Detecting Credit Card Fraud by Generative Adversarial Networks and Multi-head Attention Neural Networks

Zhaorui Meng ⋆, Yanqi Xie, Jinhua Sun

***Abstract*—Credit cards have become a very important method of consumption in modern life. With the rapid development of the credit card industry, the frequency of credit card fraud is gradually increasing as well. Though credit card transaction data is considerably uneven, fraud transaction data is far less uneven than that of normal transactions. Therefore, this paper proposes a credit card fraud detection method based on Generative Adversarial Networks (GAN) and the attention mechanism. Firstly, medical Generative Adversarial Networks (medGAN) is used to generate samples of the minority class to overcome the problems of noise expansion and over fitting when Synthetic Minority Oversampling Technique (SMOTE) and Adaptive Synthetic Sampling (ADASYN) are used to generate new samples. Secondly, a multi-head attention mechanism of deep learning is applied to credit card fraud detection, which expands the application scope of deep learning technology and improves the model prediction performance. Experiments show that the proposed framework can better improve the prediction effect of credit card datasets with extremely unbalanced data by integrating Area Under the ROC Curve (AUC), Balanced Error Rate (BER), and Geometric Mean of Accuracy Rates (G-mean).**

Index Terms—Fraud detection; Generative adversarial networks; Attention mechanism; Imbalanced classification

I. INTRODUCTION

With the rapid development of e-commerce, credit card consumption has become an important method of consumption in modern society. However, the wide use of credit cards also means increasingly more credit card fraud cases. Recent research has shown that the global economic loss caused by credit card fraud in 2018 was as high as US $27.85 billion, an increase of 16.2% over 2017. Moreover, the upward trend is accelerating [1]. Therefore, finding an effective credit card fraud detection method is an urgent matter. In credit card transaction data, the amount of normal data is much larger than that of fraud data, so credit card fraud detection mainly includes two aspects: how to deal with the imbalanced data effectively and how to establish an effective prediction model.

For imbalanced data, the undersampling and random oversampling are often used to improve the prediction performance. Undersampling method can effectively improve the classification performance for the minority class, but this method is unable to make full use of the existing information and will lead to information loss caused by discarding useful data. Random oversampling easily causes over fitting [2]. As an improvement to random oversampling, Chawla et al. proposed the SMOTE [3]. He et al. proposed ADASYN [4], and Menardi et al. proposed Random Over-Sampling Examples (ROSE) [5]. However, these traditional oversampling methods can only generate data of the minority class based on the information contained in the current minority class, which lacks data diversity and will cause over fitting to a certain extent.

Researchers used to use traditional machine learning methods to detect credit card fraud, for instance: association rules [6], random forest [7], and support vector machine [8]. Because deep learning has been well applied in many fields, many researchers have begun to use neural networks to improve the performance of credit card fraud detection. Jurgovsky et al. proposed a long short-term memory network (LSTM)-based model for credit card fraud detection [9]. The experiment results showed that the LSTM did improve fraud detection accuracy. Kang et al. proposed a Convolutional Neural Network (CNN) based fraud detection framework that could capture the inherent patterns of fraudulent behavior [10]. Huang et al. proposed a model that combined classic deep learning techniques with variational automatic coding (VAE) to detect credit card fraud [11].

GAN, a deep learning technique that learns the hierarchies of concepts by building multiple layers of abstraction, has achieved very good results in generating real-looking images [12]. In recent years, a variety of GAN applications in different fields have also bloomed, for instance: image and video generation [13], translation between image and text [14], and dialogue generation [15].

Therefore, this paper uses GAN to generate samples of a minority class to balance the data, and uses a deep learning network based on the attention mechanism to improve the effect of the classification algorithm. The main contributions of our work are as follows:

Manuscript received April 14, 2022; Revised February 9, 2023.

Zhaorui Meng is a senior engineer in School of Computer and Information Engineering, Xiamen University of Technology, Xiamen, Fujian,361024, China (Corresponding author; Email: mengzhaorui@ xmut.edu.cn).

Yanqi Xie is a lecturer in School of Computer and Information Engineering, Xiamen University of Technology, Xiamen, Fujian,361024, China (Email: yqxie@ xmut.edu.cn).

