Financial Fraud Detection using Deep Support Vector Data Description

Masoud Erfani, Farzaneh Shoeleh, Ali A. Ghorbani
Canadian Institute for Cybersecurity
University of New Brunswick
Fredericton, Canada
{masoud.erfani, fshoeleh, ghorbani}@unb.ca

2020 IEEE International Conference on Big Data (Big Data) | 978-1-7281-6251-5/20/$31.00 ©2020 IEEE | DOI: 10.1109/BigData50022.2020.9378256

***Abstract*—Nowadays, most financial transactions are virtual all over the world. The rapid usage of credit cards and online transnational applications raises fraudulent activities using these services. So, fraud detection is one of the challenging real-world problems. One of the main challenges in fraud detection is imbalanced datasets, where there are very few cases of fraud in an extremely large amount of non-fraud samples. Also, the behavior of fraud changes frequently making the learning process for the state-of-the-art machine learning binary classifiers complicated. As a result, in this paper, we propose an efficient framework for fraud detection. Our framework consists of a novel preprocessing and subsampling step, which is followed by applying deep support vector data description for fraud detection. We provide a trend analysis based on the size of the training, test datasets, and performance of the model using Area Under the Receiver Operating Characteristic Curve(ROC-AUC) and Average Precision(AP) as metrics. Finally, based on results, our approach outperforms SVM and Random Forest as the state-of-the-art binary classifiers in different scenarios. It achieves a remarkable performance in terms of AP and ROC-AUC equal to 90% and 93%(Best results), respectively.**

***Index Terms*—Fraud detection, DeepSVDD, imbalanced datasets, one class classification, Financial Fraud**

I. Introduction

Fraud attempts have been increasing recently. In different sectors of industry including the financial sectors, banking, government agencies, insurance, credit card, and law enforcement, fraud detection is applicable. Fraud imposes a considerable financial loss to different divisions of a business. According to released statistics, most convictions and the largest reported losses due to fraud were seen in Ontario, Quebec, Alberta, and British Columbia. Also, fraud incidents have caused great financial losses. For example, in Canada, fraud caused a loss of 14.3 million dollars to the province of British Colombia [1]. Also, there has been an increase in the number of fraud incidents in recent years such as ZeuS malware and its variants that are utilized to cause fraud and steal money from banks [2], which delegates fraud detection as an important research topic. One of the main challenges in fraud detection is that practically only a small percentage of users have fraudulent intentions. In the fraud detection problem, the dataset is highly imbalanced because the number of fraud samples is significantly lower than non-fraud examples. An imbalanced dataset makes the learning process complex.

Two research perspectives exist toward an imbalanced dataset. In the first direction, the main goal is balancing the dataset using undersampling or oversampling techniques. In the undersampling approach, a random subset of samples from the majority class (non-fraud) along with fraud instances are used to train the detection model [3]. While oversampling means generating a new set of examples similar to the minority class (fraud) to make the training dataset more balanced. For example, Fiore et. al. exploited generative adversarial networks (GAN) as an oversampling technique to improve classification effectiveness in credit card fraud detection [4]. The second outlook emphasizes using a proper machine learning algorithm that fulfills the requirements of the imbalanced problem. Numerous machine learning algorithms, namely supervised and unsupervised, have been proposed for fraud detection. Supervised models look at fraud detection as a binary classification problem, where the goal is to classify labeled samples in two groups. Unsupervised approaches aim in categorizing unlabeled examples into clusters in which their members possess similar behavior.

This paper focuses on both outlooks by proposing clustering-based subsampling along with leveraging a deep one-class classifier to deal with the imbalanced dataset.Here, we proposed a fraud detection based on Deep Support Vector Data Description (DeepSVDD). DeepSVDD was utilized as a new anomaly detection algorithm in other domains such as computer vision [5]. To the best of our knowledge, there has not been any research to show the effectiveness of the DeepSVDD model in fraud detection problem.

In this paper, we propose an efficient framework for fraud detection with the following contributions:

- We propose a new framework using DeepSVDD as an efficient one-class classifier to model non-fraud samples.
- We propose a new preprocessing methodology for subsampling based on clustering the non-fraud samples.
- We provide trend analysis for the imbalanced class problem based on the size of the training dataset and imbalanced ration of the test dataset.
- We employ two versions of our proposed model using both one-class and soft-boundary versions of SVDD. We also exploit their behavior and performance in the fraud detection domain.
- Our experimental results demonstrate that our proposed

framework outperforms the state of art machine learning algorithms in terms of the Receiver Operating Characteristic Curve(ROC-AUC) and Average Precision(AP).

The rest of the paper is organized as follows. Section II provides a summary of the previous approaches for fraud detection. We propose our framework in section III. Section IV contains our experimental results using a publicly available real dataset. Finally, we conclude our proposal and argue for possible future works in section V.

II. Related work

Based on the amount and availability of data, fraud detection solutions are classified into two groups: Rule-based system and AI-based methods [6]. Rule-based systems work based on establishing a set of predefined rules to detect fraud patterns, whereas AI-based methodologies rely on learning the behavior of samples. Nowadays, with the enormous amounts of data that we encounter, rule-based approaches do not apply to fraud detection anymore. However, AI-based techniques that utilize machine learning and deep learning as new tools can be employed to keep up with demand.

We present the related work in two subsections, namely statistical machine learning and deep learning approaches.

A. Statistical machine learning approaches

In terms of the availability of labels, fraud detection models that used statistical machine learning approaches can be categorized into two main groups, namely unsupervised and supervised models.

