Credit Card Fraud Detection Using Sparse Autoencoder and Generative Adversarial Network

Jian Chen
Department of Computer Science and Engineering
Shanghai Jiao Tong University
Shanghai, China
cs_jerrychen@sjtu.edu.cn

Yao Shen*
Department of Computer Science and Engineering
Shanghai Jiao Tong University
Shanghai, China
yshen@cs.sjtu.edu.cn

Riaz Ali
Department of Computer Science and Engineering
Shanghai Jiao Tong University
Shanghai, China
riazbabuzai@sjtu.edu.cn

***Abstract*—Current credit card detection methods usually utilize the idea of classification, requiring a balanced training dataset which should contain both positive and negative samples. However, we often get highly skewed datasets with very few frauds. In this paper, we want to apply deep learning techniques to help handle this situation. We firstly use sparse autoencoder (SAE) to obtain representations of normal transactions and then train a generative adversarial network (GAN) with these representations. Finally, we combine the SAE and the discriminator of GAN and apply them to detect whether a transaction is genuine or fraud. The experimental results show that our solution outperforms the other state-of-the-art one-class methods.**

Keywords—credit fraud detection, deep learning, sparse autoencoder, generative adversarial network

I. INTRODUCTION

Credit card fraud refers to using forged credit cards to purchase goods without paying, and it's also associated to identity theft, such as using others' credit cards illegally to defraud money or property. Over the last few decades, with the development of online transactions, credit cards begin to be one of the most popular online payment methods. And the continued diversity of frauds increases difficulty in anti-fraud, leading to huge losses for customers and banks. Several main types of credit card frauds are given as follows:

- Application fraud. Fraudulent people often obtain the identity information of other people by stealing their telephone bills, bank bills and so on. And then applying for credit cards with stolen identity information.
- Mail non-receipt fraud. This refers to fraudulent transactions that credit cards are intercepted and activated by others during the delivery.
- Lost or stolen fraud. This means unauthorized or fraudulent use of credit cards lost by the cardholders.
- Card not present fraud. Criminals use credit cards fraudulently by telephone, email or the Internet which don't need to present a real card while just need to give credit card numbers and names.
- Account take-over fraud. Fraudsters obtain some relevant information of cardholders, and then report the loss of credit cards and ask banks to resend new cards to the designated address of the fraudsters for fraudulent transactions.

Therefore, having an efficient fraud detection strategy becomes a main and important direction for helping banks to reduce lots of losses as well as to increase the confidence of customers.

Recently, with the fast development of information technology and the advancement of data mining methods, researchers are committed to applying some data mining methods to this field. From single pattern recognition methods such as neural networks and decision trees to combined methods, they all provide scientific basis for anti-fraud. Although these methods overcome the deficiency of knowledge acquisition in traditional rule-based expert systems. They still have some deficiencies, especially these methods are based on the idea of supervised learning, which needs balanced dataset of both normal transactions and fraudulent transactions. So these methods do not work well in anti-fraud field since fraudulent transactions are much fewer than normal transactions. Therefore, we design a new method which can build an anti-fraud model using only normal transactions.

In contrast with traditional classification methods which are focused on classifying samples of two or more classes, one-class classification methods try to learn a model on samples of one class and distinguish them from samples of the other classes. In our work, we focus on distinguishing fraudulent transactions from normal transactions, but we use normal transactions to train our model to distinguish the other class (fraudulent transactions). The basic idea of our method is stated as follows. Firstly, we map the normal transactions into vector space through SAE [1], and then we train a GAN [2] on the basis of representations of normal transactions, its generator is to generate fake normal transactions and its discriminator is to identify whether a transaction is genuine or fake. Finally, we determine if a new transaction is fraudulent or normal by combining SAE and the discriminator of GAN to predict.

The main contributions of our work are as follows:

- We proposed a model based on SAE and GAN to distinguish fraudulent transactions from normal transactions.
- Our model can be treated as a one-class classification method to some extent, so we don't need a mixed dataset containing both positive samples and negative samples, which help fix imbalanced problem.
- Our model can achieve higher performance than the other state-of-the-art one-class methods according to f1 score.

