“© 2019 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works.”

Learning Latent Distribution for Distinguishing Network Traffic in Intrusion Detection System

Ly Vu[1],Van Loi Cao[1], Quang Uy Nguyen[1], Diep N. Nguyen[2], Dinh Thai Hoang[2], Eryk Dutkiewicz[2]
[1] Le Quy Don Technical University, Hanoi, Vietnam
[2] School of Electrical and Data Engineering, University of Technology Sydney, Australia

***Abstract*—In this paper, we develop a new deep learning approach, Multi-distributed Variational AutoEncoder (MVAE), to enhance network intrusion detection. MVAE introduces label information of data samples into the loss function of VAE. This label information together with reconstruction error function of VAE will force each class of network data into a different region in the latent feature space of MVAE. As a result, the network traffic samples are more distinguishable in the new representation space, thereby improving the accuracy in detecting intrusions for classifiers in the latent feature space of MVAE. To evaluate the efficiency of the proposed solution, we carry out intensive experiments on two popular network intrusion datasets, i.e., NSL-KDD and UNSW-NB15 under four conventional classifiers including Gaussian Naive Bayes (GNB), Support Vector Machine (SVM), Decision Tree (DT), and Random Forest (RF). The experimental results demonstrate that our proposed approach can significantly improve the accuracy of intrusion detection algorithms up to 0.246 compared to the original one.**

Keywords- Deep learning, Variational AutoEncoder (VAE), Intrusion Detection system (IDS).

I. Introduction

Over the last five years, we have been experiencing an explosion in communications and information technology in the Internet-of-Things era. Cisco predicted that the Global IP traffic will increase nearly threefold over the next 5 years, and will have increased 127-fold from 2005 to 2021 [1]. Furthermore, IP traffic will grow at a Compound Annual Growth Rate of 24% from 2016 to 2021. The unprecedented development of communication networks has great contributions for human beings, but also places many challenges for information security problems due to diversity of emerging cyberattacks. According to the US government statistics, the number of ransomware attacks increased 300% from 2015, with over 4,000 attacks detected per day in 2016 [2]. As a result, Intrusion Detection Systems (IDSs) have been playing a crucial role in preventing cyberattacks and ensuring confidentiality, integrity, and availability (CIA) of information in communication networks [3], [4], [5].

An IDS monitors the network traffic to identify abnormal activities. There are three popular approaches for analyzing the network traffic to detect intrusive behaviors [6], i.e., statistical based, machine learning based, and knowledge-based methods. Among these, machine learning-based methods have received a great attention and achieved remarkable success [5], [7], [8]. Recently, deep learning have been applied to improve the effectiveness of machine learning in IDSs. There are three common structures of deep learning: Autoencoders (AEs) [9], Convolution Neural Networks (CNNs) [10] and Recurrent Neural Networks (RNNs) [11]. Among these, AEs are the networks that can represent input data into lower dimension and less complex feature space in their bottleneck layer (latent feature space) [9].

An extension of AEs is Variational AutoEncoder (VAE) [12]. The VAE can force the input data in a new representation feature space called as bottleneck layer (latent feature space) into a well-shape, Gaussian distribution [12]. Thus, VAEs not only inherit good characteristics from AEs, such as projecting input data in lower dimension, and discovering more relevant features, but also possess a new advanced characteristic, i.e., forming the latent feature space into Gaussian distribution. These characteristics make the latent feature space of VAEs more powerful to represent input data than ordinary AEs, thereby improving the accuracy in identifying anomaly for machine learning-based methods [3], [13]. However, current VAE approaches only force data into one Gaussian Distribution, and thus data instances from different classes are mixed into the same distribution that is difficult for classification algorithms. In this paper, we propose Multi-distributed Variational AutoEncoder (MVAE) that allows to project the network traffic data into the latent feature space in which each traffic class will be resided in a separated area. We then perform intensive simulations on IDS datasets, i.e., NSL-KDD and UNSW-NB15,using four conventional classifiers including Gaussian Naive Bayes (GNB), Support Vector Machine (SVM), Decision Tree (DT), and Random Forest (RF), to verify the efficiency of the proposed solution in terms of reliability and robustness. The main contributions and novelty of the paper include:

- We propose a new deep learning model, namely MVAE, that incorporates label information into the its loss function to impose the input data to an appropriate latent feature space, thereby sampling classes more distinguishable.
- We evaluate the proposed solution by performing intensive simulations on two well-known IDS datasets under a number of classifiers on the latent feature space of MVAE.

