaction features

δ: exchange operations between transaction features

q_0: $q_0 \in Q$, transaction data features initial state

F: $q_0 \times \delta \longrightarrow F$, transaction data features state finally arranged

3.3. Feature Sequencing. Each fraudulent transaction consists of multiple transaction features. The arrangement of these trading features does not affect the physical meaning of the transaction, but different feature arrangements will have a different effect on the model after the convolution process. This is why we add the feature sequencing layer into the

```
Input: The weight matrix $A$, the rows of the weight matrix $A_1A_2...A_n$, initial state of the input set Q, list of model accuracy rates $Acc$, auxiliary sequence $c_1c_2...c_n$($c_j$ is the number of rows below $a_j$, satisfy the first condition $0 \le c_j \le j$) and $o_1o_2...o_n$($o_j$ control the direction of $c_j$ change)
Output: The best permutation of weight matrix.
1: $c_j \longleftarrow 0$
2: $o_j \longleftarrow 1(1 \le j \le n)$
3: $Acc \longleftarrow [\,]$
4: Access matrix $A$
5: $j \longleftarrow n$, $s \longleftarrow 0$ ($s$ is the number of $c_k$ satisfying $k > j$ and $c_k = k - 1$ )
6: $q \longleftarrow c_j + o_j$:
7: if $q < j$ then
8:     go to 19
9: end if
10: if $q = j$ then
11:     go to 14
12: end if
13: $A_{i-c_j+s} \longleftrightarrow A_{j-q+s}$, $c_j \longleftarrow q$, go to 4
14: if $j = 1$ then
15:     input $Q * A$ to model and calculate the $accuracy$
16:     $Acc.append(accuracy)$
17:     if $len(Acc) == n!$ then
18:         return max($Acc$) and $A$ that max($Acc$) corresponding
19:     end if
20: else
21:     $s = s + 1$
22: end if
23: $o_j = -o_j$, $j = j - 1$, go to 6
```

Algorithm 1: Find the best permutation of input features.

model. Each transaction feature has the potential to exist anywhere, in order to ensure that all the arrangements of features can be taken, so the nodes between the initial input layer and the final input layer are fully connected (Figure 3), but the connection weight changes during each iteration.

The reason why convolutional neural networks can accurately classify images is that they automatically find important classification features in a brute-force, mass-data fashion. This is both an advantage of CNN being able to identify precisely, and a disadvantage of it not being able to identify the location of the image effectively. So, whether the image is complete or not, we have reason to believe that CNN can identify the desired object on the map. For example, when a nose and eyes misplaced the face picture, CNN will still determine it as a normal face picture. A location problem similar to this problem also exists in the context of using CNN for transaction data identification.

In the model constructed in this paper, one-dimensional feature vector into the model, and the convolution layer is processed with one-dimensional convolution kernel feature vector. In the process of convolution, the principle is that as in image processing, information extraction is performed on partial features. Figure 2 depicts a feature vector convolution transformation process, such as 1 × 2 convolution kernel. The process of convolution is to select two adjacent features for convolution and generate derivative features. For the multivariate feature vectors, the sequence of the features is different. Naturally, the results of the single-layer convolution are directly affected. The CNN, in turn, contains multiple convolutional layers. After the layer convolution, the effect will lead to the identification of the entire model.

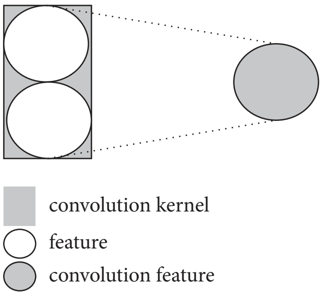

Figure 2: The convolution procedure process between two features.

This paper constructs a CNN model based on feature sequencing which adds a feature sequencing layer before the input layer. Firstly, the network structure of convolution layer, pooling layer, and fully connected layer are determined by data features, and the optimal permutation order of all permutations is determined by the feedback of model results. Then the model parameters are trained with the input that fixes the permutation. The time complexity of Algorithm 1 that aims at finding the optimal sequencing of all features is

```
**Input:** The weight matrix $A$, the rows of the weight matrix $A_1 A_2 \ldots A_n$, list of model accuracy rates $Acc$, two integers $i$, $j$, initial state of the input set Q, the iteration times that you want to do $m$
**Ouput:** The best permutation of weight matrix.
$A \longleftarrow n - DimensionalIdentityMatrix$
$Acc \longleftarrow [\,]$
$i \longleftarrow random(\,)$
$j \longleftarrow random(\,)$ $(0 \leq j \leq n-1)$
Access matrix $A$
$A_i \longleftrightarrow \mathrm{A}_j$
$Q \longleftarrow Q * A$:
input $Q$ to model and calculate the accuracy
$Acc.append(accuracy)$
**if** $len(Acc) == m$ **then**
    return max($Acc$), $A$ that max($Acc$) corresponding
**end if**
```

Algorithm 2: Find the optimal arrangement within a fixed number of times.