Jinhua Sun is an associate professor in School of Computer and Information Engineering, Xiamen University of Technology, Xiamen, Fujian,361024, China (Email: jhsun@ xmut.edu.cn).

1) We proposed a credit card fraud detection framework based on GAN and the attention mechanism.

2) Using medGAN to generate sample data of a minority class can overcome the problem of losing data information in the undersampling process and the problem of expanding noise and over fitting when generating new samples using the oversampling techniques.

3) Applying the deep learning model based on a multi-head attention mechanism in credit card fraud detection expands the application scope of deep learning technology. The experimental results show that the proposed model outperforms the other 6 compared methods, especially in terms of G-mean and BER values.

The structure of this paper is as follows: Section 2 presents the related work; Section 3 explains our model in detail. In Section 4, we discuss related experimental content. Finally, Section 5 concludes the paper.

II. Related work

2.1 GAN

The basic idea of GAN comes from the two-person zero sum game in game theory [12]. It includes a generator and a discriminator, and learns from the confrontation between the generator and discriminator. In the process of model training, the generator will try its best to generate the same artificial data as the source data, which makes the discriminator unable to accurately determine which data is the real sample data and which data is generated by the generator. The discriminator, as far as possible, will accurately determine which of the input data is the real data and which is the data generated by the generator. In order to win in such a two-person zero sum game, the generator will continue to improve the ability to generate samples while the discriminator will continue to improve the ability to judge whether the samples are true or false. The ultimate goal is to achieve the Nash equilibrium between the generator and the discriminator.

The main shortcomings of the original GAN are the disappearance of the gradient, the instability of the training gradient, the lack of diversity caused by the imbalance of punishment, the difficulty in judging the convergence, and the difficulty in evaluating the quality of the generated data. Therefore, researchers have put forward a variety of improvement methods.

Wasserstein GAN (WGAN) [16], proposed by Arijovsky et al., uses the Wasserstein-1 distance to measure distances between two distributions. The experiments show that the loss is more correlated with sample quality, as the Wasserstein-1 distance is continuous and differentiable almost everywhere.

In addition, Mirza et al. proposed the conditional generative adversarial networks (CGAN) [17]. In this model, the researchers added some restrictions to the generator model and discriminator model. The restriction can be a label or data of different modes. Then, the condition generation network can be built by inputting data and restriction conditions into the model, which can allow the CGAN model to achieve the convergence condition quickly. Further, the model is not easy to collapse.

With the wide application of GAN in natural language processing, text modeling, image generation and machine translation, researchers have begun to study the application of GAN in credit card fraud. Fiore et. al. proposed an approach using GAN to generate mimicked fraudulent samples [18]. The experiments show that the classifier trained on the original data combined with generated fraudulent data has better performance than the classifier trained on the original data only. Chen et al. proposed a credit card fraud detection framework using a sparse autoencoder and GAN. The experiments showed that their proposed model outperformed the One-Class GP and SVDD, which are the state-of-the-art one-class methods [19].

2.2 Attention mechanism

The attention mechanism aims to imitate human visual attention mechanism, which can strengthen the learning ability and generalization ability of network models and cooperate with neural network to solve the weak interpretability of CNN and Recurrent Neural Network (RNN). In recent years, the multi-head attention mechanism has been widely used in speech recognition, text translation, and other fields. It not only has the advantages of self-attention, but it also has the advantages of fast parallel computing and multiple query information, for instance, stacked convolutional layers [20] and multi-layer perceptron (MLPs) [21]. With the application of the attention mechanism, the learning performance of deep learning models in the field of table data have been greatly improved. Arik et al. proposed a deep tabular data learning architecture based on sequential attention, called TabNet[22]. Their experiments demonstrated that the TabNet model outperforms other tabular learning models on various datasets. Yang et al. presented an end-to-end neural network based on the attention mechanism to predict the number of subway passengers [23]. The relationship between the variable length flow table and the subway station is expressed as a constant length vector by the attention mechanism. A large number of experiments showed that this method has good performance in capturing potential dependencies.