The rest of paper is organized as follows. Section II briefly reviews related works on IDS. Section III presents the fundamental background of VAE. The proposed method is then described in Section IV. The experimental settings are provided in Section V. After that Section VI presents experimental results together with analysis. Conclusions and future works are discussed in Section VII.

II. Related Work

Machine learning algorithms for network intrusion detection have received an increasing attention in the research community due to its outstanding advantages [7], [8]. The main idea of applying machine learning techniques for IDSs is to automatically build a detection model based on training datasets. Typically, machine learning for IDSs can be divided into two categories, i.e., single and hybrid methods. The single method attempts to use only one machine learning technique such as SVM, RF, DT [5], [8], to find appropriate models which are used to classify or recognize whether the incoming requests are normal or malicious. Meanwhile, the hybrid methods aim to incorporate several learning techniques to enhance the performance of the IDSs. There are some ways in which different algorithms can be hybridized. The hybrid classifiers can be built based on cascading different classifiers. For example, the authors in [7] propose a two stage hybrid classification method using an SVM as anomaly detection in the first stage, and artificial neural network as misuse detection in the second stage to enhance accuracy for IDSs. Other methods are based on re-sampling data samples and taking a majority vote of the resulting weak learners [17].

Recently, deep learning has been emerging as a breakthrough technology to extract meaningful features from intrusion datasets, thereby enhancing the accuracy of IDSs [18], [19]. In particular, Salama et al. [18] introduce a method for anomaly intrusion detection using Restricted Bolzman Machine-based Deep Believe Network (DBN). In this work, DBN is used as a feature reduction method and SVM is utilized as a classifier. Kim et al. [19] demonstrate that long short term memory can be used in Recurrent Neural Network to improve the effectiveness for IDSs.

Some recent research works have introduced new ideas of incorporating deep neural networks with VAE to represent the data samples. In particular, the authors in [14], [15] propose using label information in VAE model, namely Conditional VAE (CVAE), to address problems in image captioning. In these works, the label information is concatenated with input samples as the input of the encoder network. Moreover, the input of the decoder network is the concatenation of a latent variable and a label information. However, this model only can be used to synthesize the data samples with specific labels. In other words, it cannot be applied to a classification problem due to using the label information as an input. Therefore, the authors in [16] propose an extension of VAE model, called Intrusion Detection-Conditional Variational AutoEncoder (ID-CVAE) for IDSs. The ID-CVAE model combines the label information and the input of the decoder network to reduce the RE of the VAE network. Then, this model classifies a new data sample by adding all class labels to reconstruct samples associating with each class label. However, this model uses Euclide distance to measure the similarity of reconstructed samples and original samples, yielding to a low accuracy with high dimensional data.

Different from other deep learning approaches based on the VAE structure, we propose a new deep learning model by incorporating the label information into the KL loss term of optimizing VAE model. This approach will impose the input samples to a new representation space which is lower dimensions and more distinguishable than the input sample space. The new representation space can improve the performance of classification algorithms applied to IDSs.