Unsupervised fraud detection algorithms attempt to group different samples based on their behavior. In [7], an unsupervised framework based on k nearest neighbor (KNN) was proposed for fraud detection. A behavior cluster-based method was conducted to deal with an imbalanced dataset in [8]. In this work, major and minor classes in the training dataset were grouped into different clusters. After removing the user behavior noise, distinct clusters of the minor class were mixed with a group of samples from the major class. In their proposal, behavior noise is considered those transactions in which behavior is like the opposite class members. In another paper [9], authors implemented a hierarchical clustering strategy to solve the imbalanced datasets problem. They exploited two clustering algorithms, namely K-means and Gaussian Mixture Model (GMM), to build the clustering tree-based framework on the behavior of the dataset. The clustering tree-based model possesses three kinds of leaves: 1) *Single label* which consists of examples from one class, 2) *Class-balanced* where samples are balanced and the ratio of majority and minority classes is lesser than a specific threshold, 3) *leaf nodes* with abnormal samples in which the number of samples is fewer than the threshold of the first group. They adopted a peculiar approach for dealing with three kinds of leaves. For the first group, their strategy returned the label for samples in this category. The second group is balanced so, traditional machine learning algorithms can be used to classify examples. The third group is still imbalanced; therefore, they employed anomaly detection algorithms to distinguish fraud samples.

Supervised learning or classification approaches rely on the availability of labels to train the models that are able to classify unseen samples. Random forest is an ensemble of decision trees that has been utilized for classification. In [10], two versions of random forest were applied to fraud detection problems. In the first version, at each node, first, a subset of attributes was selected. Then, the centers of the different classes of data were computed in the current node. Finally, the tree was structured based on the distance of attributes to centers of data. Whereas, the second version of the random forest employed the best attribute for internal nodes. The best attribute was selected based on the GINI index. The survey paper [11] summarized different supervised machine learning approaches that were applied for fraud detection, such as Logistic Regression, Support Vector Machine, Decision Tree, Random Forest, and Naive Bayes. In [12], fraud detection was formulated in a streaming setting. Authors proposed a system to raise every day a fixed and a small number of alerts, which would lead to more investigations and finally label the associated transactions. Their proposal was an ensemble learner consisting of two classifiers, namely feedback and delayed. The feedback classifier is based on the mentioned active learning process; however, the delayed learner is trained on old transactions. In [13], support vector data description(SVDD) with a combination of REDBSCAN as an enhanced version of DSCAN for feature subsampling was proposed for fraud detection. Lucas et. al. employed Hidden Markov Model to assign a likelihood for each transaction [14]. After this assignment, the obtained value was considered as an additional feature in the fraud detection dataset. Finally, the Random Forest algorithm was trained on the enhanced dataset to distinguish fraud and non-fraud samples.

B. Deep learning based approaches

Deep learning is a family of machine learning techniques that is based on artificial neural networks with representation learning. Recently, it has enabled many practical applications of machine learning. Generally, deep learning algorithms can be categorized into two groups: discriminative and generative models. First, we present fraud detection models using discriminative methods such as convolutional neural networks and recurrent neural networks. Then, we review the important generative models that have been applied in this area.

Convolutional Neural Networks (CNN) is a neural network that is firstly designed and applied in computer vision and image processing. A CNN is composed of three kinds of layers, namely convolutional, pooling, and fully connected layers. Zhang et al. introduced an online transaction fraud detection model based on the convolutional neural network [15]. They employed the transactions from a commercial bank B2C for training and testing. In [16] researchers design a CNN architecture to detect credit card fraud. They also defined and extracted a novel trading feature called trading entropy based on the latest consumption preference for each

customer. All extracted features were transformed into a feature matrix to feed into the CNN model. Besides, they generate synthetic fraud samples from real fraud using the cost-based oversampling technique. In another research paper, a deep feed forward by using a hybrid optimization method was utilized to detect fraud incidents in IOT [17]. Jurgovsky et al. proposed sequence classifiers for credit-card fraud detection [18]. They used Long Short-Term Memory (LSTM) network to incorporate transaction sequences. In addition, they used the undersampling approach to deal with imbalanced datasets. Based on a fascinating conclusion they presented, fraud samples detected by sequence learners are different from those distinguished by statistical learners such as Random Forrest. Recurrent Neural Network (RNN) is another well-known family of discriminative deep neuron network, where the connection graph contains at least one cycle. In [19], the authors applied RNN and LSTM as their deep learning models on a dataset with 80 million transactions collected from different banks. A major limitation of their models is the low performance in terms of detecting fraud samples.

Deep Auto Encoders (DAEs) and Restricted Boltzmann Machine (RBM) are examples of unsupervised deep learning approaches. DAEs are composed of two symmetrical deep-belief networks named encoders and decoders. Encoders typically hold four or five shallow layers and the second set of layers represents the decoder. Pumsirirat et al. utilized DAE and RBM as their deep learning model for fraud detection [20].

In the next section, we present our proposed fraud detection framework utilizing deep support vector data description, as a well-known semi-supervised deep learning model which only needs non-fraud samples to classify samples by finding a transformation function.

III. Methodology

We propose a methodology for dealing with the imbalanced problem in fraud detection datasets by applying DeepSVDD as a one-class classifier and proposing a clustering-based subsampling technique. Figure 1 demonstrates our proposal for an efficient fraud detection approach. It mainly consists of two phases: 1) preparing the data by applying PCA to reduce the dimensionality of the problem and clustering-based subsampling, then 2) applying DeepSVDD as a one-class classifier to utilize non-fraud samples for fraud detection. In the following subsections, we will elaborate on each phase in detail.

A. Preprocessing and Subsampling

In this phase, the raw training data is processed to obtain a new training dataset that is more balanced and suitable for training the detection model. This phase consists of three sequential steps: feature reduction, subsampling, and normalization. The output is a dataset suitable for the detection learning process.

1) Feature Reduction: We employ Principal Component Analysis (PCA) as a feature reduction technique to reduce the dimensionality of a dataset while preserving as much statistical information as possible. We apply PCA because of its two main advantageous for real-world detection problems: first, PCA is an unsupervised learning method, and it simplifies the complexity in high-dimensional data while keeping trends and patterns.