III. Credit card fraud detection framework based on medGAN and Multi-head attention mechanism

3.1 Credit card fraud detection procedure

Credit card fraud detection can be taken as a traditional binary classification problem with imbalanced data. Generally speaking, fraud data only accounts for a very small part of the total data, less than 0.1% in some datasets. Therefore, the learning objective is to find a minority class amongst a large amount of data. The proposed detection procedure is illustrated in Fig. 1. The goal of the proposed framework is to increase the learning effect of fraud data through the synthetic data augmented by medGAN and the multi-head attention mechanism.

3.2 GAN

GAN includes two neural networks: a generation model (G) and a discrimination model (D). The generation model generates data based on noise space Z, and the discrimination model judges whether the data is real or generated by the generation model. The training objective of G is to make the generated data close to the distribution of the real data. The discriminator is trained to recognize the real data and the generated data. These two networks iterate and optimize each other, so that the performance of D and G are continuously enhanced. Finally, the two networks reach a dynamic equilibrium. The probability that the data generated by the discriminant model is true is close to 0.5. At this time, the data generated by the generator is approximate to the real data.

The generator and discriminator compete in a two-player minmax game with value function:

$$\min_{G}\max_{D} V(D,G) = \mathop{E}_{x\sim pdata}[\log D(x)] + \mathop{E}_{z\sim pz(z)}[\log(1 - D(G(z)))] \quad (1)$$

The values x and z are samples from real data and noise data, respectively.

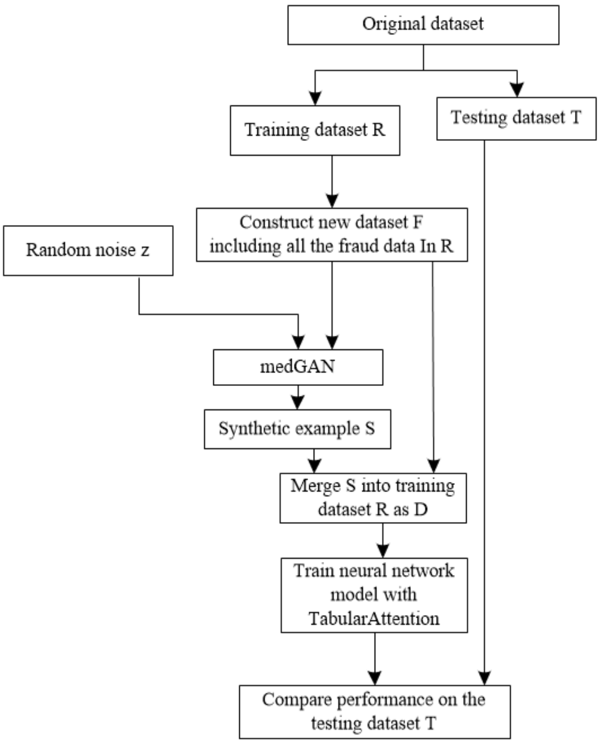

Fig. 1. Flow chart of credit card fraud detection

3.3 medGAN

GAN demonstrates the ability to produce high quality synthetic images. However, it's easy to lead a vanishing gradient problem and mode collapse while using a traditional GAN to generate tabular data because continuous data in tabular data does not conform to Gaussian distribution and distributions of discrete data are often non-differentiable. medGAN proposed by Choi et al., can be used to generate high-dimensional multi-label categorical variables and numerical variables [24].

For medGAN, an autoencoder is implemented between the generator and the discriminator. In the main training phase, the gradient flows from the discriminator to the decoder and then to the generator, which enables end-to-end fine-tuning. Fig. 2 is the architecture of medGAN. As Fig. 2 shows that an autoencoder consists of an encoder and a decoder that decompress $Enc(x)$ to Dec(Enc(x)) as the reconstruction of the original input x. The autoencoder is used to minimize the reconstruction error:

$$\frac{1}{m}\sum_{i=0}^{m} ||x_i - x_i^{'}||_2^2 \quad (2)$$

$$L_{rec} = \sum_{i=1}^{m} x_i \log(\tilde{x}_i) + (1 - x_i)\log(1 - \tilde{x}_i) \quad (3)$$

where $\tilde{x}_i = Dec(Enc(x_i))$.