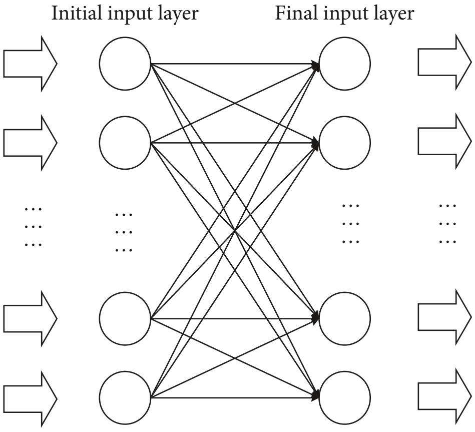

Figure 3: Feature sequencing layer.

O(n!). If the transaction data has more feature dimensions, the time complexity of the algorithm will be very high, which is not conducive to the construction and training of the model. Therefore, we also construct Algorithm 2 that randomly transforms feature arrangements and finds the optimal arrangement within a specified number of times and a fixed number of iterations. The algorithm can subjectively set the number of model iterations and can find better feature sequencing in a short period of time relatively.

Figure 2 shows the network structure of the feature sequencing layer. The number of features selected in the model is n, and the number of all arrangements of features is m. The initial input layer is the original input and final input layer is the input features after the sequence transformation. The order of the input features is changed through the transformation of initial input layer and final input layer connection weight matrices. Set the connection weight matrix A and initialize the connection matrix to A_0. Each time a matrix row is transformed, the connection weight matrix for the next iteration is generated.

We arrange the data features in the initial state Σ as a one-dimensional vector $\Sigma_0 = [\mathrm{x}_0, \mathrm{x}_2, \mathrm{x}_3, \ldots, \mathrm{x}_\mathrm{n}]$. The position transform operation can be expressed as the product of Σ and the connection matrix A. In special cases, the connection weight matrix is as shown in A_0, our model will degenerate into a regular CNN, and the original feature input order will not be changed.

$$A_0 = \begin{bmatrix} 1 & 0 & \cdots & 0 \\ 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots \\ 0 & \cdots & 0 & 1 \end{bmatrix} A_1 = \begin{bmatrix} 0 & 1 & \cdots & 0 \\ 1 & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots \\ 0 & \cdots & 0 & 1 \end{bmatrix} \cdots\cdots \mathrm{A}_{m-1} = \begin{bmatrix} 0 & 0 & \cdots & 1 \\ 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots \\ 1 & \cdots & 0 & 0 \end{bmatrix} \tag{1}$$

$$\Sigma_\mathrm{i} = \Sigma_{\mathrm{i}-1} \times A_i \tag{2}$$

If the feature sequencing layer connection matrix is as shown in formula (1), the first step of the transformation process is as follows:

$$\Sigma_1 = \Sigma_0 \times A_1 = [\mathrm{x}_1, \mathrm{x}_2\mathrm{x}_3, \cdots, \mathrm{x}_\mathrm{n}] \times \begin{bmatrix} 0 & 1 & \cdots & 0 \\ 1 & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots \\ 0 & \cdots & 0 & 1 \end{bmatrix} \tag{3}$$

$$= [\mathrm{x}_2, \mathrm{x}_1\mathrm{x}_3, \cdots, \mathrm{x}_\mathrm{n}]$$

3.4. CNN Network Structure with Feature Sequencing Layer.

Compared with the existing CNN model, the network structure of this model has a feature sequencing layer. The network