The rest of paper is structured as follows. Section II presents the related work, Section III introduces Autoencoder (AE) and GAN. Afterward, Section IV explains our model in

978-1-5386-7266-2/18/$31.00 ©2018 IEEE

detail and the evaluation and analysis are shown in Section V. Finally, Section VI concludes the paper.

II. Related Work

A. Credit Card Fraud Detection

Credit card fraud detection has attracted widespread attention from academia. A review about credit card fraud detection techniques is described in [3]. It shows several previous types of fraud detection techniques such as neural networks, decision tree, genetic algorithms and some other outlier detection techniques. In recent years, researchers try to apply some new approaches to this field. Soriano and Vergara [4] presented a new approach for automatic detection of frauds in credit card transactions based on non-linear signal processing and it can be applied to several datasets using parameters derived from key performance indicators of business. Bahnsen, Stojanovic and Aouada [5] proposed a cost sensitive method based on Bayes minimum risk to represent realistically the monetary gains and losses due to fraud detection. Fu, Cheng and Tu [6] captured the essential characteristics of frauds by using a model based on convolutional neural network (CNN). Halvaiee and Akbari [7] addressed credit card fraud detection using Artificial Immune Systems. Vlasselaer, Bravo and Caelen [8] combined intrinsic and network-based features to detect fraudulent credit card transactions conducted in online store. Zareapoor and Shamsolmoali [9] compared various data mining techniques and introduced the bagging classifier based on decision tree.

B. One-class classification

One-class classification methods only use one class of samples for training, they are suitable for some extreme cases where negative samples are unavailable or training datasets are extremely imbalanced. In these cases, imbalanced learning approaches we mentioned above achieve poor performance compared to one-class classification approaches. SVM are widely adopted into classification field, thus some people try to solve one-class classification based on SVM. Tax and Duin [10] put forward a novel approach called Support Vector Data Description (SVDD) which can be treated as One-Class SVM to some extent, but this method is very sensitive to outliers in the training data. Yin, Zhu and Jing [11] proposed a robust one-class SVM which performs better when the training set is corrupted by outliers. Kemmler, Rodner and Denzler [12] proposed novel one-class classification methods based on Gaussian process (GP) priors which outperform the SVDD approach. However, One-Class GP needs to set a threshold to determine a transaction is fraudulent or normal, but the threshold can only be obtained either by expert or by training on a small set of balanced data. Désir, Bernard and Petitjean [13] proposed a method based on a one-class random forest method and an original outlier generation procedure which utilized classifier ensemble randomization principles. In our work, we proposed a new model composed of SAE and GAN, it can help handle the situation where only one class of samples is available and it can tune parameters itself and require no additional manual intervention or training like One-Class GP.

III. Autoencoder And GAN

A. Autoencoder

AE is one type of Feedforward Neural Network, and it is primarily used for dimensionality reduction and feature extraction [14]. Unlike other Feedforward Neural Networks which are concerned with the output layer and error rate, autoencoders are focused on the hidden layer. And AEs typically have only one hidden layer while others are deeper.

The original autoencoder is simple with only one input layer, one hidden layer and one output layer. It has two constraints, the first one is the dimension of the hidden layer should be smaller than the dimension of the input layer. The second one is to minimize the error between input and output since output is the re-formation of input. The structure of original AE is shown in Fig. 1.

We represent the input by a vector composed of the hidden units (here we call it code). We name the part that compress the input into the code as encoder and the part that restore the code to input as decoder. Thus, autoencoder is trying to get the minimum of the value function:

$$\phi, \psi = \text{argmin}_{\phi,\psi} L(X, (\psi \circ \phi)X) \tag{1}$$

Where ϕ is the encoder and ψ is the decoder, and L means the error between input and output which usually use mean-square error method or cross-entropy method. From the value function (1) we can see that autoencoder can be thought as an advanced PCA approach since autoencoder has nonlinear units, leading to more refined code which can represent the input better.