The traditional GAN loss is modified as follows:

$$L_d = \frac{1}{m}\sum_{i=1}^{m} \log(D(x_i) + \log(1 - D(Dec(G(z_i))))) \quad (4)$$

$$L_g = \frac{1}{m}\sum_{i=1}^{m} \log(D(Dec(G(z_i)))) \quad (5)$$

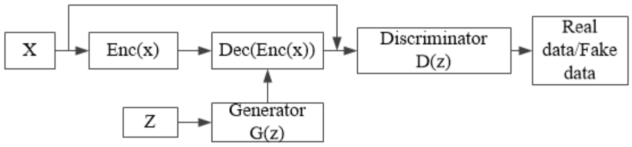

Fig. 2. Architecture of medGAN

3.4 TabularAttention

The goal of TabularAttention is to map the original high-dimensional feature vector to the low dimensional space, while using the attention mechanism to improve the learning ability of the model. Fig. 3 is the architecture of the proposed model. At first, input feature vector X is feed into an embedding layer which projects all input features to the same low-dimensional space. Next, all embedded fields are passed into a multi-head attention layer, which computes the attention of the linear projection vectors in parallel, and then concatenates the context vectors from all the heads and feeds them into the output layer. The output layer simply uses non-linear projection to predict the possibility of credit card fraud.

3.4.1 Embedding layer

In the embedding layer, categorical features and numerical features are transformed from high-dimensional spaces into low-dimensional spaces separately. The categorical feature is represented as follows:

$$e_i = \frac{1}{q} V_i x_i \quad (6)$$

where q is the number of samples in the i-th field, V_i is an embedding matrix for field i, and x_i is a multi-hot vector.

The numerical feature is represented as follows:

$$e_m = v_m x_m \quad (7)$$

where x_m is a scalar value, and v_m is an embedding vector for field m.

A combination of multiple embedding vectors is the output of the embedding layer.

3.4.2 Attention layer

As Fig. 4 shows, the attention model can be understood as a mapping from a query to a series of key value pairs. The source is the input sequence, which is composed of a series of key-values. At the same time, there is a query vector in the network. The essence of the attention mechanism is to compute the similarity between the query and each key using a similarity function, get the weight of each key, and finally sum the weight with the corresponding value. The formula of the attention model is shown in equation (8).

$$Attention(Query, Source) = \sum_{i=1}^{L} Similarity(Query, Key_i) \times Value_i \quad (8)$$

where L is the length of the input sequence, and the function similarity is the method to calculate the similarity between the query and each key.

The multi-head attention structure adopted in this paper is a kind of attention model. Its query, key, and value are obtained by linear transformation of the input sequence. By calculating the weight of each feature and the other input sequence features, the correlation degree of any two features in the sequence is obtained, and the dependence degree between the features in the input sequence is captured.

Denoting that the input sequence of multi-head attention model is X, the multi-head attention model process is as follows:

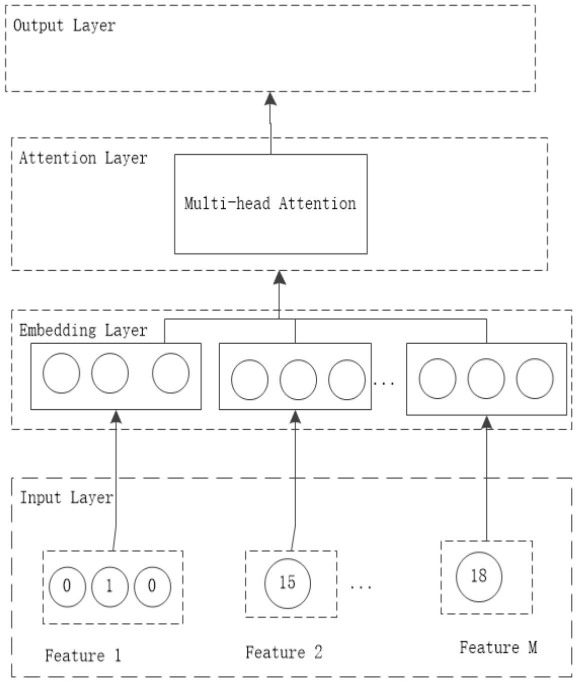

Fig. 3. Architecture of TabularAttention

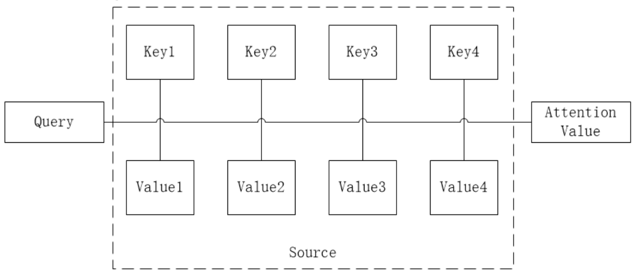

Fig. 4. Attention mechanism

1) The input sequence X is multiplied by different weight matrices W^Q, W^K, and W^V to get Q(Query), K(Key), and V(Value), respectively:

$$\begin{cases} Q = XW^Q \\ K = XW^K \\ V = XW^V \end{cases} \tag{9}$$

2) Using the scaled dot product model as the attention function, the output of a single head attention is obtained, as shown in equation (10).

$$S(Q,K,V) = soft\,max(\frac{QK^T}{\sqrt{d}})V \tag{10}$$

where, $\sqrt{d}$ is the square root of the dimension of vector K.

3) In order to make the model learn different information of the input sequence in different representations of subspaces, we used multiple headers to obtain different information of the sequence. Multi-head attention concatenates a matrix of the calculation results of a single head attention. The multi-head attention is calculated as follows:

$$M = Concat(Head_1, \ldots, Head_h)W^O \tag{11}$$

$$Head_i = S(Q,K,V) \tag{12}$$

where M is the output of multi-head attention, h is the number of heads, $Head_i$ is the output of every single attention, and W^O is the parameter matrix.

Finally, a residual connection layer is added to improve the performance of the model.

$$f_m^{Res} = Re\,L\,U(M + W_{Res}M) \tag{13}$$

where W_{Res} is the parameter matrix and $Re\,L\,U(z) = max(0, z)$ is a non-linear activation function.

Using the attention layer, the representation of each feature f_m will be transformed into a new representation of high-order features f_m^{Res} .

3.4.3 Output layer

The final credit card fraud prediction can be obtained by concatenating all of the output feature vectors and then applying a non-linear project, as follows:

$$y = \sigma(w^T Concat(f_1^{Res}, f_2^{Res}, \ldots, f_M^{Res}) + b) \tag{14}$$

where w^T is a column project vector, b is the bias, and $\sigma(x) = \frac{1}{(1+e^{-x})}$ transforms the values to credit card fraud probabilities.

IV. EXPERIMENTS AND ANALYSIS

4.1 Experimental data

Because of data privacy issues, banks usually do not disclose credit card fraud data. In this experiment, a public credit card fraud dataset from Kaggle was used. The dataset contains 284,807 transaction records of European credit card holders from September 2013, 492 of which are credit card fraud records, accounting for 0.172% of the total. Therefore, the data distribution of fraud records among the total number of records is extremely unbalanced. During the experiment, the data set was divided into two parts: 80% as training set and the remaining 20% as a test set. As a result, there were 394(0.173%, out of 227,845) fraud data entries in the training dataset and 98(0.173%, out of 56,961) fraud data in the test dataset.

4.2 Evaluation method

At present, most of the existing machine learning classification algorithms perform well in terms of prediction accuracy, but they ignore the prediction of minority classes. For example, for the prediction of credit card fraud, the classification accuracy reaches 99% and does not identify credit card fraud data, so this classifier is meaningless. Therefore, the selection of evaluation criteria is very important for the evaluation of a credit card fraud detection algorithm.

In this paper, the performance of the algorithm on the imbalanced dataset was evaluated according to the AUC, BER, and G-mean.

By adjusting the classification threshold i, different confusion matrices C(i) can be obtained. The true positive rate and false positive rate corresponding to each threshold i were calculated. When drawing the Receiver Operating Characteristic (ROC) curve, the false positive rate was taken as the abscissa and the true positive rate as the ordinate. The ROC curve reflects the relationship between the recognition ability of the positive examples and the recognition ability of the negative examples. The area under the ROC curve is often used as the performance evaluation criteria of the classifier.