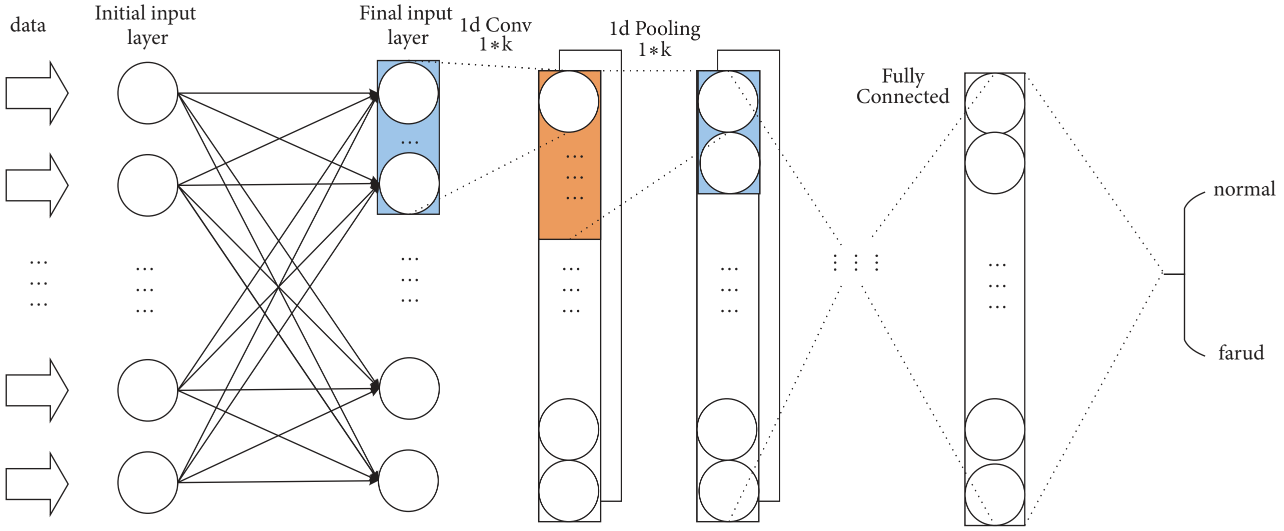

Figure 4: Overall network structure.

structure is designed to ensure that it can be applied to network transaction data and quickly identify online transaction data. The whole network consists of a feature sequencing layer, four alternating distribution convolutional layers and pooling layers, and a fully connected layer (Figure 4).

The feature sequencing layer is the order arrangement processing to the input features since convolution of the different order feature input layers results in different effects of the model. The convolutional layer function is to extract the local feature of the input data; in this scenario, we can understand that the convolutional layer will automatically derive new features based on the input features. These new derivative features, although we do not explain their physical meaning, are indeed helpful to the classification of the model. The pooling layer joins the features of the adjacent areas together into a single higher level feature that reduces the redundancy of the data. The fully connected layer plays the role of the final classification. For different input data, the number of nodes in each layer of the network varies. In my experiment, the channels are two, and the number of nodes in the fully connected layer is 144.

For each trained network model, we save the current model and compare it with the previous model. If the current model works better than the previous model, we replace the previously saved model with the current model so that these trained models can be directly applied to the detection of real-time trading data.

4. Experiment Verification and Analysis

4.1. Dataset. The experimental data of this paper comes from a commercial bank B2C online transaction data; a total of about 5 million transaction data are extracted for the experiment. The positive sample is approximately 33 times that of the negative sample; each transaction record has 62 dimensions. All transaction data has a time span of 6 months, and we take two samples to construct a more balanced experimental dataset of positive and negative samples. In order to ensure the sequential continuity of transaction data, we use one-month data batches as experimental data. When the model is trained and validated, we divide the data of one month, about 500,000, into training sets and test sets, and the ratio is about 3:1. In order to ensure the consistency and availability of data, we have done routine processing such as data cleaning, data transformation, and data reduction. All comparative tests were performed on the same dataset.

Based on the feature engineering methods in the literature [17, 18] and the statistical analysis of the raw data, we find that the characteristics of user transaction behavior, such as time, amount, and location, and other derived features are very significant in the fraud detection model. These users' data are also information that fraudsters usually steal. Combined with the results of the data cleaning, we select 8D features such as transaction ID (because of the data confidentiality, this paper does not mention all the data dimensions) as input to the model.