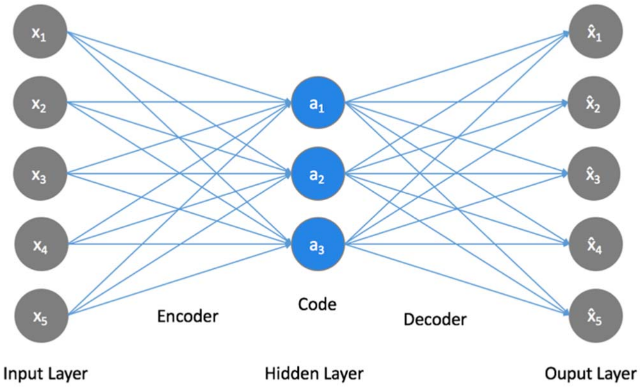

Fig. 1. The Structure of original AE

SAE is just to add sparse constraint to the code of AE. The hidden layer has more neurons than the input layer and output layer and some neurons of it are the same as input, while the others tend to zero. Fig. 2 shows us the structure of SAE. Sparseness has some merits as follows.

- It helps extract key features of input, therefore, model has better anti-noise ability compared to original AE.
- Sparseness is easier to understand and interpret on account of most real-world cases meet the constraint.

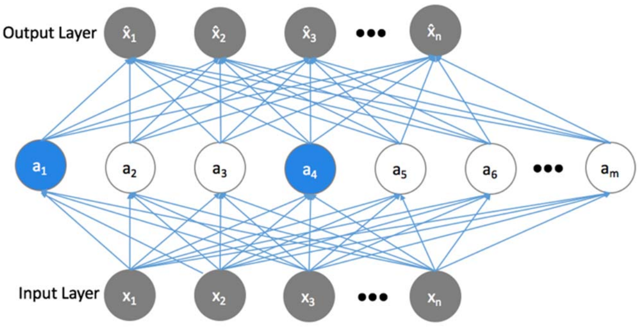

Fig. 2. The Structure of SAE

After adding sparse constraint, we can define the loss function:

$$L_{sparse} = L + \beta \sum_j KL(\rho || \hat{\rho}_j) \quad (2)$$

Where KL means Kullback-Leibler Divergence, KL-divergence is a standard function which is used for measuring how different two distributions are. ρ represents the expected activation level of neurons in the network, and $\hat{\rho}_j$ denotes the average activation level of the jth neuron. β controls the weight of sparsity of penalty. Here the KL divergence can be defined as:

$$KL(\rho || \hat{\rho}_j) = \rho \log \frac{\rho}{\hat{\rho}_j} + (1-\rho) \log \frac{1-\rho}{1-\hat{\rho}_j} \quad (3)$$

$$\hat{\rho}_j = \frac{1}{m} \sum_i [a_j(x^{(i)})] \quad (4)$$

Where $x^{(i)}$ means the i-th sample and $a_j(x^{(i)})$ means the activation of the hidden unit when it's given a specific input x. When $\hat{\rho}_j = \rho$, $KL(\rho || \hat{\rho}_j)$ will be 0, and otherwise it increases monotonically as $\hat{\rho}_j$ diverges from ρ.

B. Generative Adversarial Network

GANs are inspired by the idea of two-player game. In a two- player game, the sum of interests of the two players are zero or a constant, that means if one has a gain, the other must have a loss. For GANs, a generative model and a discriminative model are the two players. Usually both the two models are multilayer neural networks, the generative model learns the distribution of samples, while the discriminative model estimates the probabilities that a sample comes from the original training data or generated data.

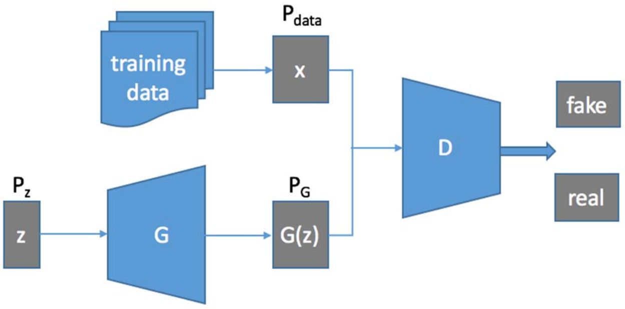

Fig. 3. The Structure of GAN

Fig. 3 shows us the basic structure of GANs. The z is noise which is randomly generated initially, $G(z)$ means that the generator G tries to learn a distribution P_G from the distribution of noise P_z and make P_G as close as possible to the distribution of real data (P_{data}). Discriminator D tries to identify whether the sample is real or not. We constantly adjust G and D until D can't distinguish the real data and the generated data during training, that is to say, we achieve the global optimality of $P_G = P_{data}$. Certainly, we often fail to achieve the global optimality in practice, we just stop when p_G converges to p_{data} within acceptable threshold. G tries to confuse D while D does its best to distinguish generated samples from normal samples in this process, so we can define the objective functions about G and D as follows.