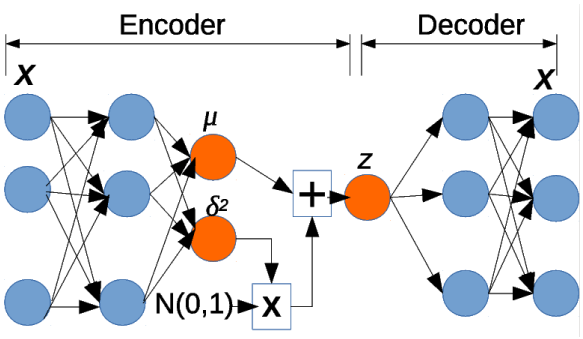


Fig. 1. Illustrations of VAE model.

III. Background

In this section, we describe the structure and the loss functions of VAE [12] and CVAE [14], [15]. They are the fundamentals of our proposed model presented in section IV.

A VAE includes two neural networks, i.e., encoder and decoder as described in Fig. 1. The encoder network approximates $q_\phi(z|x)$, which represents the Gaussian distribution, to project data x into the latent variable z. The decoder approximates $p_\theta(x|z)$ which represents the original data distribution given the latent variable z. As presented in [12], the goal of training VAE is to optimize a variational lower bound on the marginal likelihood of data with respect to variational parameters θ and generative parameters ϕ. The variational lower bound is written as follows:

$$\begin{aligned}\log p_\theta(x) &= KL(q_\phi(z|x)||p_\theta(z|x)) + \\ &\quad \mathbb{E}_{q_\phi(z|x)}[-\log q_\phi(z|x) + \log p_\theta(x,z)], \\ &\geq -KL(q_\phi(z|x)||p_\theta(z)) + \mathbb{E}_{q_\phi(z|x)}[\log p_\theta(x|z)].\end{aligned} \tag{1}$$

In VAE, $q_\phi(z|x)$ is an approximation of the true posterior distribution $p_\theta(z|x)$. Assuming that the distribution of latent variables is a Gaussian distribution, the first term of (1) can be marginalized. However, the second term of this equation needs to approximate by drawing samples z_k with $k = (1, \ldots, K)$ generated by $q_\phi(z|x)$. Thus, the objective function of VAE can be written as follows:

$$\ell_{VAE}(x, \theta, \phi) = -KL(q_\phi(z|x)||p_\theta(z)) + \frac{1}{K}\sum_{k=1}^{K}\log p_\theta(x|z). \tag{2}$$

Generally, encoder part tries to generate the latent variable z which is close to the probability distribution of the latent variable, i.e., the Gausian distribution. This is presented by minimizing the KL loss term, $KL(q_\phi(z|x)||p_\theta(z))$. Furthermore, the decoder part is learned to reconstruct input data x from the latent variable z, which is represented by maximizing $E_{q_\phi(z|x)}[\log p_\theta(x|z)]$. Then, to keep the label information of

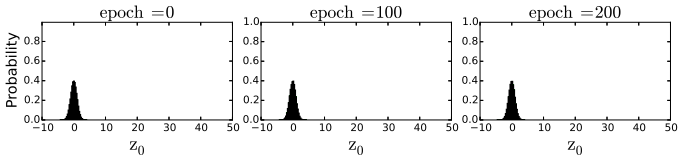


Fig. 2. Histogram of latent variable z_0 in training CVAE [15].

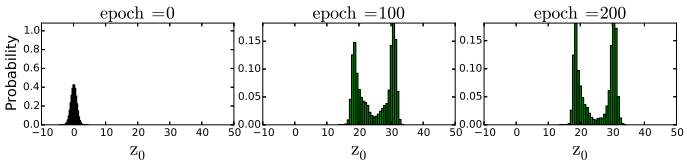


Fig. 3. Histogram of latent variable z_0 in training MVAE.

data samples in the latent space, several research works [15], [16] are developed from VAE, called CVAE and ID-CVAE. ID-CVAE only conditions the label information in the decoder part, while CVAE usually conditions both the encoder and decoder to the class label c as in (3). Therefore, the encoder and decoder are conditioned to two variables, i.e., $q_\phi(z|x,c))$ and $p_\theta(z|x,c)$, respectively. As a result, the objective function of CVAE can be written as follows:

$$\ell_{CVAE}(x,\theta,\phi) = -KL(q_\phi(z|x,c)||p_\theta(z)) + \frac{1}{K}\sum_{k=1}^{K}\log p_\theta(x|z,c). \tag{3}$$

where c as the label of the data sample x.