G-mean is the geometric mean of the accuracy rate of the classifier for each class, which comprehensively evaluates the classification performance of each class. The calculation formula is shown in equation (15). If the classifier divides all samples into one class, then the G-mean value is 0.

Table I Comparison of G-mean values as the number N_g of generated examples is varied

N_g	GAN-TabularAttention	TabularAttention-SMOTE	TabularAttention-ADASYN	TabularAttention	RandomForest	RandomForest-SMOTE	RandomForest-ADASYN
0	0.8969	0.8747	0.8688	0.8629	0.8390	0.8571	0.8571
402	0.8969	0.8747	0.8688	N/A	N/A	0.8571	0.8571
811	0.8916	0.8747	0.8688	N/A	N/A	0.8571	0.8571
1630	0.8919	0.8862	0.8748	N/A	N/A	0.8571	0.8689
4155	0.8918	0.8917	0.8746	N/A	N/A	0.8806	0.8747
8704	0.8889	0.8845	0.8879	N/A	N/A	0.8689	0.8747
67841	0.8969	0.8917	0.8919	N/A	N/A	0.8629	0.8570
113331	0.9029	0.8969	0.8916	N/A	N/A	0.8747	0.8629
227451	0.9138	0.8859	0.8926	N/A	N/A	0.8747	0.8688

Table II Comparison of BER values as the number N_g of generated examples is varied

N_g	GAN-TabularAttention	TabularAttention-SMOTE	TabularAttention-ADASYN	TabularAttention	RandomForest	RandomForest-SMOTE	RandomForest-ADASYN
0	0.0975	0.1175	0.1226	0.1277	0.1481	0.1327	0.1327
402	0.0979	0.1175	0.1226	N/A	N/A	0.1327	0.1327
811	0.1026	0.1175	0.1226	N/A	N/A	0.1327	0.1327
1630	0.1076	0.1073	0.1176	N/A	N/A	0.1327	0.1225
4155	0.1015	0.1025	0.1175	N/A	N/A	0.1123	0.1175
8704	0.1010	0.1057	0.1022	N/A	N/A	0.1226	0.1175
67841	0.0979	0.1025	0.1089	N/A	N/A	0.1277	0.1328
113331	0.0925	0.0979	0.1027	N/A	N/A	0.1175	0.1274
227451	0.0826	0.1076	0.1280	N/A	N/A	0.1175	0.1226

Table III Comparison of AUC values as the number N_g of generated examples is varied

N_g	GAN-Attention	Attention-Smote	Attention-ADASYN	Attention Only	RandomForest	RandomForest-Smote	RandomForest-ADASYN
0	0.8774	0.8825	0.9020	0.8723	0.8520	0.8673	0.8673
402	0.8774	0.8825	0.9020	N/A	N/A	0.8673	0.8673
811	0.8774	0.8825	0.8974	N/A	N/A	0.8673	0.8673
1630	0.8824	0.8926	0.8923	N/A	N/A	0.8672	0.8774
4155	0.8824	0.8975	0.8975	N/A	N/A	0.8877	0.8825
8704	0.8977	0.8923	0.8873	N/A	N/A	0.8775	0.8825
67841	0.8975	0.8975	0.9021	N/A	N/A	0.8723	0.8672
113331	0.8973	0.9075	0.9021	N/A	N/A	0.8825	0.8723
227451	0.8919	0.8923	0.9174	N/A	N/A	0.8825	0.8773

$$G_{-mean} = \sqrt{\frac{TN}{TN+FP} \times \frac{TP}{TP+FN}} \quad (15)$$

BER is the arithmetic mean of the error rate of the classifier for each class of samples. BER treats all kinds of error rates equally and improves the influence of minority classes in imbalanced classification into some extent. The calculation formula of BER is shown in equation (16):

$$BER = \frac{1}{2}\left(\frac{FP}{TN+FP} + \frac{FN}{TP+FN}\right) \quad (16)$$

The research of 12 common performance evaluation criteria of classifiers shows that G-mean and BER are more suitable for the performance evaluation of imbalanced classifiers, because they are more sensitive to the classification results of minority samples and can better reflect the classification ability of the classification algorithm for minority samples [25].