$$\min_G E_{z \sim p_z}[\log(1 - D(G(z)))] \quad (5)$$

$$\max_D E_{x \sim p_{data}}[\log(D(x))] + E_{z \sim p_z}[\log(1 - D(G(z)))] \quad (6)$$

Therefore, GANs can be described as a minimax problem $\min_G \max_D V(G,D)$ with the value function $V(G,D)$:

$$V(G,D) = E_{x \sim p_{data}}[\log(D(x))] + E_{z \sim p_z}\left[\log\left(1 - D(G(z))\right)\right] \quad (7)$$

IV. Proposed Method

A. Feature Selection

There are usually several behaviors of customers in actual credit card transactions such as transfer, deposit, repayment and so on. Therefore, it is hard to find a proper pattern to extract features which are useful for anti-fraud. Deep learning provides us with a new way of thinking, here we try to apply a SAE to do feature extraction which can help separate normal transactions and fraudulent transactions.

We can pass the initial features of normal transactions to the SAE and train it to extract appropriate features. SAE are used to increase the number of features and extract key features with a sparse constraint since there are few features in our dataset. During training, SAE computes the reconstructed sequence of normal transactions and optimizes parameters with the loss function (2).

B. Training

We use the hidden representation of normal transactions obtained in trained SAE as the input of GAN after extracting key features of normal transactions. In order to fool the discriminator of GAN, the generator of GAN takes random noise as input and tries to generate fake normal transactions which are close to the real normal transactions. The discriminator is trained to distinguish whether the example is real or fake. We train the GAN and optimize the parameters of G and D with the loss functions (5) and (6) respectively. The structure of our model is illustrated in Fig. 4, it uses SAE to do feature extraction and input hidden representations of real normal examples to GAN for training the discriminator to distinguish fake normal transactions generated by the generator from normal transactions, after training. After training, the discriminator will have the ability to distinguish fraudulent transactions since they are very different from normal transactions.

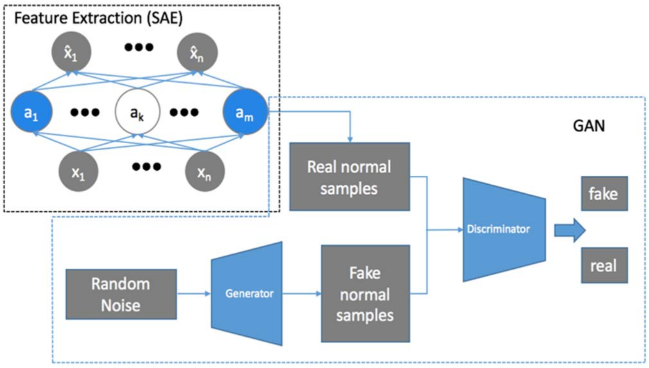

Fig. 4. The Structure of Our Model

C. Testing

We obtain trained SAE and GAN after above phases, and then we combine the SAE and the discriminator of GAN to predict the type of new transactions. Firstly, we input the raw features to SAE and get their hidden representations. Secondly, the discriminator of GAN takes the representations as input and determine whether they are fraudulent transactions or not.

V. Experimental Results

A. Datasets

The Machine Learning Group of ULB (Université Libre de Bruxelles) and Worldline cooperated to collect the dataset for big data mining and fraud detection. This dataset contains two days' transactions of credit cards in European which is composed of 284,315 normal transactions and 492 fraudulent transactions. Owing to the requirement of guaranteeing users' privacy, the original data has been transformed to numerical value by PCA and we don't know their actual meaning. The dataset has 31 features, 28 of them are named from V1 to V28 while the other three are named as Time, Amount and Class respectively. Here, Time means seconds elapsed since the first transaction, Amount means the transactions' amount, and Class is the class of transactions (value 1 represents fraud and 0 otherwise). We use t-SNE [15] to reduce the dimension of raw data to 2 and visualize them in Fig. 5 (1000 normal transactions and 492 fraudulent transactions).