After data standardization, PCA computes the covariance Matrix of input features (X) and label (Y) using Equation 1.

$$cov(X,Y) = \frac{1}{(n-1)} \times \sum_{i=1}^{n}(X_i - \overline{x})(Y_i - \overline{y}) \qquad (1)$$

Where n is the number of samples, X and Y are the mean of input features and labels, respectively. After computing the Covariance Matrix, the eigenvectors of this matrix corresponding to the d ($d \leq$ number of features) highest eigenvalues are selected to map the dataset into a new feature space with d dimensions.

2) Clustering-based Sampling: Fraud detection datasets are significantly imbalanced. One possible way to overcome this issue is subsampling. Here, we propose a clustering-based sampling approach to have a more balanced dataset. We adopt K-means algorithms with $k = 3$ to categorize training data into three groups, named *Far*, *Middle*, and *Close*. K-means identifies k number of centroids randomly, and then allocates every data point to the nearest cluster while keeping the centroids as small as possible. In our case, we select centroids based on their distance to all non-fraud samples. To do so, first, we calculate the average distance of fraud samples to each other, using Equation 2.

$$f = \frac{1}{m^2}\sum_{i=1}^{m}\sum_{j=1}^{m} \parallel x_i' - x_j' \parallel^2, \qquad (2)$$

Where m is the number of fraud sample, and x' denotes a fraud sample. The goal of clustering in our proposal is to categorize non-fraud samples into three groups based on their Euclidean distance with fraud samples. Therefore, we randomly select the three centroids, namely C_1, C_2, and C_3, by considering Equation 3 and the following conditions:

1) C_1 is centroid of *Close* cluster if $d(C_1) \leq \beta$.
2) C_2 is centroid of *Middle* cluster if $\beta < d(C_2) \leq \beta^2$.
3) C_3 is centroid of *Far* cluster if $d(C_3) > \beta^2$.

$$d(C_i) = \frac{1}{m}\sum_{j=1}^{m} \parallel C_i - x_j' \parallel^2, \qquad (3)$$

where β is a parameter to form the clusters based on the three cluster's average distances to fraud average distance. These conditions should be satisfied while initializing the centroids and also updating the centroids of clusters during the clustering process. Besides, in the assigning phase where the non-fraud samples will be assigned into clusters, the updated clusters must meet three following conditions by considering Equation 4 where $\{C_k^*\}$ denotes the cluster of n non-fraud samples, x_i:

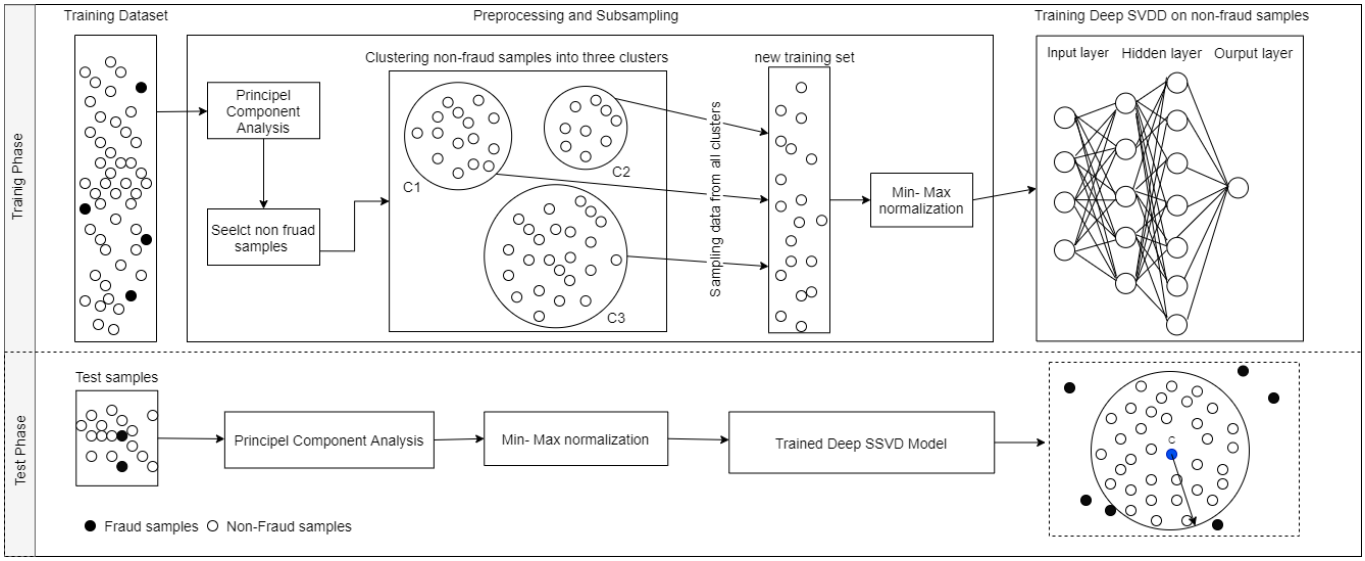

Fig. 1. Overall view of proposed framework

1) C_1^* is *Close* cluster if $d(\{C_1^*\}) \leq \beta \times f$. It means that if the average distance of each sample of a cluster from fraud examples is lesser than β times the average distance of fraud samples then this cluster is called *Close*.
2) C_2^* is *Middle* cluster if $\beta \times f < d(\{C_2^*\}) \leq \beta^2 \times f$. In Middle cluster, this average distance should be between $\beta \times f$ and $\beta^2 \times f$.
3) C_3^* is *Far* cluster if $d(\{C_3^*\}) > \beta^2 \times f$. In *Far* cluster, this average distance is more than $\beta^2 \times f$.

$$d(\{C_k^*\}) = \frac{1}{n \times m} \sum_{i=1}^{n} \sum_{j=1}^{m} \parallel x_i - x'_j \parallel^2 \tag{4}$$

After clustering the non-fraud samples regarding their distance with fraud examples into three groups, we select r_k ratio ($r_k < 1$) of instances of each cluster to insert into a new training dataset. Since our training dataset consists of only non-fraud samples, this subsampling of non-fraud samples can help our model better generalize to different non-fraud samples with similar behavior and distinguish non-fraud examples from fraud instances.