4.2. Model Validation. This paper uses accuracy, precision, recall, and F_1 score to evaluate the effectiveness of this model. In this paper, the CNN model based on feature sequencing is compared with the existing CNN[2] and BP neural network in the same data set (Figures 5–11). All comparative tests were performed on the same dataset. From the following six groups of test results, we can deduce that our model's precision rate can be stabilized at around 91% and recall rate can be stabilized at around 94%, which increased by 26% and 2%, respectively, compared with the existing CNN for fraud detection. At the same time, we also compared with the traditional BP neural network with two hidden layers. The effect of each index of this model has been greatly improved compared with the traditional BP neural network.

Firstly, this paper makes a detailed analysis and judgment on the data set using traditional BP neural network and optimizes the model by adjusting the number of nodes and

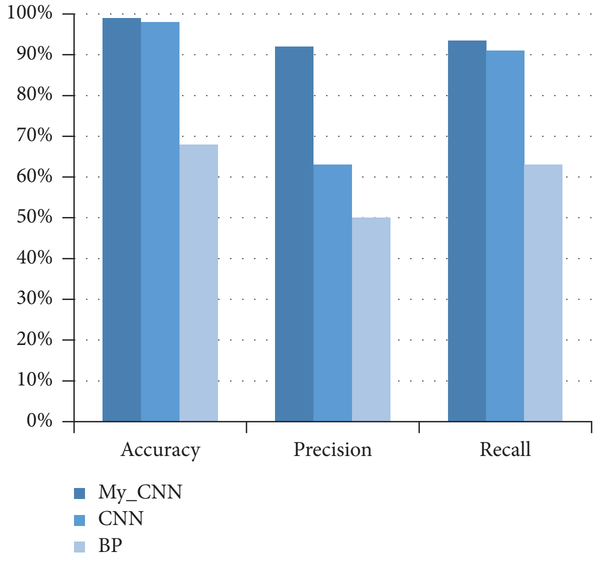

Figure 5: Different performances of three models on Set_1.

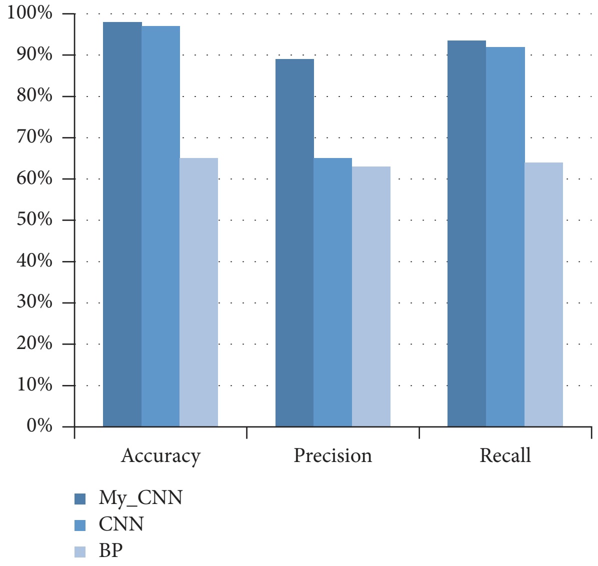

Figure 6: Different performances of three models on Set_2.

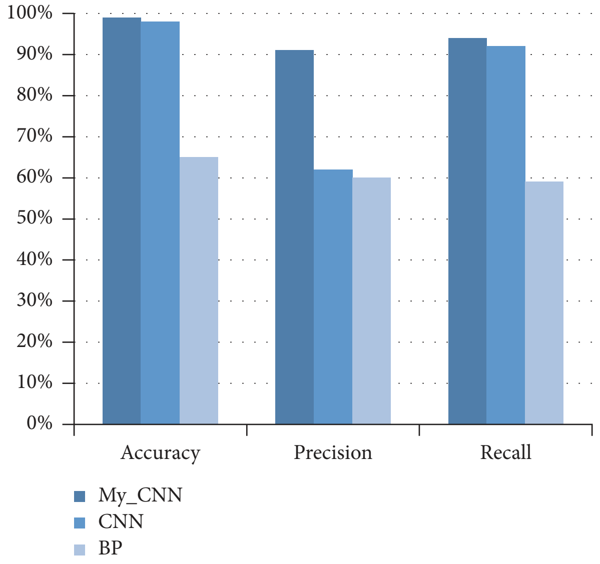

Figure 7: Different performances of three models on Set_3.