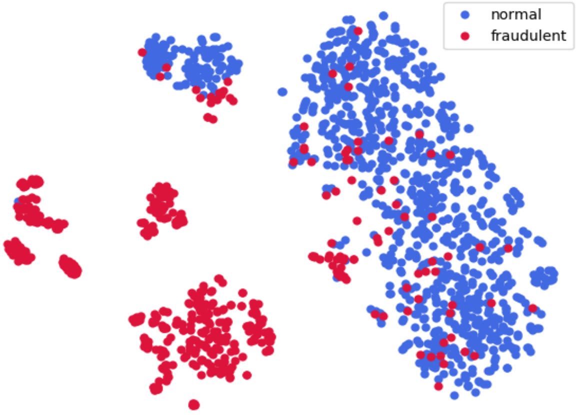

Fig. 5. The Distribution of our dataset in 2D

In our experiments, we randomly select 5000 normal transactions for training SAE and GAN. After training, we use 492 normal transactions and all 492 fraudulent transactions for testing and calculate several indicators such as precision, recall and f1 score to compare the performance between our model and state-of-the-art one-class methods.

B. Indicators

Three indicators (precision, recall and f1 score) are used to evaluate the performance of our model. Precision (also called positive predictive value) is the fraction of true frauds among all samples which are classified as frauds, while recall (also known as sensitivity) is the fraction of frauds which have been classified correctly over the total amount of frauds. And F1 score is a measure that combines precision and recall.

$$\text{Precision} = \frac{TP}{Tp+FP} \tag{8}$$

$$\text{Recall} = \frac{TP}{Tp+FN} \tag{9}$$

$$\frac{1}{F1} = \frac{1}{Precision} + \frac{1}{Recall} \tag{10}$$

TP (True Positive) means the amount of frauds which have been classified correctly. FP (False Positive) means the amount of normal transactions which have been classified as frauds. FN (False Negative) means the amount of frauds which have been classified as normal ones. TN (True Negative) means the amount of normal transactions which have been classified correctly.

C. Results

Three groups of experiments are set up as follows to prove the effectiveness of our model.

- Change the hidden dimension of SAE to see the effect of performance of our model.
- Compare our model to the state-of-the-art one-class methods (SVDD and One-Class GP).
- Compare our model with or without SAE.

The results are illustrated in Fig. 6, Fig. 7 and Fig. 8 respectively. In Fig. 6, precision is much higher than recall since our model is trained on normal transactions, so fraudulent transactions are more likely to be mistaken. And the trend of precision, recall and f1 score of fraud detection are all clearly demonstrated with SAE Hidden dimension changing from 30 to 65, the precision doesn't change much while the recall firstly increases and when dimension comes to 50, it begins to decrease. The situation of precision is based on the same reason why precision is much higher than recall, since the model is less likely to classify normal transactions into fraudulent class. For the recall, it's because of that with the increase of dimension, raw data will be mapped into a higher dimension, thus raw data which is inseparable can be separated now, but when the dimension becomes too large, it leads to information redundancy and our model is overfitting on normal transactions to some extent. The f1 score is mainly influenced by recall since precision changes little. We can notice that when dimension is 35, the precision and the recall seem to violate the overall trend, we think that random sampling may lead to distribution difference of train dataset and test dataset.

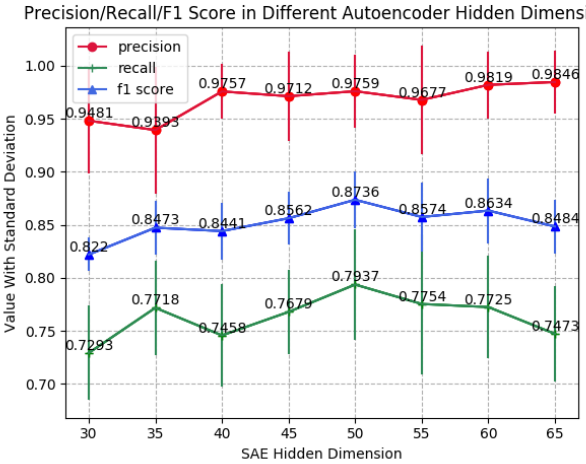

Fig. 6. Precision/Recall/F1 Score of our model in different SAE hidden dimensions