3) Normalization: After generating a new training dataset using clustering-based subsampling, we utilize the Min-Max normalization approach to normalize the continuous data into the range of [0, 1]. It is crucial to normalize data before applying a deep learning method because unscaled input variables can result in a slow or unstable learning process. The mathematical formula for Min-Max normalization is represented as follow:

$$X_{new} = \frac{X - X_{min}}{X_{max} - X_{min}} \tag{5}$$

B. One Class Deep Support Vector Description

Deep SVDD learns to extract the common factors of variation of the data distribution by training a neural network. This algorithm endeavors to find the smallest hypersphere which encloses the normal data. In other words, the neural network structure in SVDD plays as a mapping function to transform the data into the hypersphere of minimum volume. So, in addition to the ability to model non-fraud samples, DeepSVDD can extract useful feature representations of the data with the benefit of using deep neural network structure.

DeepSVDD has two versions, namely soft-boundary Deep SVDD and one-class DeepSVDD. Both versions have an objective function for finding the optimal discriminative hypersphere. In the objective function of soft-boundary DeepSVDD, the goal is to find the smallest hypersphere that holds all normal samples inside and all anomalies outside. The following equation represents the soft-boundary objective [5]:

$$\begin{aligned} \min_{R,W} \quad & R^2 + \frac{1}{vn} \sum_{i=1}^{n} max\{0, ||\phi(x_i; W) - c||^2 - R^2\} \\ & + \frac{\lambda}{2} \sum_{l=1}^{L} ||W^l||_F^2, \end{aligned} \tag{6}$$

where W denotes the weights of neural network transformation,n is the number of non-fraud samples, L is the number of layers, c and R are the center and radius of optimal hypersphere, respectively, and v is a hyperparameter to control the trade-off between the volume of the sphere and violations of the boundary. The last term for this objective function is a weight decay regularizer on textitFrobenius norm of W ($\parallel W \parallel_F^2$) with hyperparameter $\lambda > 0$.

One-class DeepSVDD aims to handle the imbalanced datasets. In contrast to soft-boundary, training examples of this version of DeepSVDD consist of only normal samples. Since in fraud detection datasets, the majority of instances

belong to the non-fraud category, and a too-small portion is associated with the fraud class, one-class is more suitable than soft-boundary in this domain. The objective function of one-class DeepSVDD is defined as following [5]:

$$\min_{W} \quad \frac{1}{n}\sum_{i=1}^{n}||\phi(x_i;W)-c||^2+\frac{\lambda}{2}\sum_{l=1}^{L}||W^l||_F^2 \qquad (7)$$

Soft-boundary Deep SVDD contracts the hypersphere by penalizing the radius directly and the data representations that fall outside the sphere. Whereas One-Class DeepSVDD contracts the hypersphere by minimizing the mean distance of all non-fraud samples representations to the center. The neural network extracts the common factors to map the non-fraud data as close to center c as possible. Since the majority of training data is from one class, One-Class DeepSVDD penalizes the mean distance over all samples instead of allowing some data to fall outside the hypersphere.

IV. Experimental Results

In this section, we examine our proposed framework on well-known dataset, which publicly available. We also present some trend analyses of fraud and non-fraud samples considering our proposed clustering subsampling approach. Besides, we evaluate the obtained results of both versions of DeepSVDD on fraud detection problem. Finally, We employ the support vector machine and random forest on the selected dataset and compare their performance with our proposed method.

The dataset that we exploit for our experiments includes the transactions made by credit cards in September 2013 by European cardholders[1]. This dataset contains transactions that occurred in two days, including 492 frauds out of 284,807 transactions. The dataset is highly imbalanced where only 0.172 percent of all transactions are fraud accounts. We perform the clustering-based subsampling with $\beta = 5$, as described in our proposed method on the dataset. The non-fraud samples are grouped into three clusters, and I summarizes clusters' properties.

TABLE I
CLUSTERING NON-FRAUD SAMPLES AND THEIR PROPERTIES.

Name	Population	Average distance to fraud samples
Close	272263	145.29
Middle	11439	688.063
Far	613	3305.65

For evaluating models' performance, we utilize the Area Under the Receiver Operating Characteristic Curve (ROC-AUC) and Average Precision(AP) metric. ROC-AUC describes the degree or measure of separability. It indicates how much the model can differentiate between classes. The higher the ROC-AUC, the better the model is at separating between fraud and non-fraud samples. Considering fraud as a target class, Average Precision summarizes the precision-recall curve as a weighted mean of precision achieved at each threshold, with the increase in recall from the previous threshold used as the weight. The following equation summarizes how AP is calculated.

$$AP = \sum (R_n - R_{n-1}) \times P_n, \qquad (8)$$

where R_n and P_n are recall and precision at the nth threshold.

[1]https://www.kaggle.com/mlg-ulb/creditcardfraud

TABLE II
STRUCTURE OF NEURAL NETWORK.

Layer	Type	Activation function	Input	Output
Layer 1	FC	LeakyRelu + Dropout	29	25
Layer 2	FC	LeakyRelu	25	15
Layer 3	FC	LeakyRelu	15	10
Layer 4	FC	LeakyRelu	10	1

TABLE III
HYPER-PARAMETERS OF NEURAL NETWORK.

Optimizer algorithm	Learning rate	Epochs	batch size
Adam	0.001	50	128

In this paper, we investigate the performance of two versions of Deep SVDD in the fraud detection domain. In both versions, a feed-forward neural network is exploited as the transformation function. Table II and III present the topology and hyperparameters of the this neural network, respectively.

In the following, we firstly present the obtained results of one-class DeepSVDD. Second, we report the performance of soft-boundary DeepSVDD applying on the selected dataset. Then, we demonstrate a comparison between the performance of one-class and soft-boundary DeepSVDD models. Finally, we compare our proposed framework with the state of art machine learning models.