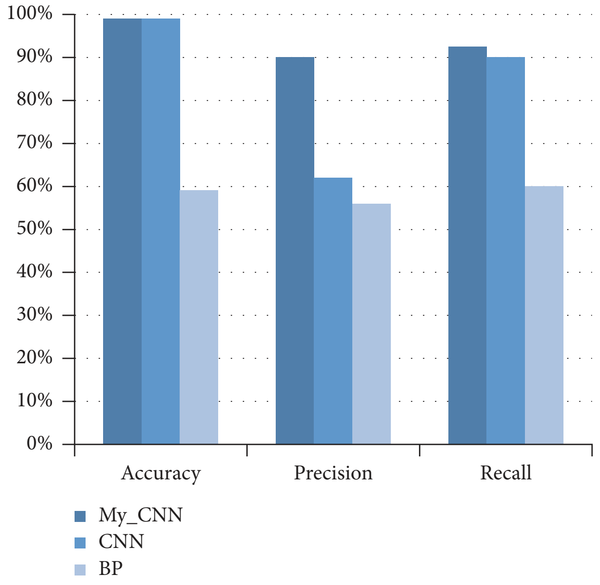

Figure 8: Different performances of three models on Set_4.

the number of layers of the network. However, the final model structure is not ideal, so this result is only used as the basic comparison work of our paper.

In order to verify whether the different sequencing of the features has an optimal effect on the model, we use Algorithm 2, set m=10, and record the performance of the 10 times (Figure 8). It can be seen from the experimental results that different feature sequencing methods have an effect on the model's results. The best experimental results in the ten sequences (the eighth) are better than the original ones, and if the computational capabilities allow, we will be able to find out more superior feature sequencing (Figure 12).

In terms of time performance, the same monthly data set was used to experiment: the BP neural network was significantly faster than the two convolutional neural networks; compared with the two convolutional neural networks, the epoch was 1, and the batch size was 1000. The feature sequencing convolutional neural network has a training time of 65 seconds for once feature arrangement. Model training time with ten times features arrangements is 752 seconds. The traditional CNN training time is 352 seconds without the feature derivative work. If we add the processing time of the derived features, the time performance of the model is far worse than our model.

The CNN model based on feature sequencing is compared with the existing convolutional neural network. The experiment shows that the model constructed in this paper is superior to the existing CNN model in the effect of each indicator and does not need to do a large number of derivative

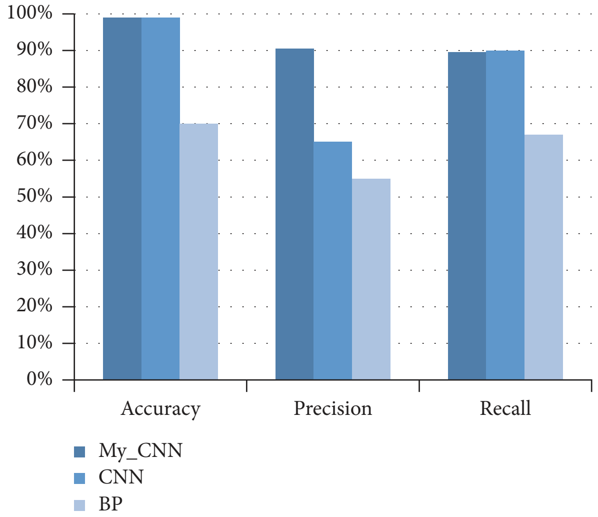

FIGURE 9: Different performances of three models on Set_5.

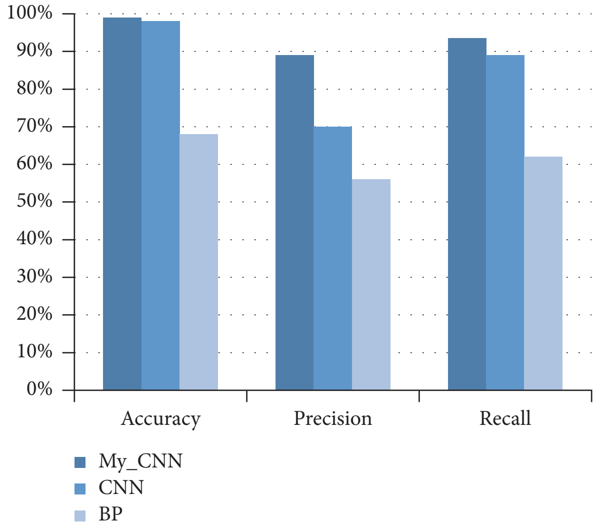

FIGURE 10: Different performances of three models on Set_6.