IV. Proposed method

This section proposes a new extension of VAE, namely MVAE. MVAE is different from VAE is that MVAE incorporates a new regularizer term in the original VAE loss function. The new regularizer term, called Multi-KL, and reconstruction loss are used to make a robust representation of the original data. The regularizer condenses the original data close to the normal distributions as described in Fig. 2, while the reconstruction loss keeps important information of original data in order to reconstruct the original data at the output layer. Then, in the bottleneck layer (latent feature space) of MVAE, Multi-KL will condense each class sample as close as possible to each separated region.

In order to distinguish the sample classes in the latent feature space, we calculate the latent error by the KL divergence of the latent variable and the distributions which have mean values defined by the class labels. As presented in [20], the KL divergence is a directed divergence between two distributions. It simply implies how much one distribution diverges from another. Consequently, if we have two distributions p and q, a smaller value of the KL divergence implies more similarity/less divergence and vice versa. Thus, each class of samples will be represented by a separated Gaussian Distribution $\mathcal{N}(\mu_c, \sigma^2)$ with a mean μ_c, and a standard deviation σ. The mean values of these Gaussian Distributions are different from each others. These will help to impose class samples to the separate distributions. The Gaussian Distributions, $p_\theta(z|c)$, are defined as follows:

$$p_\theta(z|c) = \frac{1}{\sigma\sqrt{2\pi}} e^{-(z-\mu_c)^2/2\sigma^2}. \tag{4}$$

After that, the objective function of MVAE can be written by:

$$\ell_{MVAE}(x,\theta,\phi) = -KL(q_\phi(z|x)||p_\theta(z|c)) + \frac{1}{K}\sum_{k=1}^{K}\log p_\theta(x|z,c). \tag{5}$$

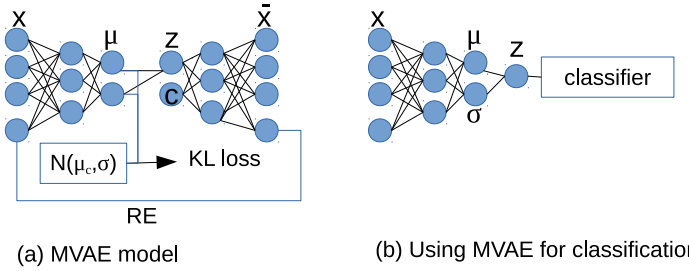
Fig. 3 visualizes the histogram of the first dimension of the latent variable z_0 of training NSL-KDD dataset. As described in Fig. 3, after a number of training epochs, the latent data is transformed from one Gaussian Distribution originally to separated distributions (two Gaussian distributions) with different mean values. Comparing to the histogram of the latent variable z_0 in the original CVAE [15], it can be observed that the KL loss term of CVAE only helps to downsize the distribution tail to the origin. On the other hand, our proposed regularizer encourages them to distinguish two separate distributions according to two class labels. Thus, the new regularizer can impose the original data samples with one distribution to a new representation space. In this representation space, the new data samples can be separated by different distributions. This helps to improve the classifiers which are used to classify the data samples as a specific group of attacks or normal.

The MVAE model is used to represent input data as a different feature space. This new representation space has lower dimensions, higher representative, and more distinguishable class labels. Fig. 4(a) presents the proposed MVAE model. In this figure, c and μ_c are the label and mean value of the distribution respectively. Compared to the previous CVAE models, we change the KL loss term by measuring the latent distribution and the normal distribution defined by μ_c. Thus, the KL loss term forces each class of samples to the different normal distribution with the mean value μ_c

In Fig. 4(b), we present how to use MVAE model for classification. First, MVAE model imposes input data x to latent variable z. Second, classifiers such as GNB, SVM, DT, and RF are applied on the latent variable z to identify the normal or attack data. As shown in this figure, the new representation space has lower dimensions. Moreover, it is more distinguishable as displayed in Fig. 3. As a result, the classification algorithms applied on this new representation can enhance the accurate in detecting intrusion.