A. One-class DeepSVDD

Since one-class DeepSVDD is a semi-supervised method, we train our model on non-fraud samples. However, the test dataset includes different portions of non-fraud samples and all fraud data (492 samples). Table IV and V illustrate the performance of our model in terms of ROC-AUC and AP, respectively. In both tables, for the trend analysis, the number of non-fraud samples begins at 1000 and increases by multiplication of two (2,4,8,16,32,64 and 128). Also, the number of training samples is considered 40000, 8000, and 16000. As these tables indicate, by decreasing the percentage of fraud samples in the test dataset, the ROC-AUC of trained model increase, whereas the AP of model decrease. Figures 2 and 3 illustrate these trends. It is shown that as the test dataset becomes more imbalanced, means the percentage of fraud samples in the test dataset decrease, the average precision of

TABLE IV
TREND ANALYSIS BASED ON ROC-AUC METRIC FOR ONE-CLASS DEEPSVDD

Training samples	Number of non-fraud samples in Test							
	1000	2000	4000	8000	16000	32000	64000	128000
40000	0.90	0.91	0.88	0.90	0.90	0.91	0.92	0.93
80000	0.89	0.89	0.91	0.91	0.89	0.92	0.93	0.93
160000	0.89	0.90	0.89	0.87	0.92	0.90	0.92	0.93
	32%	19%	10%	5.79%	2.98%	1.51%	0.76%	0.38%
	Percentage of fraud samples in Test							

TABLE V
TREND ANALYSIS BASED ON AP METRIC FOR ONE-CLASS DEEPSVDD

Training samples	Number of non-fraud samples in Test							
	1000	2000	4000	8000	16000	32000	64000	128000
40000	0.90	0.88	0.83	0.82	0.80	0.79	0.77	0.72
80000	0.90	0.87	0.85	0.83	0.80	0.79	0.75	0.72
160000	0.90	0.87	0.83	0.81	0.81	0.78	0.74	0.70
	32%	19%	10%	5.79%	2.98%	1.51%	0.76%	0.38%
	Percentage of fraud samples in Test							

our models declines. The reason behind that is a group of fraud samples holds similar behavior to non-fraud examples. So, the model predicts them as non-fraud, which affects the average precision. Also, this dataset contains transactions from a different group of anonymous users. Various credit card-holders typically hold different behavioral patterns. Despite the facts that mentioned, Deep one-class classifiers acquired comparable results in terms of AP. On the other hand, ROC-AUC increases since the number of non-frauds augments. Thus, the model can predict them correctly. In the case of imbalanced datasets, AP with outliers as the target is a better metric to evaluate the model performance because ROC-AUC is not able to reflect the performance of models in imbalanced datasets. This metric counts the number of instances that are labeled correctly. Therefore, in imbalanced cases, it calculates a high number for classifiers that predict all samples' labels normal.

Figure 4 demonstrates the training loss of the model over 128 epochs considering different sizes of the training dataset. As the figure shows, the training model on more than 80000 samples does not help the learning process.

B. Soft-boundary DeepSVDD

The second version of our model utilizes soft-boundary DeepSVDD, which establishes the hyper-sphere based on both fraud and non-fraud samples and endeavors to keep the fraud examples out of the hyper-sphere. The structure of the neural network in the soft-boundary DeepSVDD model is the same as the structure in the one-class SVDD model. In our analysis, the training dataset consists of non-fraud samples and 100 fraud examples. Soft-boundary results are reported in Tables VI and VII. As tables indicate, soft-boundary DeepSVDD results are promising, but one-class DeepSVDD still outperforms this algorithm in terms of both evaluation metrics, namely ROC-AUC and AP.

The soft-boundary trend analysis brings an important issue that we need to consider in our future work. The soft-boundary

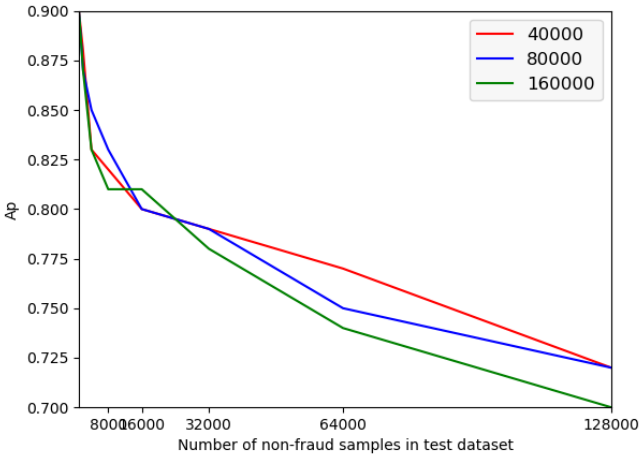

Fig. 2. Trend analysis based in terms of AP for One-class DeepSVDD

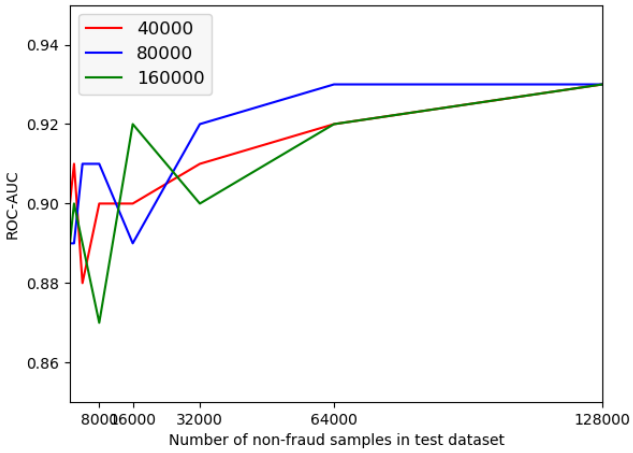

Fig. 3. Trend analysis in terms of ROC-AUC for One-class DeepSVDD

training phase accepts a small portion of fraud samples. Feeding an effective set of fraud samples using sample selection into the model may enhance its performance. To do so, a comprehensive analysis of the behavior of fraud examples is also required.

