An Efficient Domain-Adaptation Method using GAN for Fraud Detection

Jeonghyun Hwang[1], Kangseok Kim[2]*
Department of Artificial Intelligence and Data Science
Graduate School of Ajou University, Suwon, Korea[1, 2]
Department. of Cyber Security, Ajou University, Suwon, Korea[2]

***Abstract*—In this paper, an efficient domain-adaptation method is proposed for fraud detection. The proposed method employs the discriminative characteristics used in feature maps and generative adversarial networks (GANs), to minimize the deviation that occurs when a common feature is shifted between two domains. To solve class imbalance problem and increase the model's detection accuracy, new data samples are generated by applying a minority class data augmentation method, which uses a GAN. We evaluate the classification performance of the proposed domain-adaption model by comparing it against support vector machine (SVM) and convolutional neural network (CNN) models, using classification performance evaluation indicators. The experimental results indicated that the proposed model is applicable to both test datasets; furthermore, it requires less time for learning. Although the SVM offers a better detection performance than the CNN and proposed domain-adaptation model, its learning time exceeds those of the other two models when dataset increases. Also, although the detection performance of the CNN-based model is similar to that of the proposed domain-adaptation model, its learning process is longer. In addition, although the GAN used to solve the class imbalance problem of the two datasets requires slightly more time than SMOTE (synthetic minority oversampling technique), it shows a better classification performance and is effective for datasets featuring class imbalances.**

Keywords*—Fraud detection; domain adaptation; data augmentation; deep learning; GAN

I. Introduction

With the rapid development of information technology, the existing financial industry paradigm is changing; the new paradigm, following the evolution of smartphones and mobile technologies, is creating new forms of electronic financial services, increasing the number of non-face-to-face transactions (through the use of various devices and communications technologies), and simplifying and diversifying payment methods. However, alongside these developments, concerns over security incidents (e.g., cyber threats involving the leakage and hacking of financial and personal information) are also increasing, owing to the new approaches facilitated by the Internet, device diversity, transaction simplicity, and ease of data flow. Therefore, the performances of fraud detection systems (FDS) must be improved, to actively respond to these diversified and intelligent cyber threats. Accordingly, machine- and deep-learning based technologies, which learn large quantities of data to improve prediction and classification accuracies, have recently been developed; thus, research incorporating these technologies has increased accordingly, to improve the performances of FDSs. However, the existing FDS's abnormal-transaction-detection method which combines machine- and deep-learning techniques to identify abnormal transactions in large quantities of real-time data is time-consuming and computationally expensive. Therefore, this study presents a faster-learning abnormal-transaction-detection model, by training a model suitable for data across different domains and utilizing the common features and information thereby found. The proposed model to detect anomalies between different domains is constructed using domain adaptation method [1] which is one of transfer learning [2], a machine-learning method that utilizes pre-learned domain information from similar domains when a specific task or domain is changed. The datasets employed in the proposed domain-adaptation method are generally used in research relating to abnormal-transaction detection; in particular, they are benchmark datasets for fraud detection in credit card [3] and financial [4] datasets. However, because both datasets feature an unbalanced ratio between the normal transactions and fraudulent or anomalous ones, the classes must be balanced to improve the machine learning performance and ensure smoothly learning. Then, a data augmentation method can be used to increase the total number of data when datasets are insufficient; this method is applied to the minority class using a generative adversarial networks (GANs) [5]; the augmented data are used for training/test data of the proposed domain-adaptation model, and the results are compared with those of SMOTE (Synthetic Minority Oversampling Technique) [6] which is one of oversampling methods. Therefore, in this study, a GAN and SMOTE are used to solve the class imbalance problem for credit-card and financial-transaction fraud datasets; then, the domain-adaptation method is used to implement a model for detecting abnormal transactions in the two datasets; finally, the method's effectiveness is verified through a comparison of its classification performance against those of support vector machine (SVM) [7] and convolutional neural network (CNN) [8] based methods. The remainder of the paper is organized as follows: In Section II, the background and related research are described; in Section III, the model and datasets employed are described in detail; in Section IV, the experimental environment, learning method, and hyperparameters are described; in Section V, the classification performance of the model is compared and analyzed against those of the SVM- and CNN-based models; and in Section VI, the conclusions and limitations of the research are described, and future research directions are considered.

*Corresponding Author.

II. Related Works

This section describes existing fraud detection methods, data augmentation approaches, and domain adaptation methods.

A. Fraud Detection

Abnormal transaction detection is a data mining approach used to detect transactions that differ from normal transaction patterns. The detection results are divided into two transaction classes: normal and abnormal. A variety of detection technologies are constantly being studied to minimize the risks posed to users by fraudulent transactions. Studies for abnormal transaction detection include the development of procedures for classification (a field of supervised learning), clustering (a field of unsupervised learning), deep learning and so on. In the existing research on classification-model-based abnormal transaction detection approaches, [9] proposed *Very Fast Decision Tree*, which can manage unbalanced data using decision trees; [10] employed a hidden Markov model (HMM) to learn a normal credit card transaction, and they classified transactions that were not accepted by the HMM as abnormal; [11] detected abnormal transactions using *k-Nearest Neighbors*, which offers reduced memory consumption compared to other machine learning methods. Furthermore, [12] proposed a model to detect abnormal transactions and money laundering, by applying an SVM. In addition, deep learning models have been applied to abnormal transaction detection using auto-encoders or GANs as a solution for data unbalancing [13, 14]. In addition, a significant number of abnormal detection models have been proposed to increase the accurate detection rate of FDS. In our study, for the classification performance of fraud detection, the proposed domain-adaptation model was evaluated by comparing it with the SVM and CNN models, which are supervised learning-based analytical models.

B. Oversampling

Approaches to solving the data imbalance problem can be divided into four categories: sampling-based, cost-based, kernel-based, and active-learning-based methods [15]. The approach of changing the distribution between the majority and minority classes in unbalanced datasets is a sampling-based method; the distribution balance can be adjusted to reduce the number of data samples in the majority class (undersampling) or to increase the number in the minority class (oversampling). SMOTE is an oversampling method: it generates data between the minority class' data samples by connecting a straight line between them. Majority Weighted Minority Oversampling Technique [16] identifies minority class data and assigns weights according to the Euclidean distance between them and the nearest data samples in the majority class; then, a clustering approach generates data between the weighted minority class data in the same way as SMOTE. Meanwhile, the Random Oversampling Examples (ROSE) [17] method generates new minority data based on the existing kernel-density estimate; robROSE [18] is an oversampling method that overcomes the shortcomings of ROSE (which can deviate under the influence of outliers). Of the above methods, we used SMOTE to solve the class imbalance problem, because it is easier to implement and understand than other methods and offers excellent performance characteristics.