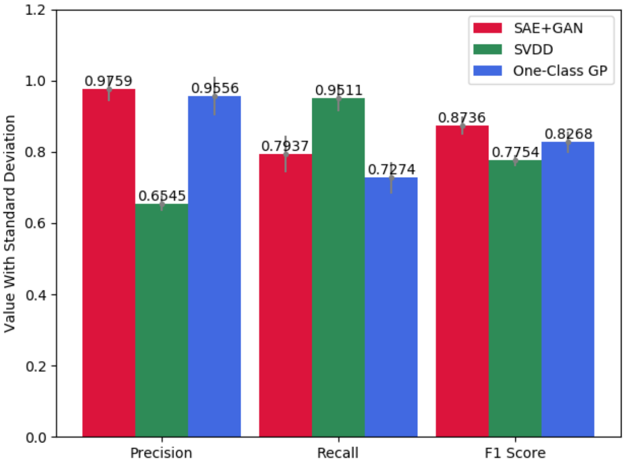

Fig. 7. Precision/Recall/F1 Score of our model compared with the other two state-of-the-art one-class methods

As shown in Fig. 7, our model performs both better in precision and recall than One-Class GP, that's because One-Class GP needs a small portion of both normal transactions and fraudulent transactions (about 50 samples in total) to obtain the threshold which has a significant impact on the result, but we all know that the threshold obtained with so few samples is often not good enough. However, our model learns the distribution of normal transactions and generates fake normal transactions for training the discriminator, therefore it gets a higher performance than One-Class GP. For SVDD, it tries to get a hypersphere with minimum radius to make most normal transactions locate in its inner space but it's sensitive to outliers and this may cause offset of the hypersphere. Thus, some of normal transactions will be classified into the fraudulent class, resulting in decrease of precision. Meanwhile, it will have a relatively high recall since fraudulent transactions are easily located outside of the hypersphere. In general, our model gets a higher f1 score than the other two methods, so experimental results prove that our model performs better than these two methods.

Fig. 8 shows the performance of our model with or without SAE, the result leads to the conclusion that the recall doesn't change much but the precision is much higher using SAE to do feature extraction since SAE can help extract key features and remove useless information, and then it's easier for generator to learn the distribution of normal transactions, resulting in a higher precision. Thus, a higher f1 score will be achieved due to the increase of precision.

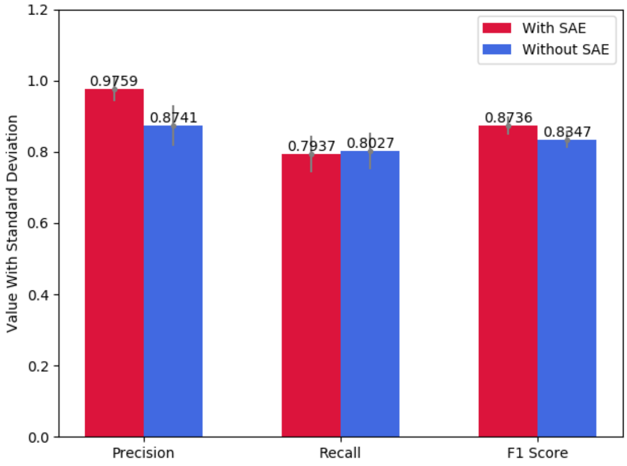

Fig. 8. Precision/Recall/F1 Score of our model with or without SAE

VI. CONCLUSION

In this paper, we propose a new model based on SAE and GAN to detect frauds. Our model requires only normal transactions for training and needs no additional help, so it can handle imbalanced situations well. Experiments conducted on this dataset show that our model outperforms the start-of-the-art one-class methods (SVDD and One-Class GP). And applying the model to other one-class scenarios rather than only credit card fraud detection is our future target. Moreover, we find that the standard deviation of our model is higher than the other two methods during training. The f1 score is illustrated in Fig. 9, in which f1 score is still not stable even when training epochs become large. We will research the reason of this problem and try to find an approach to solve it in the future work.