V. Experimental Settings

This section presents the datasets and the experimental settings used in this paper.



(a) MVAE model (b) Using MVAE for classification

Fig. 4. MVAE model and application in classification.

A. Datasets

In order to test the effectiveness of the proposed method we use two well-known network traffic datasets, i.e., NSL-KDD and UNSW-NB15.

1) NSL-KDD: NSL-KDD is an IDS dataset [21] which is used to solve some intrinsic problems of the KDD'99 dataset. The NSL-KDD dataset contains 148,517 records in total, which is divided into the training set (125,973 data samples) and the testing set (22,544 data samples). Each sample has 41 features and is labeled as either a type of attacks or normal. The training set contains 24 specific attack types, and the testing set has 14 new types of attacks that are not presented in the training set. The simulated attack samples belong to one of four categories, i.e., DOS, R2L, U2R, and Probing. Three categorical features,i.e., *protocol* type, *service*, and *flag*, are preprocessed by one-hot-encoding which increases the number of features to 122

2) UNSW-NB15: The dataset is created by utilizing the synthetic environment as the IXIA PerfectStorm tool in the Cyber Range Lab of the Australian Centre of Cyber Security (ACCS) [22]. The number of records in the training set and the testing set are 175,341 records and 82,332 records, respectively. There are nine categories of attacks in UNSW-NB15, which are Fuzzers, Analysis, Backdoors, DoS, Exploits, Generic, Reconnaissance, Shellcode and Worms. Each data sample has 49 features which have been generated by using the Argus, Bro-IDS tools and twelve their algorithms to analyze characteristics of network packets. The data samples are labeled either 0 for normal or 1 indicating a type of attack. The categorical attributes, such as *protocol*, *service*, and *state*, are preprocessed by one-hot encoding which increases the number of features to 196.

B. Experimental Settings

In order to demonstrate the effectiveness of the latent representation of MVAE in facilitating classification-based IDSs, we compare the latent representation with those from AE, VAE and ID-CVAE. We choose the same structure for these AEs. Each model has totally five hidden layers. We use RELU activation function for all hidden layers excepts the last layer of the encoder (Linear) and decoder(Sigmoid) networks. Adam optimization [23] with learning rate $1e-6$ is used to train these deep learning models.

Four popular classification algorithms are used to evaluate these latent representations from MVAE, AE, VAE and ID-CVAE in detecting network intrusions. We utilize the implementation of these classifiers in the popular machine learning packet in Python, Scikit learn [24]. In order to lessen the impact of experimental parameters to the performance of the classifiers, we used the grid search technique for each algorithm except GNB (no hyper-parameters). Table I shows the range of values that are used for tuning the hyper-parameters of SVM, DT and RF.

TABLE I
PARAMETER RANGES OF THE GRID SEARCH FOR CLASSIFIERS

Classifiers	Parameters
SVM	$kernel = rbf; gama = 0.001, 0.01, 0.1, 1.0$
DT	$max-depth = 5, 6, 7, 8, 9, 10, 50, 100$
RF	$n-estimators = 20, 40, 80, 150$

C. Evaluation Methods

First, we define some values representing the performance of a classification algorithm as follows:

$$TPR = \frac{TP}{TP + FN}. \tag{6}$$

$$FPR = \frac{FP}{TN + FP}. \tag{7}$$

Here, TPR is the proportion of real positive samples that are correctly predicted positive and FPR is the proportion of incorrect predicted positive samples.