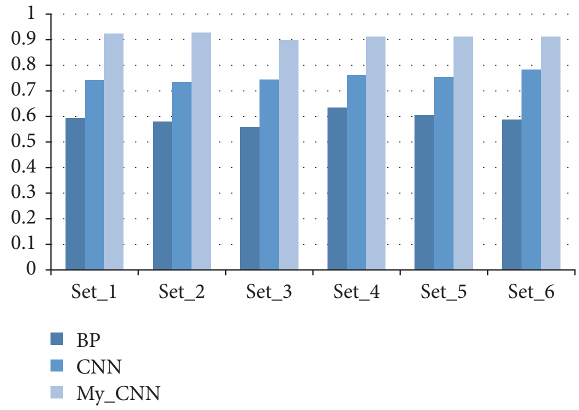

FIGURE 11: Different F_1 scores of the three models on various sample sets.

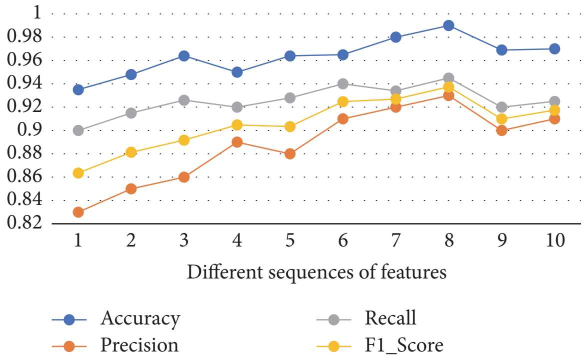

FIGURE 12: Different effects of the model with different input features sequencing.

variables in the data preprocessing part. We use the raw 8-dimensional feature as input, which saves the time for model construction and solves the problem that low-latitude data is not conducive to building a network fraud detection model.

5. Conclusion and Discussion

The CNN model based on feature rearrangement constructed in this paper has an excellent experimental, performance with a good stability. The model needs neither high dimensional input features nor derivative variables and can find a relatively good ordered arrangement of input within a certain number of times. Compared with most existing CNN model, this model saves much calculation time of the derived variables, which makes the design and adjustment process of the model quick and easy. And there is a higher level of availability in an environment where online transactions require rapid response and accurate identification.

In the future work, we will pay more attention to the discovery of sequence characteristics of transactions. In addition, we will apply the LSTM algorithm to make our model have a good memory of the trader's behavior in order to discover more fraudulent transactions accurately. For different sequences of features, the model has different effects. And from this point, we will continue to discover the relationships of data characteristics and find out the characteristic combinations that have an important influence on the model by controlling and transforming the size of the convolution kernel.

Data Availability

The data support for this study is derived from a bank's internal data and will not be provided to those who have not signed a confidentiality agreement with the bank. So, these data are not open.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (nos. 61472004 and 61602109), Shanghai Science and Technology Innovation Action Plan Project (no. 16511100903), and Key Laboratory of Embedded System and Service Computing of Tongji University of Ministry Education (2015).

References

[1] S. Maes, K. Tuyls, B. Vanschoenwinkel, and B. Manderick, “Credit Card Fraud Detection Using Bayesian and Neural Networks,” in *Proceedings of the 1st International Naiso Congress on Neuro Fuzzy Technologies*, pp. 261–270, 2002.

[2] K. Fu, D. Cheng, Y. Tu, and L. Zhang, “Credit Card Fraud Detection Using Convolutional Neural Networks,” in *Neural Information Processing*, vol. 9949 of *Lecture Notes in Computer Science*, pp. 483–490, Springer International Publishing, 2016.

[3] S. Bhattacharyya, S. Jha, K. Tharakunnel, and J. C. Westland, “Data mining for credit card fraud: a comparative study,” *Decision Support Systems*, vol. 50, no. 3, pp. 602–613, 2011.

[4] N. Carneiro, G. Figueira, and M. Costa, “A data mining based system for credit-card fraud detection in e-tail,” *Decision Support Systems*, vol. 95, pp. 91–101, 2017.

[5] W. Yin, K. Kann, M. Yu, and H. Schtze, “Comparative Study of Cnn and Rnn for Natural Language Processing,” 2017, https://arxiv.org/abs/1702.01923.

[6] R. Brause, T. Langsdorf, and M. Hepp, “Neural data mining for credit card fraud detection,” in *Proceedings of the 11th IEEE International Conference on Tools with Artificial Intelligence (ICTAI ’99)*, pp. 103–106, November 1999.