C. Data Augmentation

Data augmentation, which was first introduced in [19], is a popular method for processing image data; it generates noise whilst preserving the amount of information in the data. GANs are suitable models for performing data augmentation; it consists of two artificial neural networks (ANNs) that learn by competing against each other: one is a generator that receives random noise as an input and processes it to resemble the distribution of the original data; the other is a discriminator that distinguishes the original data from those created by the generator. The generator seeks to make the data it produces indistinguishable from the original data as much as possible, and the discriminator tries to classify the two types of data with the highest possible probability, in opposition to the generator. As a result, data that pass through a network consisting of generators and discriminators are generated with a distribution similar to that of the original data. By varying the structures and purposes of GANs, researchers have successfully applied them to various fields; in particular, the field of image-data-related research [5, 20] has found considerable use for them, and models for increasing their performance and generating new image data have been proposed. Among them, deep convolutional GANs [21] provided guidelines for stable learning, and the *Wasserstein GAN* (WGAN) [22] improved the stability by attributing unsuccessful learning to the limit of the Kullback–Leibler (KL) divergence and redefining the loss function. Most of studies (e.g., [23, 24, 25]) have aimed to improve the network performance for image data. However, some studies have attempted to solve the data imbalance problem using GAN. In particular, the study [25] applied numerical data, not image data, to GAN. However, since GANs learn via the gradient descent method, learning problems can occur due to the loss functions [22]. Therefore, in this study, data augmentation was performed for the minority class data samples of each dataset, by applying the loss function of WGAN to alleviate the GAN's limitations and generate datasets more closely resembling the original data. Because the GAN-based minority class data-augmentation method is similar to the oversampling method, it is applied by integrating it with oversampling techniques rather than data augmentation. Therefore, in this study, we use the terms "data oversampling" and "data augmentation" interchangeably.

D. Domain Adaptation

A transfer learning is a machine-learning method that utilizes pre-learned domain information from similar domains when a specific task or domain is changed. The area in which the transfer learning model previously worked is referred to as the source domain, and the new one is referred to as the target domain; transfer learning, depending on the presence or absence of labels in the domain, is primarily divided into multi-task learning [26], in which the class exists only in the target domain; self-taught learning [27], in which the class exists in the source domain but no classes exist in the target domain; and domain adaptation [1], in which the class exists in both domains. In this study, we consider a domain adaptation model to detect anomalies between different domains. Regarding domain adaptation [28], several previous studies [29, 30, 31] have focused on minimizing the differences between the source and target domain feature-map distributions; most of these have

used the maximum mean discrepancy [32] loss function. *Deep Correlation Alignment* [29] matches the mean and covariance of the two distributions. In [31], the addition of a fully connected layer to the domain adaptation model was proposed, and a method was derived to determine the resulting value of the binary label and approximate the uniform distribution via the domain confusion loss. *ReverseGrad* [30], a gradient-reversal algorithm, calculates the gradient in the reverse direction when deriving the loss function in the network; it has exhibited a faster learning performance than comparable methods. In addition to [30], a study investigating methods of reconstructing images in the target domain was also presented in [31]. In [33], probabilities were used to learn the distribution between the two domains, and the distance between data within the same class across the two domains was expressed as a probability; learning was conducted to maximize this probability. *Adversarial Discriminative Domain Adaptation* (ADDA) [34] applied the loss function used in discriminator of GAN to match the distributions between the two domains, thereby enabling more effective learning. This method has the advantage of being able to interact with other domain-adaptation models. In this study, to facilitate interactions between similar domains, considering the advantages of ADDA, it was applied to the abnormal transaction detection model.

III. Methodology

This section describes a set of approaches conducted for fraud detection in FDSs. Section A describes the experimental dataset used in this study. Section B and C describe data augmentation to solve class-imbalance problems in learning. GAN model was used for data augmentation of minority class through the creation of new samples. It was compared to SMOTE used for data oversampling as well. Section D presents the proposed domain adaptation method, which is capable of evaluating classification performances on two datasets of similar domains. Fig. 1 shows the simplified overall structure of the model proposed in this study, and Fig. 2 illustrates the flow of this structure.

A. Dataset

The credit card dataset here employed consists of data collected by the Machine Learning Group [3] and Worldline. The dataset contains a total of 284,315 normal and 492 abnormal transaction data samples. For the data, owing to security issues (e.g., financial and personal information leaks); the test was conducted using a total of 30 variables. Similarly, the financial transaction dataset is an artificial (owing to security issues) dataset based on actual data. This dataset contains simulation results obtained through PaySim [4], using real financial transaction samples taken over a period of one month; it consists of a total of 11 variables and includes 6,354,407 normal and 8,213 abnormal transaction data samples. Unlike the credit card fraud dataset, this dataset was processed via min-max normalization before being used as input data in this work.

B. Data Oversampling

SMOTE oversamples the minority class data when class imbalances occur; in this study, it was adopted as the oversampling method because it delivers a strong performance whilst also being theoretically simple and easy to implement. First, SMOTE takes the data of a minority class and then finds the k-nearest neighbors of these data. Next, the differences between the current sample and these k neighbors are obtained, multiplied by a random value (between 0 and 1) to generate data, and combined with the original sample. It also shifts the existing data slightly, to account for the neighbors it adds. In this study, SMOTE was implemented using the imbalanced-learn Python library [35]. The oversampled data were tested with ratios of 0.3:1, 0.5:1, 0.7:1 and 1:1 between the minority and majority classes, respectively.

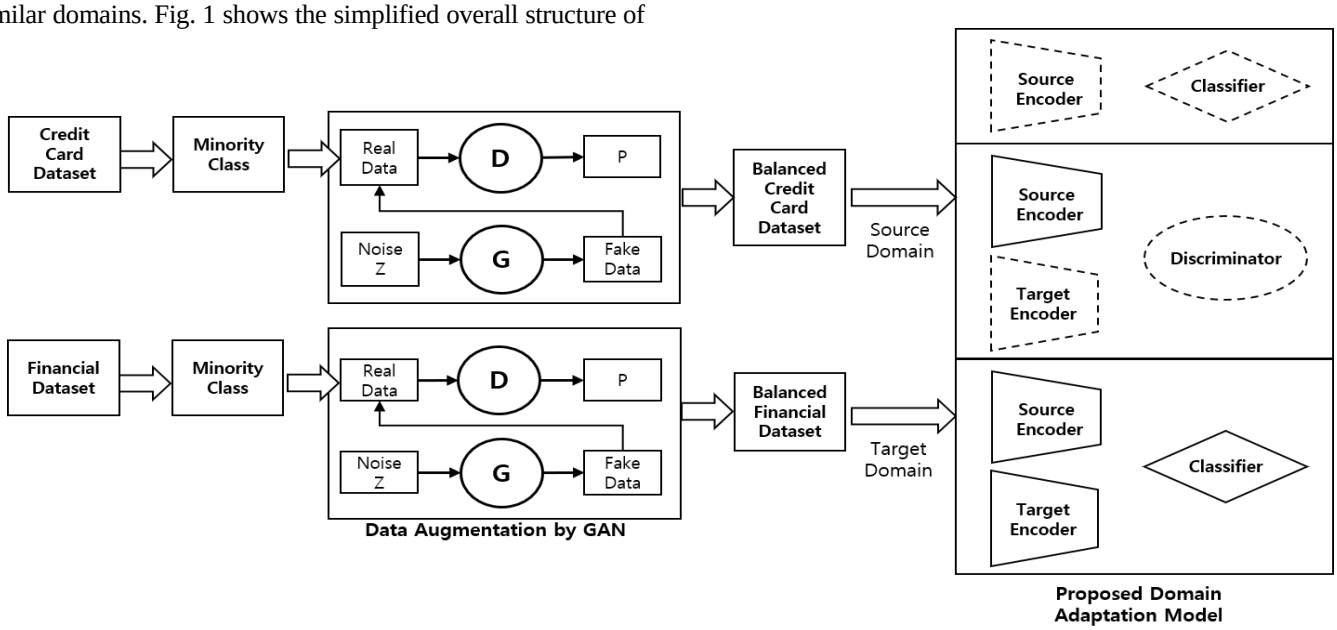

Fig. 1. Simplified Overview of the Proposed Methodology.