The Area Under ROC Curve (AUC) [25] which is often created by plotting the true positive rate (sensitivity) against the false positive rate (specificity) at various threshold settings. The ROC analysis plots the rate TPR against the rate FPR. A perfect classifier will score in the top left hand corner ($FPR = 0$, $TPR = 100\%$). A worst case classifier will score in the bottom right hand corner ($FPR = 100\%$, $TPR = 0$). The space under ROC curve is represented as AUC score. This measures the average quality of a classification model at different thresholds. The optimum in practice is the area under the simple trapezoid defined by the model as 8. A random classifier has the AUC value of 0.5 and the value of the AUC score for a perfect classifier is 1.0. Therefore, most classifiers have the value of the AUC score between 0.5 and 1.0.

$$AUC = \frac{TPR - FPR + 1}{2}. \tag{8}$$

VI. Simulation Results and Discussions

To observe the effects of our proposed method, we visualize the distribution of the latent data produced from ID-CVAE and MVAE when they are trained on the NSL-KDD dataset. Both these models also add the label information into training process. We set up the dimension of the latent variables as 2 to display them in a 2-D coordinate system. There are two classes in the training dataset including normal (blue dots) and attack (red dots) data samples. As displayed in Fig. 5, ID-CVAE can represent the input samples into a lower latent

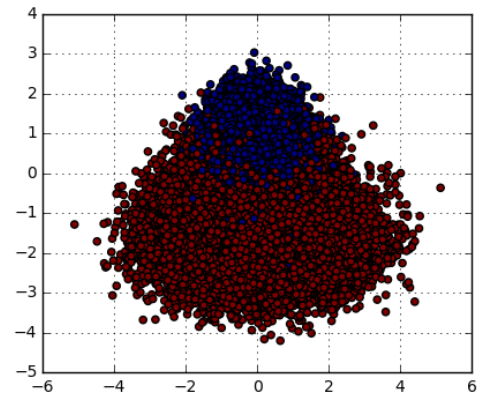


Fig. 5. Distribution of latent variable z in ID-CVAE model.

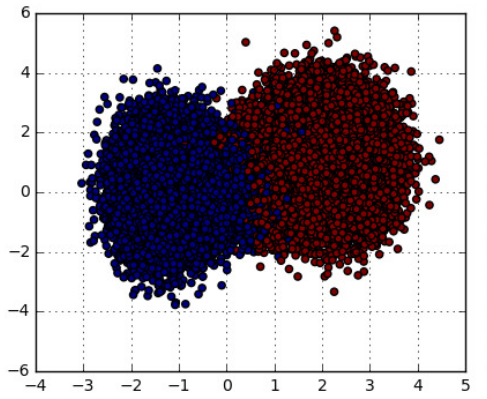


Fig. 6. Distribution of latent variable z in MVAE model.

feature space. However, the distribution of the latent data can not be distinguished into separated classes as showed in Fig. 5. In addition, our proposed model, MVAE can impose the input data samples to the separate distributions as shown in Fig. 6. The reason is that adding the class information to the Multi-KL loss term of MVAE helps to project the input samples to the separate areas based on their classes. Thus, our proposed model can represent the network traffic samples in a latent variable space, but also make class samples more distinguishable, thereby improving the performance of classification algorithms applied to detect network intrusion.

Fig. 7 presents the the values of the loss function in training MVAE. In this figure, the total loss is the summary of the KL loss and the RE. The KL loss and the RE decrease quickly after a number of epochs, leading to the reduction of the total loss value. Reducing of the total loss value after in training process evidences that MVAE can convergence with the new regularizer term.