4.3 Experimental results and analysis

This paper used TabularAttention, TabularAttention-SMOTE, TabularAttention-ADASYN, RandomForest, RandomForest-SMOTE, and RandomForest-ADASYN as the comparison algorithms. TabularAttention-SMOTE and RandomForest-SMOTE use the SMOTE oversampling method. TabularAttention-ADASYN and RandomForest-ADASYN use the ADASYN oversampling method.

The medGAN network parameters were set as follows: both the encoder and decoder are single layer feedforward networks. Both the generator G and the discriminator D are feedforward network with two hidden layers, each of which has 128 dimensions.

N_g represents the number of examples generated by medGAN. The ratio of the generated examples to the original data is 0.35%,0.53%,0.89%,2%,4%,30%,50%, and 100%, respectively. medGAN was trained for 1,000 epochs and the minibatch was set to 1,000 records.

The results of the G-mean value of different algorithms are shown in Table I. The higher the G-mean value, the better the performance of the algorithm on the imbalanced data set. As Table I shows, the methods based on RandomForest have the lowest G-mean values. The G-mean value of TabularAttention is higher than the methods based on RadomForest, which means TabularAttention does have a better performance on imbalanced data sets by applying the attention mechanism. G-mean values of TabularAttention-SMOTE and TabularAttention-ADASYN are higher than TabularAttention, which means the two oversampling methods SMOTE and ADASYN are still effective for the attention mechanism-based algorithm. Among all 7 algorithms, the G-mean values of GAN-TabularAttention are the highest, regardless of the value of N_g .

The results of the BER value of different algorithms are shown in Table II. The lower the BER value, the better the algorithm. The experimental results are similar to the above G-mean experiments. The BER value of the three methods based on RandomForest are the highest. The BER value of TabularAttention is lower than the above three methods. The oversampling method SMOTE and ADASYN can improve algorithm performance in terms of BER value. Among all 7 algorithms, the BER values of GAN-TabularAttention are the highest.

The results of the AUC value of different algorithms are shown in Table III. The higher the AUC value, the better the algorithm. Among the TabularAttention based methods and RandomForest based methods, TabularAttention based methods have higher AUC values. However, the experimental results of the four TabularAttention based methods have no strong tendency. Overall, the AUC value of TabularAttention is the lowest. Attention-ADASYN outperformed the other 6 methods in 8 out of 9 experiments. This shows that AUC value cannot reflect the performance of an imbalanced classification algorithm, so it is necessary to evaluate the imbalanced classification algorithm with other performance indicators. In general, GAN-TabularAttention has a higher AUC value, the highest g-mean value and the lowest BER value, so it is better than the other algorithms in an imbalanced classification dataset.

V. Conclusion

In this work, a framework based on medGAN an TabularAttention was proposed for credit card fraud detection, which is an imbalanced data classification problem. The data generated by medGAN can greatly increase the minority class data in imbalanced data sets. At the same time, the use of a multi-head attention mechanism can improve the classification effect. Experimental results showed that the proposed algorithm has better classification performance on the extremely imbalanced credit card dataset, compared with the other 6 comparison algorithms. The experiment also showed that the AUC value is insufficient to evaluate the performance of an imbalanced classification algorithm, and it is more accurate when combining AUC value with other evaluation criteria. Under the joint evaluation of AUC, BER, and G-mean, the proposed algorithm has better imbalanced classification performance than other algorithms.

The next step is to verify the application of this algorithm in other imbalanced data sets, and to explore the application of the new GAN and attention mechanism algorithm in imbalanced data classification problems.

References

[1] Robertson D.The Nilson report, Kirchhain, Germany [Online]. Available: https://nilsonreport.com/mention/407/1link/, Accessed on: April. 18, 2021.

[2] He, H., and E. A. Garcia. "Learning from Imbalanced Data." IEEE Transactions on Knowledge and Data Engineering 21.9(2009):1263-1284.