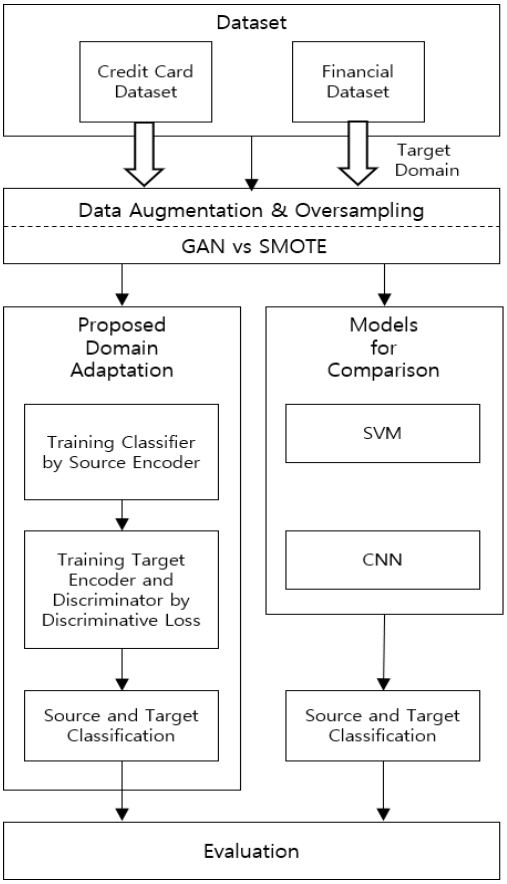

Fig. 2. An Overview Flow Chart of the Proposed Methodology.

C. Data Augmentation using GANs

In existing GANs, several problems can arise when training the GAN via the gradient descent method [22]. First, if the discriminator makes an incorrect judgment, the generator does not receive accurate feedback, and the loss function cannot learn properly. Second, if the discriminator makes a very accurate judgment, the gradient of the loss function quickly converges to 0, resulting in a significant delay or disturbance to the learning speed. Because of these two problems, existing GANs are limited. WGANs compensate for these GAN shortcomings; in them, the KL divergence, which is used to define the loss function in existing GANs, is redefined using the Wasserstein distance (also referred to as the Earth mover's distance); this is an index that measures the distance between the two probability distributions. Under KL divergence, the distance value is 0 when the two distributions overlap each other, and it is infinite or constant when they do not overlap, showing an extreme distance value. The Wasserstein distance can be readily applied in training because a constant value is maintained regardless of whether the distributions overlap. Therefore, WGANs redefine the loss function using this Wasserstein distance, to smoothly train and improve the data such that it resembles the existing data as much as possible. Therefore, in this study, oversampling was performed using the WGAN loss function within a general GAN model and inputting the minority class of the original data. The structure of the GAN-based data oversampling model is as shown in Fig. 3. Although it has an identical structure to the general GAN, the potential problems of the existing GAN have been resolved by applying the WGAN theory and loss function. For each epoch, a random noise z is fed into the generator to generate fake data, and the fake data are merged with the abnormal transaction data (the minority class) from the original dataset. The random noise is expressed as a vector of the size to be generated, and the combined data are input to the discriminator, which attempts to distinguish the original data from the fake data (generated by the generator) and classify them as either real (1) or fake (0). Using the discriminator's classification results, the generator applies loss function to minimize the classification probability and the discriminator seeks to maximize it. The loss function is expressed as.

$$\nabla_{\omega} \frac{1}{m} \sum_{1=1}^{m} \left[f\left(x^{(i)}\right) - f\left(G\left(z^{(i)}\right)\right) \right] \quad (1)$$

$$\nabla_{\theta} \frac{1}{m} \sum_{1=1}^{m} \left[f\left(G\left(z^{(i)}\right)\right) \right] \quad (2)$$

and (2) are the loss functions applied to the discriminator and generator, respectively. Above, ω is the parameter of the discriminator, and ∇ω is the gradient descent for ω. Also, θ is the parameter of the generator and ∇_{θ} is the gradient descent for θ. x is the original data, z is the random noise and G is the generator. These loss functions differ from that of existing GANs, and the purpose of the discriminator also differs therefrom. Instead of using a direct criterion for identifying the fake data generated by the generator, the discriminator learns the K-Lipschitz continuous function, which is used to calculate the Wasserstein distance. In this process, as the loss function decreases, the Wasserstein distance becomes smaller and the fake data generated by the generator approach the actual data distribution [22].

For oversampling, the WGAN loss function was applied in a GAN. Only the data in the minority classes were selected and input to the model; the random noise followed the distribution of the input data through the interaction of the generator and discriminator. Finally, when the probability of distinguishing between the input and generated data converged to 0.5, the model was terminated, and the generated data combined with input data to resolve the original data imbalance. The proportions of generated data and random noise were determined by adjusting the ratio according to the quantity of original data. For the data oversampled through SMOTE, the amount of minority class data was determined according to the sampling strategy of the original data. If the sampling strategy was 1, the [minority class: majority class] ratio became [1:1]; if the sampling strategy was 0.5, it became [0.5:1]. Therefore, to generate GAN oversampling results similar to the data processed through SMOTE, the amount of random noise z was set to (0.3, 0.5, 0.7, 1) times the size of the majority class.

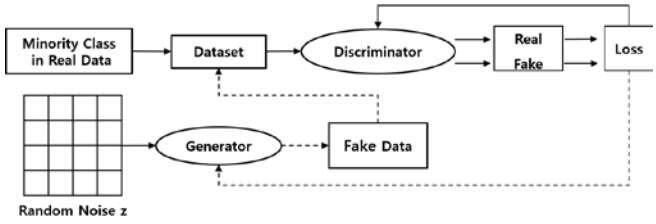

Fig. 3. Architecture for GAN-based Data Augmentation.