Table 8 presents the AUC score of classification algorithms such as GNB, SVM, DT, and RF applied on the original data samples (ORIGINAL) and the latent variable spaces of AE, VAE, ID-CVAE and MVAE, respectively. This aims to

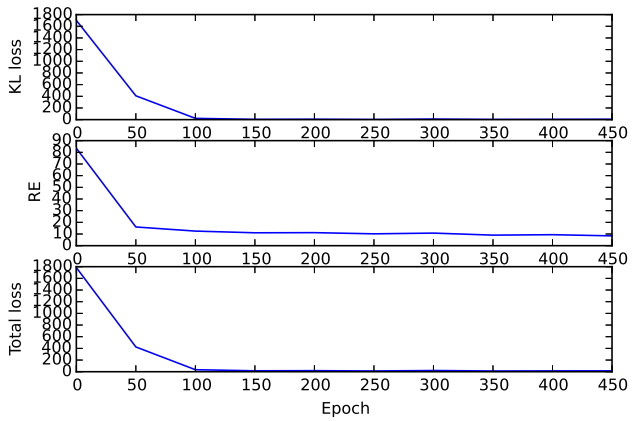


Fig. 7. Errors in training process of MVAE model.

compare the effectiveness of different latent representations produced from these AEs and our proposed model. The table shows the significance of our proposed method in providing the discriminative representation. It can be seen that the AUC score of the classification algorithms is improved considerably when training on the representation space generated by MVAE in comparison to those generated by other techniques, i.e., AE, VAE, and ID-CVAE. The AUC scores of GNB, SVM, DT, and RF on the NSL-KDD dataset with MVAE are increased 0.100, 0.121, 0.076, and 0.104, respectively, compared to other techniques. These values on UNSW-NB15 dataset are 0.223, 0.253, 0.242, and 0.246 respectively. Specificity, our proposed model improves AUC of GNB, SVM, DT, and RF classifiers on NSL-KDD dataset by 0.100, 0.121, 0.170, and 0.143, respectively, compared with the original VAE. These values on UNSW-NB15 dataset are 0.223, 0.253, 0.242, and 0.246, respectively.

Moreover, it can be observed that VAEs such as VAE and ID-CVAE produce relatively poor performance. The reason is that the latent data of VAEs is already in a good shape before training (see Fig. 2 at epoch 0). Thus, VAEs has less influence on learning the representation. Consequently, most of the representation power of the VAE may be used for reconstruction.

These results evidence for the benefit of our novel proposed technique. In other words, using our proposed model, MVAE, imposed the network traffic samples to more distinguishable representation feature space based on the label classes as in Fig. 6. Subsequently, it improves the effectiveness of classification algorithms when they are applied to the new representation space. This can be explained by adding label information into KL loss term of optimizing VAE model. Consequently, each input class sample is projected to the distribution that is associated with its label information. As a result, input data samples are more distinguishable in the new representation feature space, leading to the great performance improvement for classification algorithms.

In overall, the experimental results show that our proposed model can represent the network traffic samples into a new latent feature space in which class samples are more separable.

TABLE II
AUC SCORE OF ATTACK DETECTION BASED ON TWO-CLASSES CLASSIFICATION.

Dataset	**NSL-KDD**				**UNSW-NB15**			
Model	**GaussianNB**	**SVM**	**DT**	**RF**	**GaussianNB**	**SVM**	**DT**	**RF**
ORIGINAL	0.617	0.820	0.886	0.839	0.716	0.886	0.879	0.889
AE	0.792	0.801	0.802	0.831	0.721	0.842	0.881	0.90
VAE	0.824	0.824	0.792	0.800	0.705	0.692	0.712	0.715
ID-CVAE	0.724	0.717	0.689	0.721	0.696	0.662	0.702	0.712
MVAE	**0.924**	**0.945**	**0.962**	**0.943**	**0.928**	**0.945**	**0.954**	**0.961**

The classification algorithms used on the latent representation of MVAE can remarkably outperform those used on the original data and the latent representations generated from other models such as AE, VAE and ID-CVAE.