C. Comparing one-class and soft-boundary DeepSVDD

Here, we compare the performance of two DeepSVDD versions in a case where the test dataset contains 40000 non-fraud samples. As Figures 5 and 6 show, the one-class DeepSVDD outperforms soft-boundary, especially in terms of AP. Because the objective function of the one-class version is designed to focus on the pattern of non-fraud samples for learning the discriminative hypersphere. Therefore, the capability to distinguish fraud from non-fraud samples is higher than soft-boundary.

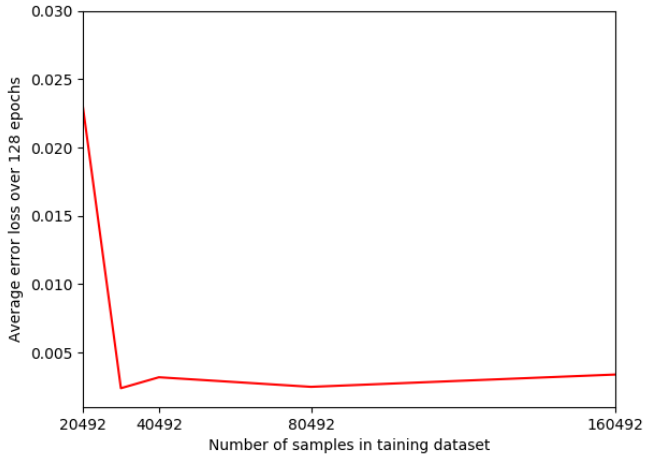

Fig. 4. Training loss

TABLE VI
TREND ANALYSIS IN TERMS OF ROC-AUC METRIC FOR SOFT-BOUNDARY DEEPSVDD

Training samples	Number of non-fraud samples in Test							
	1000	2000	4000	8000	16000	32000	64000	128000
40000	0.89	0.88	0.87	0.87	0.90	0.91	0.91	0.90
80000	0.88	0.85	0.90	0.90	0.88	0.89	0.94	0.91
16000	0.87	0.89	0.89	0.90	0.91	0.90	0.92	0.92

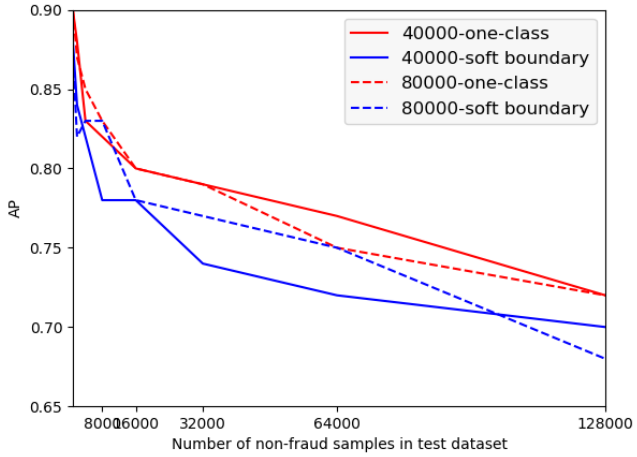

Fig. 5. Comparing two versions in terms of AP

In the second scenario, where the test dataset is more imbalanced because it contains 80000 non-fraud samples and 492 fraud samples, one-class still outperforms the soft-boundary. Therefore, we can conclude that in imbalanced datasets, one-class DeepSVDD is preferred over soft-boundary.
Selecting fraud samples for the training of the soft boundary algorithm is a vital step, and proper fraud samples may enhance the performance of this DeepSVDD version. Therefore, it requires a comprehensive analysis of the fraud examples as an extension of our proposal.

TABLE VII
TREND ANALYSIS IN TERMS OF AP METRIC FOR SOFT-BOUNDARY DEEPSVDD

Training samples	Number of non-fraud samples in Test							
	1000	2000	4000	8000	16000	32000	64000	128000
40000	0.88	0.84	0.82	0.78	0.78	0.74	0.72	0.72
80000	0.87	0.82	0.83	0.83	0.78	0.77	0.75	0.68
16000	0.87	0.84	0.83	0.77	0.76	0.75	0.72	0.67

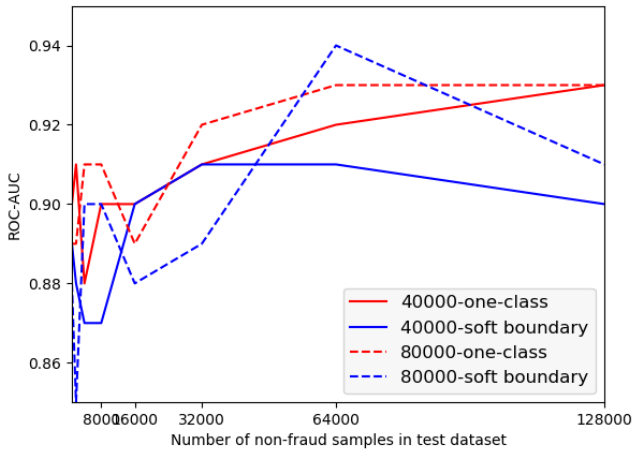

Fig. 6. Comparing two versions in terms of ROC-AUC

D. Comparison with the state of art machine learning algorithms

We examine our proposed framework by comparing it with well-known baseline methods in fraud detection domain, such as Random Forest and SVM as supervised methods. We train Random Forest and SVM on randomly selected 250 fraud samples along with different portions of non-fraud examples. For the test phase, remaining fraud data (242 samples) with a different portion of non-fraud instances are considered. For hyper-parameters of SVM, kernel and gamma values are set RBF and $\frac{1}{D}$, where D is the number of features accordingly. Tables VIII and IX summarize the performance of random forest in terms of ROC-AUC and AP, respectively. Similarly, Tables VIII and IX demonstrate the obtained results for SVM in terms of both evaluation metrics.