D. Domain Adaptation for Fraud Detection

Detecting abnormal transactions is a time-consuming and expensive process when using different models for two datasets of similar domains. Therefore, to develop a single model capable of detecting abnormal transactions from two datasets, we applied a domain-adaptation method which employs the discriminative characteristics of GANs, such as those used in ADDA [34]. While the ADDA was applied to image datasets, the proposed domain-adaptation method was applied to text datasets. Also in our study, the text datasets were augmented to avoid class imbalance problems. The domain-adaptation model used in this study was composed of source and target encoders that employed CNNs as shown in Fig. 4 and 5, respectively. Each encoder consisted of a 1D convolution layer (Conv1d), max pooling, and a fully connected layer. The convolution layer was used because it can readily extract feature maps and does not require any further layer (e.g., recurrent neural networks) for time-independent datasets. In addition, a CNN was used because these networks outperform ANNs in terms of time and performance efficiency. Two convolutional layers and two max pooling layers were used to prevent unsmooth learning or overfitting from occurring when adjusting the hyperparameters to match the feature maps. The model first learned a source encoder and classifier using the credit card fraud dataset (source domain). The loss function applied to the source encoder is expressed as follows:

$$\min L_C\left(C\left(f_s(X_S)\right), Y_S\right) \tag{3}$$

Here, C is the classifier, f_S is the source encoder, X_S is the credit card dataset, and Y_S is the credit card dataset class. Next, the financial transaction fraud dataset (target domain) was input to the CNN-based target encoder. The learning proceeded by labeling the output of the target encoder as 1 and inputting it to the discriminator. Expressed otherwise, when the discriminator receives the output of the target encoder, the learning proceeds in the direction in which the result value becomes 1. The target encoder's loss function is expressed as

$$\min L_t\left(D\left(f_t(X_t)\right), 1\right) \tag{4}$$

where D is the identifier, f_t is the target encoder, and X_t is the financial transaction dataset. The discriminator learns the distribution by labeling the output value of the source encoder as 1 (real) and the output value of the target encoder as 0 (fake), to properly distinguish between normal and fraudulent data; then, it applies a loss function. The loss function applied to the discriminator is expressed as follows:

$$\min L_D\left(D\left(f_s(X_S)\right), 1\right)$$
$$\min L_D\left(D\left(f_t(X_t)\right), 0\right) \tag{5}$$

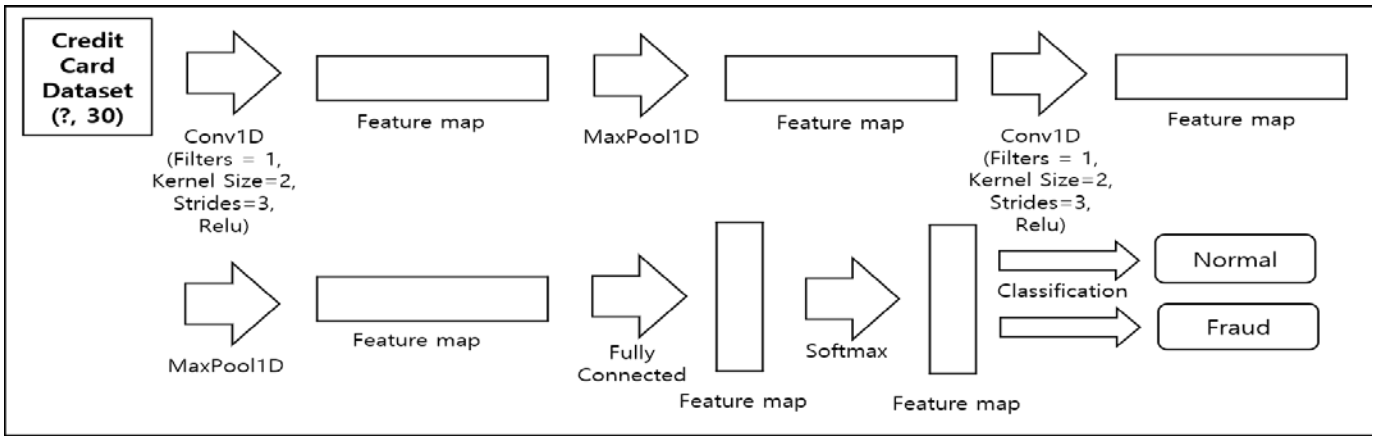

Fig. 4. Configuration of Source Encoder with Classifier.

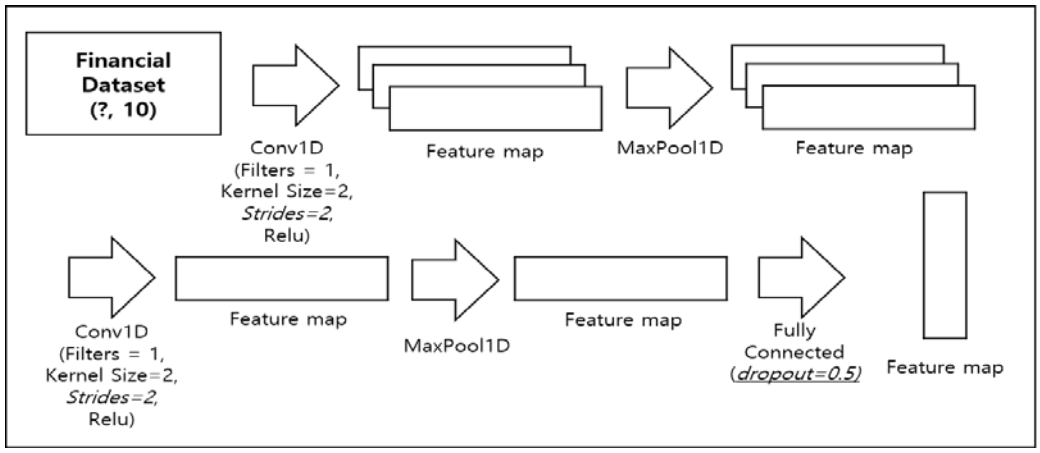

Fig. 5. Configuration of Target Encoder.

The entire learning process optimizes the loss functions described above, operating in a stepwise fashion. Based on the credit card fraud dataset (including the class information), the source encoder and classifier learn first, followed by the target encoder and discriminator. The source encoder proceeds in a fixed state whilst the target encoder and discriminator are being trained; thus, the target encoder's and discriminator's learning can proceed smoothly, without checking the state of the source encoder and classifier. Fig. 6 illustrates the overall structure of the domain-adaptation model introduced in this study; the components denoted with solid lines indicate a state in which learning is completed, and components formed of dotted lines indicate that learning takes place. Thus, the entire test process is as follows. First, the source encoder and classifier are trained on the source domain, and the discriminator and target encoder are trained from the source encoder and target domain. Finally, the proposed domain-adaptation model terminates the process when the target and source encoder can completely derive the classification results of the target and source domains, respectively.

E. Evaluation

The test results were evaluated using the area-under-curve (AUC) score, which is a classification-model performance evaluation index. The receiver operating characteristic (ROC) curve is a performance measure commonly used in binary classification and medical applications. Table I shows the confusion matrix; here, True (T)/False (F) indicates that the predicted value is the same/differs from the actual value, and Positive (P)/Negative (N) indicates how the predicted value was obtained. The ratio between the true-positive rate (TPR) and false-positive rate (FPR) is expressed as a graph of the ROC curve, and the AUC score is the area underneath this curve. The AUC score of a model with 100% incorrect prediction is expressed as 0.0, and the AUC score of a model with 100% correct predictions is expressed as 1.0; the performances of the models used in this study were evaluated accordingly.