[3] Chawla, N. V., et al. "SMOTE: Synthetic Minority Over-sampling Technique." Journal of Artificial Intelligence Research, 2002,16(1):321—357

[4] He H, Bai Y, Garcia E A, et al. "ADASYN: adaptive synthetic sampling approach for imbalanced learning." Proceeding of the 2008 International Joint Conference on Neural Networks. Piscataway, NJ: IEEE,2008:1322-1328.

[5] Menardi, G., and N. Torelli. "Training and assessing classification rules with unbalanced data." Data Mining and Knowledge Discovery 28.1(2012):92-122.

[6] Sanchez, D., et al. "Association rules applied to credit card fraud detection." Expert Systems with Applications International Journal 36.2p2(2009):3630-3640.

[7] Xuan, S., et al. "Random forest for credit card fraud detection." 2018 IEEE 15th International Conference on Networking, Sensing and Control (ICNSC) IEEE, 2018.

[8] Sahin, Y., and E. Duman. "Detecting Credit Card Fraud by Decision Trees and Support Vector Machines." lecture notes in engineering & computer science (2011).

[9] Jurgovsky, J., et al. "Sequence Classification for Credit-Card Fraud Detection." Expert Systems with Applications 100. JUN. (2018):234-245.

[10] Kang, F., et al. "Credit Card Fraud Detection Using Convolutional Neural Networks." International Conference on Neural Information Processing Springer, Cham, 2016.

[11] Huang, T., G. Cheng, and K. Huang. "Using Variational Auto Encoding in Credit Card Fraud Detection." IEEE Access PP.99(2020):1-1.

[12] Goodfellow Ian, Pouget-Abadie Jean, Mirza, et. al. Generative Adversarial Nets. In Ghahramani, Z., Welling, M.,Cortes, C., Lawrence, N. D., and Weinberger, K. Q.(eds.), Advances in Neural Information Processing Systems 27, pp. 2672–2680. Curran Associates, Inc., 2014

[13] Chai, L., et al. "CrowdGAN: Identity-free Interactive Crowd Video Generation and Beyond." IEEE Transactions on Pattern Analysis and Machine Intelligence PP.99(2020):1-1.

[14] Zhu, B., and C. W. Ngo. "CookGAN: Causality Based Text-to-Image Synthesis." 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) IEEE, 2020.

[15] Qi, W., et al. "Are You Talking to Me? Reasoned Visual Dialog Generation Through Adversarial Learning." 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) IEEE, 2018.

[16] Arjovsky, Martin, Soumith Chintala, and Léon Bottou. "Wasserstein generative adversarial networks." International conference on machine learning. PMLR, 2017.
[17] Mirza, Mehdi, and Simon Osindero. "Conditional generative adversarial nets." arXiv preprint arXiv:1411.1784 (2014).
[18] Fiore, Ugo, et al. "Using generative adversarial networks for improving classification effectiveness in credit card fraud detection." Information Sciences 479 (2019): 448-455.
[19] Chen, Jian, Yao Shen, and Riaz Ali. "Credit card fraud detection using sparse autoencoder and generative adversarial network." 2018 IEEE 9th Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON). IEEE, 2018.
[20] Shavitt, Ira and Eran Segal. “Regularization Learning Networks: Deep Learning for Tabular Datasets.” Neural Information Processing Systems (2018).
[21] Xu, L., et al. "Modeling Tabular data using Conditional GAN.", 10.48550/arXiv.1907.00503. 2019.
[22] Arik, S. Ö., and T. Pfister. “TabNet: Attentive Interpretable Tabular Learning”. Proceedings of the AAAI Conference on Artificial Intelligence, vol. 35, no. 8, May 2021, pp. 6679-87.
[23] Yang, J., X Dong, and S. Jin. "Metro Passenger Flow Prediction Model Using Attention Based Neural Network." IEEE Access PP.99(2020):1-1.
[24] Choi, E. et al. “Generating Multi-label Discrete Electronic Health Records using Generative Adversarial Networks.” ArXiv.1703.06490 (2017).
[25] Hu, B. G., and W. M. Dong. "A study on cost behaviors of binary classification measures in class-imbalanced problems." Computer Science 8.11(2014): e79774-e79774.