[7] H. Shin, H. R. Roth, M. Gao et al., “Deep convolutional neural networks for computer-aided detection: CNN architectures, dataset characteristics and transfer learning,” *IEEE Transactions on Medical Imaging*, vol. 35, no. 5, pp. 1285–1298, 2016.

[8] R. Gurusamy and V. Subramaniam, “A machine learning approach for MRI brain tumor classification,” *Computers, Materials and Continua*, vol. 53, no. 2, pp. 91–109, 2017.

[9] C. Yuan, X. Li, Q. M. J. Wu, J. Li, and X. Sun, “Fingerprint Liveness Detection from Different Fingerprint Materials Using Convolutional Neural Network and Principal Component Analysis,” *Computers Materials & Continua*, vol. 53, no. 4, pp. 357–372, 2017.

[10] Z. Pan, J. Lei, Y. Zhang, and F. L. Wang, “Adaptive Fractional-Pixel Motion Estimation Skipped Algorithm for Efficient HEVC Motion Estimation,” *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)*, vol. 14, no. 1, pp. 1–19, 2018.

[11] A. Correa Bahnsen, D. Aouada, A. Stojanovic, and B. Ottersten, “Feature engineering strategies for credit card fraud detection,” *Expert Systems with Applications*, vol. 51, pp. 134–142, 2016.

[12] S. Ghosh and D. L. Reilly, “Credit card fraud detection with a neural-network,” in *Proceedings of the 27th Hawaii International Conference on System Sciences*, vol. 3, pp. 621–630, Wailea, Hawaii, USA, 1994.

[13] V. Hanagandi, A. Dhar, and K. Buescher, “Density-based clustering and radial basis function modeling to generate credit card fraud scores,” in *Proceedings of the IEEE/IAFE 1996 Conference on Computational Intelligence for Financial Engineering, CIFEr*, pp. 247–251, March 1996.

[14] R. Patidar and L. Sharma, “Credit Card Fraud Detection Using Neural Network,” in *International Journal of Soft Computing and Engineering (IJSCE)*, vol. 1, pp. 32–38, Citeseer Press, 2011.

[15] A. I. Kokkinaki, “On atypical database transactions: Identification of probable frauds using machine learning for user profiling,” in *Proceedings of the 1997 IEEE Knowledge & Data Engineering Exchange Workshop, KDEX*, pp. 107–113, November 1997.

[16] Z. Zhang, L. Ge, P. Wang, and X. Zhou, “Behavior Reconstruction Models for Large-scale Network Service Systems,” *Peer-to-Peer Networking and Applications*.

[17] A. Gupta, D. Kumar, and A. Barve, “Hidden Markov Model based Credit Card Fraud Detection System with Time Stamp and IP Address,” *International Journal of Computer Applications*, vol. 166, no. 5, pp. 33–37, 2017.

[18] A. Srivastava, A. Kundu, S. Sural, and A. K. Majumdar, “Credit card fraud detection using hidden Markov model,” *IEEE Transactions on Dependable and Secure Computing*, vol. 5, no. 1, pp. 37–48, 2008.

[19] M. F. Zeager, A. Sridhar, N. Fogal, S. Adams, D. E. Brown, and P. A. Beling, “Adversarial learning in credit card fraud detection,” in *Proceedings of the 2017 Systems and Information Engineering Design Symposium, (SIEDS ’17)*, pp. 112–116, IEEE Press, USA.

[20] S. J. Chen and H. Yuan, “Research on the Common Characteristics of Online Transaction Fraud,” in *Journal of Chongqing University of Posts and Telecommunications*, vol. 27, pp. 96–102, 5 edition, 2015.

Journal of Engineering
The Scientific World Journal
International Journal of Rotating Machinery
Journal of Sensors
Advances in Multimedia
Advances in Civil Engineering
Hindawi
Submit your manuscripts at
www.hindawi.com
Journal of Control Science and Engineering
Journal of Robotics
Journal of Electrical and Computer Engineering
Advances in OptoElectronics
VLSI Design
International Journal of Navigation and Observation
Modelling & Simulation in Engineering
International Journal of Aerospace Engineering
International Journal of Chemical Engineering
International Journal of Antennas and Propagation
Active and Passive Electronic Components
Shock and Vibration
Advances in Acoustics and Vibration