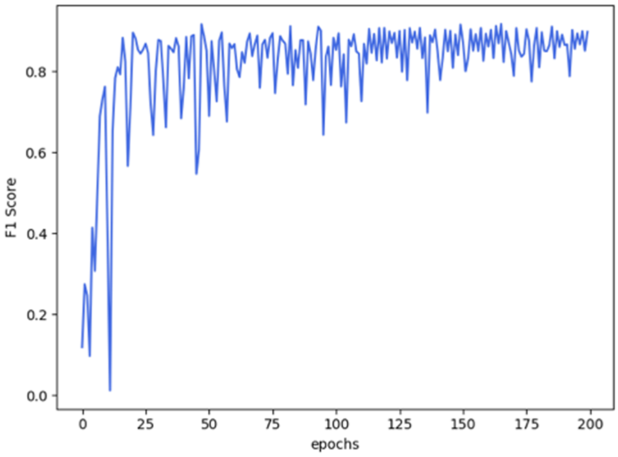

Fig. 9. F1 Score of our model during training

ACKNOWLEDGMENT

The authors thank Wordline and the Machine Learning Group of ULB (Université Libre de Bruxelles) for providing us with the dataset to evaluate our model and prove our conclusion.

REFERENCES

[1] A. Ng, Sparse autoencoder, CS294A Lecture Notes, pp. 72, 2011.

[2] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A.C. Courville, and Y. Bengio. Generative adversarial nets. In Proceedings of NIPS, pages 2672–2680, 2014.

[3] Chaudhary K, Yadav J, Mallick B. A review of fraud detection techniques: Credit card[J]. International Journal of Computer Applications, 2012, 45(1): 39-44.

[4] A. Soriano, L. Vergara, Automatic credit card fraud detection based on non-linear signal processing, Proceedings of the 46th Annual IEEE International Carnahan Conference, Boston, USA, ICCST 2012, Article number 6393560, pp. 207-212, 2012.

[5] Bahnsen A C, Stojanovic A, Aouada D, et al. Cost sensitive credit card fraud detection using Bayes minimum risk[C]. Machine Learning and Applications (ICMLA), 2013 12th International Conference on. IEEE, 2013, 1: 333-338.

[6] K. Fu, D. Cheng, Y. Tu, L. Zhang, Credit Card Fraud Detection Using Convolutional Neural Networks, International Conference on Neural Information Processing, pp. 483-490, 2016.

[7] Halvaiee N S, Akbari M K. A novel model for credit card fraud detection using Artificial Immune Systems[J]. Applied Soft Computing, 2014, 24: 40-49.

[8] Van Vlasselaer V, Bravo C, Caelen O, et al. APATE: A novel approach for automated credit card transaction fraud detection using network-based extensions[J]. Decision Support Systems, 2015, 75: 38-48.

[9] Zareapoor M, Shamsolmoali P. Application of credit card fraud detection: Based on bagging ensemble classifier[J]. Procedia Computer Science, 2015, 48: 679-685.

[10] David M. J. Tax, Robert P. W. Duin, Support Vector Data Description, Machine Learning, v.54 n.1, p.45-66, January 2004.

[11] Yin S, Zhu X, Jing C. Fault detection based on a robust one class support vector machine[J]. Neurocomputing, 2014, 145: 263-268.

[12] M. Kemmler, E. Rodner, J. Denzler, "One-class classification with Gaussian processes", Proc. 10th Asian Conf. Comput. Vision, pp. 489- 500, 2011.

[13] Désir C, Bernard S, Petitjean C, et al. One class random forests[J]. Pattern Recognition, 2013, 46(12): 3490-3506.

[14] P. Vincent, H. Larochelle, Y. Bengio, and P.-A. Manzagol. Extracting and composing robust features with denoising autoencoders. In Proceedings of the 25th International Conferenceon Machine Learning, pages 1096–1103. ACM, 2008.

[15] van der Maaten, L. & Hinton, G. E. Visualizing data using t-SNE. J. Mach. Learn.Research 9, 2579–2605 (2008).