TABLE. I. PARAMETERS IN ROC CURVE BY CONFUSION MATRIX

	Normal Prediction	Fraudulent Prediction
Normal Transaction	TN	FP
Fraudulent Transaction	FN	TP
$TPR = \frac{TP}{TP + FN}$		
$FPR = \frac{FP}{FP + TN}$		

IV. EXPERIMENTS

To evaluate the classification performance of the proposed domain-adaptation model, an SVM and CNN were employed as comparison machine and deep learning methods, respectively. Among machine learning methods, SVM has received particular attention for their excellent performance. It is a supervised learning model mainly used for pattern recognition and data analysis (in particular, classification and regression). Here, because both credit card and financial transaction datasets have class labels, SVM was used to detect abnormal transactions. The kernel of SVM uses a radial basis function. After testing values from 1 to 10,000, the hyperparameter C was set as 1000, which was found to deliver the optimal time and accuracy performances. The compositions of the source and target encoders in the proposed domain-adaptation model are as shown in Figs. 4 and 5. The source encoder sets the filter, kernel size, strides, and activation function, as shown in Fig. 4; the feature map (which undergoes max pooling after the CNN layer) passes through the fully connected layer. The output of the fully connected layer is passed to the classifier, to derive the classification result. The target encoder sets the number of strides to 2, to derive an output value with the same shape as the output value of the source encoder; other parameters (i.e., filter, kernel size, and activation function) are set identically to those of the source encoder. In addition, to prevent overfitting, a dropout was applied to the fully connected layer, with a ratio of 0.5.

The loss function of the classifier was calculated from the softmax cross-entropy, and the loss function of the discriminator was calculated using the sigmoid binary cross-entropy and optimized through the Adam optimizer (learning rate = 0.0001, beta 1 = 0.5, beta 2 = 0.99). The CNN model used the source encoder, target encoder, and classifier of the domain adaptation model. The credit card fraud data were used as the input data of the source encoder, and the financial transaction fraud dataset was used as the input data of the target encoder, to compare the classification results. The number of nodes of the hidden layer used in the GAN-based oversampling method was set to 128, the epoch was set to 20, and the Adam optimizer was set identically to the domain-adaptation model. The random noise was set as a random number extracted from a uniform distribution within the range [-1, 1]. The Ubuntu 18.04.4 LTS test environment consisted of an Intel(R) Xeon CPU E5-2620 v4 with a 2.10 GHz CPU, GTX 1080 GPU, and 64 GB RAM.

V. RESULTS AND ANALYSIS

Table II compares the classification performance results of the SVM, CNN, and proposed domain-adaption models. The experiment was conducted, and the results of the classification performance were averaged by summing only values above 0.8; this expresses the ratio between the majority and minority class when augmenting or oversampling a dataset. In other words, if the majority class is 1, a quantity of data equal to the ratio is generated to oversample the minority class.

Table III shows the time taken for each model to receive data, train it, and derive its classification results. Table IV shows the time taken to oversample each dataset with GAN and SMOTE, respectively. Fig. 7 compares the performances of the GAN- and SMOTE-based oversampling methods. The left-hand and right-hand graphs describe results for the credit card and financial transaction fraud datasets, respectively; the x-axis denotes the ratio mentioned in Table II. The AUC scores on the y-axis represent the averaged classification performances for all methods; the GAN-based oversampling method takes slightly longer than SMOTE to complete, but it exhibits a superior performance (as shown in Fig. 7). The left-hand graph in Fig. 8 shows the average classification performance for the dataset in which the GAN-based oversampling method was applied. The right-hand graph shows the time-averaged values of the GAN-based oversampling method in Table III. In Fig. 8, although the

classification performance of the domain-adaption model was inferior to those of the CNN and SVM, it was found to be suitable as an abnormal transaction detection model for both test domain datasets, because it reduced the required learning time when performing abnormal transaction detection on two datasets with similar domains. The SVM outperformed the CNN and domain-adaption models; however, it is not readily applicable to larger datasets, because its learning time increases sharply when the dataset increases. Compared to the domain-adaption model, the CNN model shows no significant difference in classification performance; however, because it requires more learning time, it is limited as a classification model for different domains.

TABLE. II. CLASSIFICATION RESULTS OF THE SVM, CNN, AND PROPOSED DOMAIN-ADAPTION MODELS IN AUC SCORE

Dataset	Method	Oversampling using GAN				Oversampling using SMOTE			
	Ratio	0.3	0.5	0.7	1	0.3	0.5	0.7	1
Credit Card	SVM	0.9984	0.9989	0.9994	0.9996	0.9639	0.9648	0.9662	0.9723
	CNN	0.9844	0.9895	0.9903	0.9862	0.9073	0.9261	0.915	0.899
	Domain Adaptation	0.9842	0.9889	0.9910	0.9888	0.9067	0.9257	0.921	0.9011
Financial	SVM	0.9986	0.9990	0.9989	0.9996	0.9701	0.9726	0.9754	0.9801
	CNN	0.88	0.8973	0.8988	0.9284	0.861	0.864	0.8721	0.9078
	Domain Adaptation	0.8821	0.8967	0.8927	0.9235	0.868	0.8701	0.8858	0.9125

TABLE. III. TIME IN SECONDS TAKEN FOR EACH MODEL TO RECEIVE DATA, TRAIN IT, AND DERIVE ITS CLASSIFICATION RESULTS

Dataset	Method	Oversampling using GAN				Oversampling using SMOTE			
	Ratio	0.3	0.5	0.7	1	0.3	0.5	0.7	1
Credit Card	SVM	198	215	257	261	185	229	265	311
	CNN	145	159	187	203	151	161	199	238
	Domain Adaptation	143	161	184	208	150	158	201	231
Financial	SVM	23940	24146	26756	28869	25442	26541	29287	30218
	CNN	1345	1973	2329	2975	1421	1898	2423	3033
	Domain Adaptation	456	657	823	1206	445	558	901	1158

TABLE. IV. DATA AUGMENTATION PROCESSING TIME IN SECONDS WITH GAN AND SMOTE

	Oversampling using GAN				Oversampling using SMOTE			
Ratio	0.3	0.5	0.7	1	0.3	0.5	0.7	1
Credit Card	16	28	43	63	0.61	0.72	0.82	1.01
Financial	303	492	714	1204	5.67	6.10	6.91	10.2