TABLE VIII
RANDOM FOREST RESULTS IN TERMS OF ROC-AUC

Training samples	Number of non-fraud samples in Test							
	1000	2000	4000	8000	16000	32000	64000	128000
40000	0.71	0.75	0.76	0.78	0.80	0.78	0.87	0.81
80000	0.73	0.74	0.75	0.72	0.76	0.81	0.78	0.71
160000	0.64	0.64	0.68	0.70	0.72	0.69	0.72	0.75

The obtained AP of SVM and Random Forest models indicate that adding more non-fraud sample does not help these algorithms to detect fraud samples. Because these methods are

TABLE IX
RANDOM FOREST RESULTS IN TERMS OF AP

Training samples	Number of non-fraud samples in Test							
	1000	2000	4000	8000	16000	32000	64000	128000
40000	0.73	0.70	0.68	0.67	0.65	0.65	0.60	0.37
80000	0.73	0.70	0.68	0.67	0.66	0.64	0.63	0.37
160000	0.73	0.70	0.68	0.67	0.66	0.65	0.63	0.4

TABLE X
SVM RESULTS IN TERMS OF ROC-AUC

Training samples	Number of non-fraud samples in Test							
	1000	2000	4000	8000	16000	32000	64000	128000
40000	0.67	0.71	0.75	0.8	0.8	0.86	0.85	0.86
80000	0.58	0.64	0.67	0.78	0.73	0.81	0.86	0.87
160000	0.52	0.55	0.57	0.61	0.70	0.70	0.73	0.79

TABLE XI
SVM RESULTS IN TERMS OF AP

Training samples	Number of non-fraud samples in Test							
	1000	2000	4000	8000	16000	32000	64000	128000
40000	0.73	0.70	0.68	0.67	0.65	0.65	0.64	0.61
80000	0.73	0.70	0.68	0.67	0.66	0.65	0.63	0.61
160000	0.73	0.70	0.68	0.67	0.66	0.65	0.63	0.61

supervised binary classifiers, and they require enough samples from both classes, specifically the minority class, to enhance their performance. Therefore, since fraud detection datasets are highly imbalanced, SVM and RF are not suitable for fraud detection.

As expected, the obtained ROC-AUC results of SVM and RF models higher than our proposed model. As the number of non-fraud samples increases, binary classifiers predict all examples non-fraud, and consequently, ROC-AUC increases because it pays attention to the number of labels that were correctly labeled in total. Therefore, SVM and RF methods are unable to distinguish fraud data from the vast amount of non-fraud samples.

Figures 7 and 8 summarize the comparison between our proposal and the state of art binary classifiers in terms of both evaluation metrics, i.e. AP and ROC-AUC. The obtained results indicate that our proposal outperforms SVM and Random Forest in all different scenarios. For example, in terms of AP, where the test dataset contains 128000 non-fraud samples with all fraud examples for our proposal and 248 fraud instances for the state of art binary classifiers, our approach has acquired 0.72. In contrast, SVM and RF have achieved 0.61 and 0.37, respectively. Our proposed method, particularly one-class DeepSVDD, can learn non-fraud samples' behavior and aims to distinguish fraud examples that possess distant patterns from the training dataset, which only contains non-fraud samples.

V. CONCLUSION AND FUTURE WORK

In this paper, we proposed an efficient framework for fraud detection. Our framework consisted of a novel clustering-based subsampling step, which is followed by applying deep support vector data description for fraud detection. One of

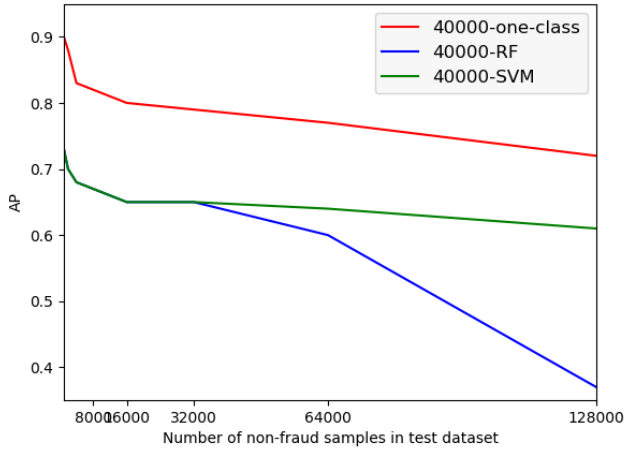

Fig. 7. Comparing with the state of art machine learning algorithms in terms of AP

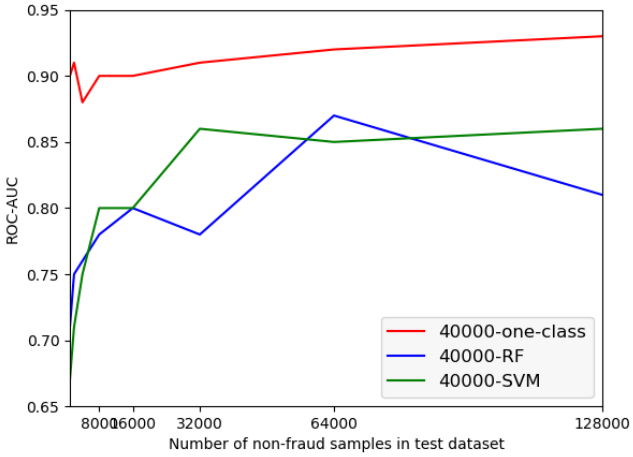

Fig. 8. Comparing with the state of art machine learning algorithms in terms of ROC-AUC

the most challenging issues in fraud datasets is imbalanced datasets. Our proposal can deal with this problem in two ways; proposing the subsampling method to select a subset of non-fraud samples effectively along with training one-class DeepSVDD as an unsupervised one classification method. Our model achieves promising results based on ROC-AUC and AP, a preferred metric in imbalanced situations. We provided a trend analysis based on the size of the test dataset and size of the training dataset for both versions of DeepSVDD and state-of-the-art machine learning classifiers, namely SVM and RF. Deep one-class classifiers outperform the state-of-the-art machine learning classifiers for fraud detection in both evaluation metrics. Fraud datasets are mostly significantly imbalanced and the state-of-the-art classifiers require a certain amount of fraud samples for training. Our proposed method outperforms them for fraud detection because of the benefits of

both non-fraud subsampling and deep one-class classification.