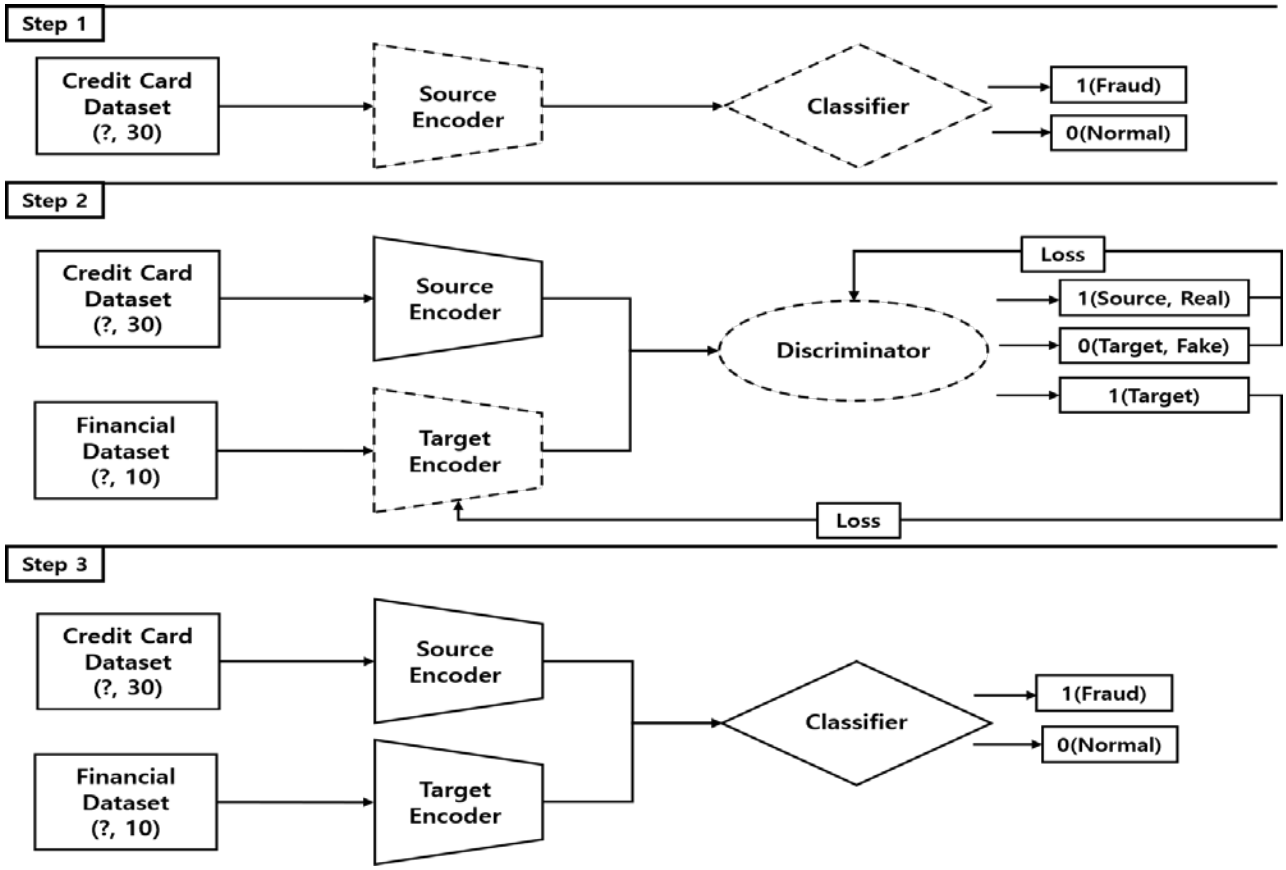

Fig. 6. Architecture of Proposed Domain-Adaptation Method for Fraud Detection.

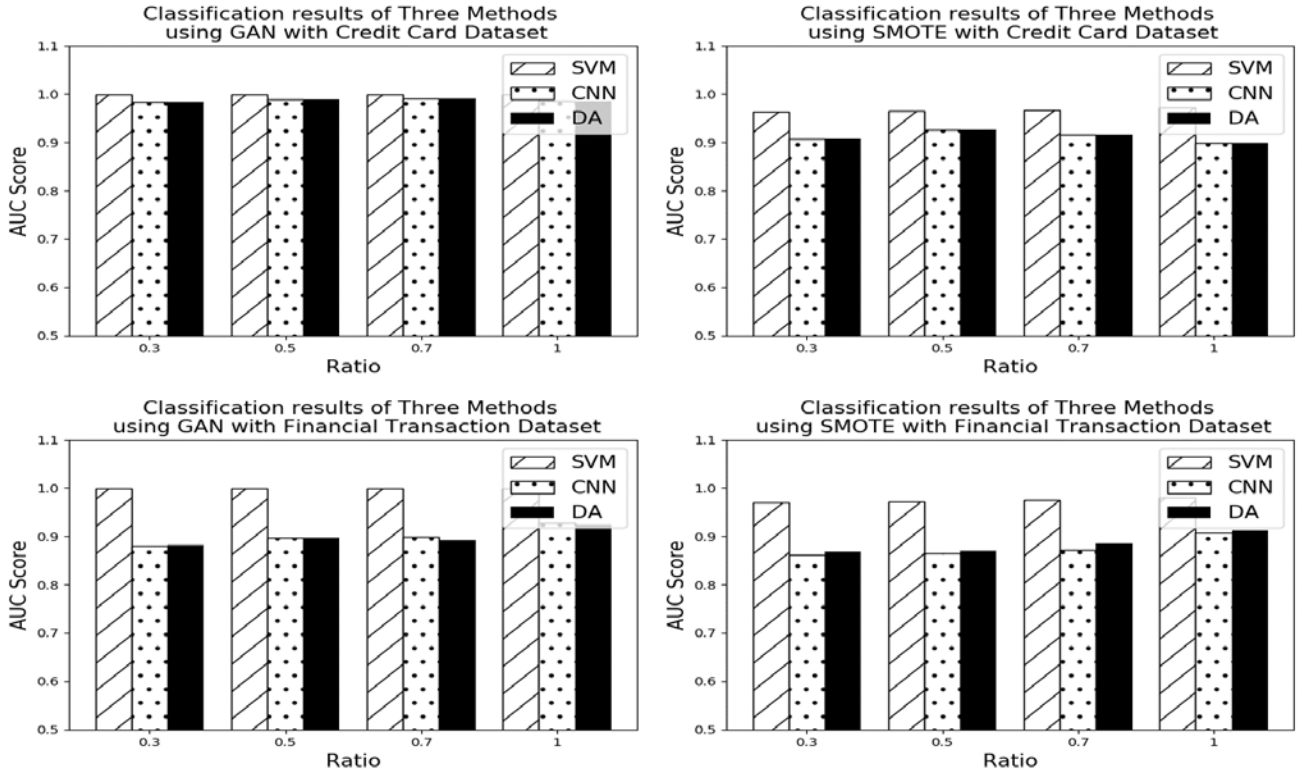

Fig. 7. AUC Scores for Datasets Augmented by GAN and SMOTE.

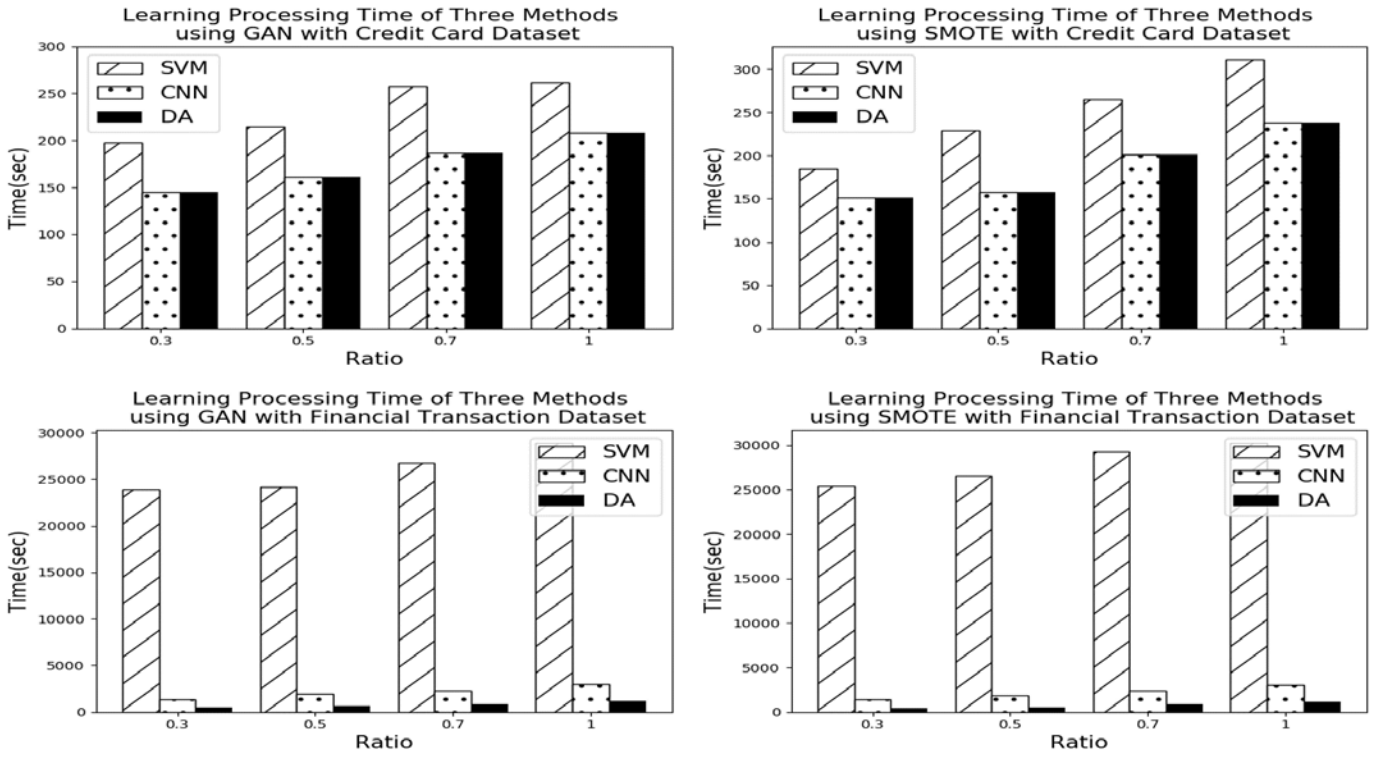

Fig. 8. Learning Processing Time for Datasets Augmented by GAN and SMOTE.

VI. Conclusions

In this study, a domain-adaptation method, applicable to data in similar domains, was proposed. The model to which the proposed domain-adaptation method was applied has the advantage of minimizing domain shifts when the domains are similar, even if the dataset has changed. In the experiments, credit card and financial transaction fraud datasets were used to evaluate the model's performance. Both datasets had a class imbalance problem; thus, oversampling was conducted using GAN and SMOTE; then, these data were used as input data of the model. Moreover, a classification performance comparison was made against SVM and CNN, to evaluate the model's performance. As a result, though the proposed domain adaption model did not achieve a better classification performance than the SVM or CNN, its performance was comparable thereto, while requiring a shorter learning time. Moreover, the GAN-based oversampling method, which was used to solve the class imbalance problem, outperformed SMOTE. Although the CNN showed a similar classification performance to the domain-adaption model, it required a longer learning time. The SVM had a high classification performance; however, it required a comparatively longer learning time than the CNN when the dataset size was increased. As a result, the proposed domain-adaptation model was shown to be capable of simultaneously classifying two datasets with similar domains and shortening the learning time compared to the SVM and CNN. However, there are several limitations to this study, which should be addressed in the future: both datasets were constructed using CNN models, to smoothly reuse the feature maps; the classification performance was insufficient compared to that of the SVM; and various domain data and results were absent. Therefore, in future research, structural changes will be made to the oversampling method proposed in this study, to make use of the various abnormal transaction data (including time-series data) and judge the performance of the model more objectively.

Acknowledgment

This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT: Ministry of Science and ICT) (No. NRF-2019R1F1A1059036).

References

[1] S. Ben-David, J. Blitzer, K. Crammer, A. Kulesza, F. Pereira, and J. W. Vaughan, "A Theory of Learning from Different Domains", Machine Learning, vol. 79, pp. 151-175, 2010, https://doi.org/10.1007/s10994-009-5152-4.

[2] L. Y. Pratt, "Discriminability-Based Transfer between Neural Networks", Advances in Neural Information Processing Systems, vol. 5, pp. 204-211, 1992, https://doi.org/10.5555/645753.668046.

[3] Machine Learning Group ULB. Credit Card Fraud Detection, 2017, https://www.kaggle.com/mlg-ulb/creditcardfraud.

[4] D. A. Lopez-Rojas, A. Elmir, and S. Axelsson, "PaySim: A Financial Mobile Money Simulator for Fraud Detection", 28th European Modeling and Simulation Symposium (EMSS 2016), vol. 28, pp. 249-255, 2016, https://doi.org/1016/j.ecolmodel.2006.04.

[5] G. Ian, P-A. Jean, M. Mehdi, X. Bing, W-F. David, O. Sherjil, C. Aaron, and B. Yoshua, "Generative Adversarial Nets", NIPS: Advances in Neural Information Processing Systems, vol. 2, pp. 2672-2680, 2014, doi/10.5555/2969033.2969125.

[6] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "SMOTE: Synthetic Minority Over-sampling Technique", Journal of Artificial Intelligence Research, vol. 16, pp. 321-357, 2002, https://doi.org/10.1613/jair.953.

[7] C. Cortes and V. Vapnik, “Support-Vector Networks”, Machine Learning, vol.20,pp.273-297, 1995, https://doi.org/10.1007/BF00994018.

[8] Y. LeCun, B. Bose, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, and L D. Jackel, “Backpropagation Applied to Handwritten Zip Code Recognition”, Neural Computation, vol. 1, no. 4, pp. 541-551, 1989, https://doi.org/10.1162/neco.1989.1.4.541.

[9] M. Tatsuya and N. Ayahiko, “Proposal of Credit Card Fraudulent Use Detection by Online Type Decision Tree Construction and Verification of Generality”, International Journal for Information Security Research, vol.1, pp. 229–235, 2013, https://doi.org/10.20533/ijisr.2042.4639.2013.0028.