As future work, we would like to conduct a comprehensive analysis of the behavior of fraud data to enhance the performance of our model. Since the dataset we used in our experiments contains information about different anonymous groups of people, it might be useful to design an ensemble of learners where there are different deep one-class classifiers and each one is trained for a specific group of users. Furthermore, we plan to conduct more experiments using other public datasets to compare our proposed approach with unsupervised deep learning approaches such as Restricted Boltzmann Machine and deep autoencoder.

ACKNOWLEDGMENT

The authors graciously acknowledge the funding support from the NSERC Discovery Grant (no. RGPIN 231074) and Tier 1 Canada Research Chair to Dr. Ghorbani, the Canadian Institute for Cybersecurity (CIC), Atlantic Opportunity Agency (ACOA), and Opportunity New Brunswick (ONB), Canada.

REFERENCES

[1] "Canada fraud statistics." https://www.theguardian.pe.ca. Accessed: 2020.

[2] N. Etaher, G. R. Weir, and M. Alazab, "From zeus to zitmo: Trends in banking malware," in *2015 IEEE Trustcom/BigDataSE/ISPA*, vol. 1, pp. 1386–1391, IEEE, 2015.

[3] A. K. I. Hassan and A. Abraham, "Modeling insurance fraud detection using imbalanced data classification," in *Advances in Nature and Biologically Inspired Computing*, pp. 117–127, Springer, 2016.

[4] U. Fiore, A. De 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.

[5] L. Ruff, R. Vandermeulen, N. Goernitz, L. Deecke, S. A. Siddiqui, A. Binder, E. Müller, and M. Kloft, "Deep one-class classification," in *International conference on machine learning*, pp. 4393–4402, 2018.

[6] J. Z. Lei and A. A. Ghorbani, "Improved competitive learning neural networks for network intrusion and fraud detection," *Neurocomputing*, vol. 75, no. 1, pp. 135–145, 2012.

[7] N. Malini and M. Pushpa, "Analysis on credit card fraud identification techniques based on knn and outlier detection," in *2017 Third International Conference on Advances in Electrical, Electronics, Information, Communication and Bio-Informatics (AEEICB)*, pp. 255–258, IEEE, 2017.

[8] Q. Li and Y. Xie, "A behavior-cluster based imbalanced classification method for credit card fraud detection," in *Proceedings of the 2019 2nd International Conference on Data Science and Information Technology*, pp. 134–139, 2019.

[9] Y. Zhang, G. Liu, L. Zheng, and C. Yan, "A hierarchical clustering strategy of processing class imbalance and its application in fraud detection," in *2019 IEEE 21st International Conference on High Performance Computing and Communications; IEEE 17th International Conference on Smart City; IEEE 5th International Conference on Data Science and Systems (HPCC/SmartCity/DSS)*, pp. 1810–1816, IEEE, 2019.

[10] S. Xuan, G. Liu, Z. Li, L. Zheng, S. Wang, and C. Jiang, "Random forest for credit card fraud detection," in *2018 IEEE 15th International Conference on Networking, Sensing and Control (ICNSC)*, pp. 1–6, IEEE, 2018.

[11] N. Yousefi, M. Alaghband, and I. Garibay, "A comprehensive survey on machine learning techniques and user authentication approaches for credit card fraud detection," *arXiv preprint arXiv:1912.02629*, 2019.

[12] F. Carcillo, Y.-A. Le Borgne, O. Caelen, and G. Bontempi, "Streaming active learning strategies for real-life credit card fraud detection: assessment and visualization," *International Journal of Data Science and Analytics*, vol. 5, no. 4, pp. 285–300, 2018.

[13] M. Khedmati, M. Erfani, and M. GhasemiGol, "Applying support vector data description for fraud detection," 2020.

[14] Y. Lucas, P.-E. Portier, L. Laporte, L. He-Guelton, O. Caelen, M. Granitzer, and S. Calabretto, "Towards automated feature engineering for credit card fraud detection using multi-perspective hmms," *Future Generation Computer Systems*, vol. 102, pp. 393–402, 2020.

[15] Z. Zhang, X. Zhou, X. Zhang, L. Wang, and P. Wang, "A model based on convolutional neural network for online transaction fraud detection," *Security and Communication Networks*, vol. 2018, 2018.

[16] K. Fu, D. Cheng, Y. Tu, and L. Zhang, "Credit card fraud detection using convolutional neural networks," in *International Conference on Neural Information Processing*, pp. 483–490, Springer, 2016.

[17] S. P. RM, P. K. R. Maddikunta, M. Parimala, S. Koppu, T. Reddy, C. L. Chowdhary, and M. Alazab, "An effective feature engineering for dnn using hybrid pca-gwo for intrusion detection in iomt architecture," *Computer Communications*, 2020.

[18] J. Jurgovsky, M. Granitzer, K. Ziegler, S. Calabretto, P.-E. Portier, L. He-Guelton, and O. Caelen, "Sequence classification for credit-card fraud detection," *Expert Systems with Applications*, vol. 100, pp. 234–245, 2018.

[19] A. Roy, J. Sun, R. Mahoney, L. Alonzi, S. Adams, and P. Beling, "Deep learning detecting fraud in credit card transactions," in *2018 Systems and Information Engineering Design Symposium (SIEDS)*, pp. 129–134, IEEE, 2018.

[20] A. Pumsirirat and L. Yan, "Credit card fraud detection using deep learning based on auto-encoder and restricted boltzmann machine," *International Journal of advanced computer science and applications*, vol. 9, no. 1, pp. 18–25, 2018.