[10] M. Singh, S. Kumar, and T. Garg, “Credit Card Fraud Detection Using Hidden Markov Model”, International Journal of Engineering and Computer Science, vol. 8, pp. 24878-24882, 2019, https://doi.org/10.18535/ijecs/v8i11.4386.

[11] N. Malini and M.Pushpa, “Analysis on Credit Card Fraud Identification Techniques Based on KNN and Outlier Detection”, International Conference on Advances in Electrical, Electronics, Information, Communication and Bio-Informatics (AEEICB), 2017, https://doi.org/10.1109/AEEICB.2017.7972424.

[12] N. K. Gyamfi and J-D. Abdulai, “Bank Fraud Detection Using Support Vector Machine”, IEEE 9th Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON), 2018, https://doi.org/10.1109/IEMCON.2018.8614994.

[13] M. A. Al-Shabi, “Credit Card Fraud Detection Using Autoencoder Model in Unbalanced Datasets”, Journal of Advances in Mathematics and Computer Science, vol. 33, pp. 1-16, 2019, https://doi.org/10.9734/jamcs/2019/v33i530192.

[14] U. Fiore, A. D. Santis, F. Perla, P. Zanetti, and F. Palmieri, “Using Generative Adversarial Networks for Improving Classification Effectiveness in Credit Card Fraud Detection”, Information Sciences, vol. 479, pp. 448-455, 2019, https://doi.org/10/1016/j.ins.2017.12.030.

[15] H. He and E. A. Garcia, “Learning from Imbalanced Data”, IEEE Transactions on Knowledge and Data Engineering, vol. 21, no. 9, pp. 1263-1284, 2009, https://doi.org/10.1109/TKDE.2008.239.

[16] S. Barua, M. M. Islam, X. Yao, and K. Murase, “MWMOTE: Majority Weighted Minority Oversampling Technique for Imbalanced Data Set Learning”, IEEE Transactions on Knowledge and Data Engineering, vol. 26, no. 2, pp. 405-425, 2012, https://doi.org/10.1109/TKDE.2012.232.

[17] G. Menardi and N. Torelli, “Rose: Random Over-sampling Examples”, Data Mining and Knowledge Discovery, vol. 28, no. 1, pp. 92-122, 2014, https://doi.org/10.1080/24699322.2019.1649074.

[18] B. Baesens, S. Höppner, I. Ortner, and T. Verdonck, “robROSE : A Robust Approach for Dealing with Imbalanced Data in Fraud Detection”, 2020, ariv:2003.11915v1.

[19] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet Classification with Deep Convolutional Neural Networks", Proceedings of the 25th International Conference on Neural Information Processing Systems (NIPPS), vol. 1, pp. 1097-1105, 2012, https://doi.org/10.1145/3065386.

[20] T. Karras, T. Alia, S. Laine, and J. Lehtinen, “Progressive Growing of GANs for Improved Quality, Stability, and Variation”, Computing Research Repository (CoRR), vol. abs/1710.10196, 2017, http://arxiv.org/abs/1710.10196.

[21] A. Radford, L. Metz, and S. Chintala, “Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks”, Proceedings of the 25th International Conference Learning Representations (ICLR), pp. 1-16, 2016, https://arxiv.org/abs/1511.06434.

[22] M. Arjovsky, S. Chintala, and L. Bottou, “Wasserstein GAN”, ICML'17: Proceedings of the 34th International Conference on Machine Learning, vol. 70, pp. 214-223, 2017, https://arxiv.org/abs/1701.07875.

[23] M. Frid-Adar, E. Klang, M. Amitai, J. Goldberger, and H. Greenspan, “Synthetic Data Augmentation using GAN for Improved Liver Lesion Classiffication”, IEEE 15th International Symposium on Biomedical Imaging (ISBI 2018), vol. 15, 2018, https://doi.org/10.1109/ISBI.2018.8363576.

[24] J. T. Springenberg, “Unsupervised and Semi-supervised Learning with Categorical Generative Adversarial Networks”, 2015, https://arxiv.org/abs/1511.06390.

[25] F. H. K. d. S. Tanaka, and C. Aranha, “Data Augmentation Using GANs”, Computing Research Repository (CoRR), 2019, http://arxiv.org/abs/1904.09135.

[26] R. Caruana, “Multitask Learning,” Machine Learning, vol. 28, pp. 41-75, 1997, https://doi.org/10.1023/A:1007379606734.

[27] R. Raina, A. Battle, H. Lee, B. Packer, and A. Y. Ng, “Self-Taught Learning: Transfer Learning from Unlabeled Data”, Proceedings of the 24th International Conference on Machine Learning(ICML), pp. 759–766, 2007, https://doi.org/10.1145/1273496.1273592.

[28] H. Daume III, and D. Marcu, “Domain Adaptation for Statistical Classifiers”, Journal of Artificial Intelligence Research, vol. 26, pp. 101-126, 2016, https://doi.org/10.1613/jair.1872.

[29] B. Sun, and K. Saenko, “Deep CORAL: Correlation Alignment for Deep Domain Adaptation”, European Conference on Computer Vision, pp. 443-450, 2016, http://arxiv.org/abs/1607.01719.

[30] Y. Ganin, and V. Lempitsky, “Unsupervised Domain Adaptation by Backpropagation”, Proceedings of the 32nd International Conference on Machine Learning (ICML-15), pp. 1180-1189, 2015, https://doi.org/10.5555/3045118.3045244.

[31] E. Tzeng, J. Hoffman, T. Darrell, and K. Saenko, “Simultaneous Deep Transfer Across Domains and Tasks”, IEEE International Conference on Computer Vision (ICCV), 2015, https://doi.org/10.1109/ICCV.2015.463.

[32] A. Gretton, A. J. Smola, J. Huang, M. Schmittfull, K. Borgwardt, and B. Schölkopf, “Covariate Shift and Local Learning by Distribution Matching”, Dataset Shift in Machine Learning (in Book), pp. 131-160, Dec. 2008, https://doi.org/10.7551/mitpress/9780262170055.003.0008.

[33] X. Xu, X. Zhou, R. Venkatesan, G. Swaminathan, and O. Majumder, “d-SNE: Domain Adaptation Using Stochastic Neighborhood Embedding”, IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 2497-2505, 2019, https://doi.org/10.1109/CVPR.2019.00260.

[34] E. Tzeng, J. Hoffman, K. Saenko, and T. Darrell, “Adversarial Discriminative Domain Adaptation”, IEEE Conference on Computer Vision and Pattern Recognition (CVPR) , 2017, https://doi.org/10.1109/CVPR.2017.316.

[35] G. Lemaitre, F. Nogueira, and C. K. Aridas, “Imbalanced-learn: A Python Toolbox to Tackle the Curse of Imbalanced Datasets in Machine Learning”, Journal of Machine Learning Research, vol. 18, pp. 1-5, 2017, https://arxiv.org/abs/1609.06570.