VII. SUMMARY

In this paper, we have developed MVAE, the novel model of VAE, by proposing the new regularizer term to the VAE model to improve the performance of classification-based IDS. Specifically, the proposed MVAE can impose the network traffic data samples to a new representation space which can separate the original data samples in different distributions. As a result, the distinguishable representation can help to increase the performance of machine learning algorithms for detecting network intrusions. The intensive experiments are then conducted on two well-known IDS datasets, i.e., NSL-KDD and UNSW-NB15, and show that our proposed technique can improve the accuracy of machine learning algorithms such as GNB, SVM, DT, and RF compared with other techniques. Alternatively, we also visualize the results of representation space of MVAE to see the effectiveness of new representation space generated by our proposed method. However, it can be seen in the visualization of new representation space that even though the new representation of MVAE is more distinguishable for sample classes, it is still overlapped together. Thus, our future work will focus partly on minimizing the distribution space for each class to completely separate the sample classes in the representation space. Thus, this can classify network data samples more accurately for IDSs.

REFERENCES

[1] Cisco Visual Networking Index: Forecast and Methodology, 2016-2021.

[2] Ransomware attacks increase 300% in 2016, Business Insights, 2017.

[3] V. L. Cao, M. Nicolau, and J. McDermott, "Learning Neural Representations for Network Anomaly Detection," *IEEE Transactions on Cybernetics*, pp 1-14, Jun. 2018.

[4] A. Attiah, M. Chatterjeey, and C. C. Zou, "A Game Theoretic Approach to Model Cyber Attack and Defense Strategies," *IEEE ICC*, pp. 1-7, Kansas City, USA, May 2018.

[5] H. Hindy, D. Brosset, E. Bayne, A. Seeam, C. Tachtatzis, R. Atkinson, and X. Bellekens, "A Taxonomy and Survey of Intrusion Detection System Design Techniques, Network Threats and Datasets," *arXiv:1806.03517v1*. 2018.

[6] X. Jing, Z. Yan, and W. Pedrycz, "Security Data Collection and Data Analytics in the Internet: A Survey, " *IEEE Communications Surveys & Tutorials*. DOI: 10.1109/COMST.2018.2863942.

[7] J. Hussain, S. Lalmuanawma, and L. Chhakchhuak, "A two-stage hybrid classification technique for network intrusion detection system," *International Journal of Computational Intelligence Systems*, vol. 9, no. 5, pp. 863-875, Sep. 2016.

[8] A. S. Eesa, Z. Orman, and A. M. A. Brifcani, "A novel feature-selection approach based on the cuttlefish optimization algorithm for intrusion detection systems," *Expert Systems with Applications*, vol. 42, no. 5, pp. 2670-2679 Apr. 2015.

[9] P. Baldi, Autoencoders. "Unsupervised Learning, and Deep Architectures," *Journal of Machine Learning Research: Workshop and Conference Proceedings* , vol. 27, pp. 37-50, 2012.

[10] Y. Lecun, L. Boottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceeding of the IEEE*, 1998.

[11] Y. Bengio, S. Patrice, and F. Paolo, "Learning long-term dependencies with gradient descent is difficult," *IEEE Transactions on Neural Networks*, vol. 5, no. 2, pp. 157-166, 1994.

[12] D. P. Kingma and M. Welling, "Auto-encoding variational bayes," *Advances in Neural Information Processing Systems*, 2014.

[13] V. L. Cao, M. Nicolau, and J. McDermott, "One-class classification for anomaly detection with dernel density estimation and genetic programming," *European Conference on Genetic Programming*, pp. 3-18, Mar. 2016.

[14] Y. Pu, Z. Gan, R. Henao, X. Yuan, C. Li, A. Stevens, and L. Carin. "Variational autoencoder for deep learning of images, labels and captions," in *Advances in Neural Information Processing Systems Proceedings*, pp. 2352-2360, 2016.

[15] K. Sohny, X. Yany, and H. Lee, "Learning structured output representation using deep conditional generative models," in *Advances in Neural Information Processing Systems Proceedings*, pp. 3483-3491, 2015.

[16] G. Dorta, S. Vicente, Y. Wang, W. Zhou, Y. Xiang, and Y. Guan, "Laplacian pyramid of conditional variational autoencoders," in *CVPR*, 2017.

[17] X. Y. Liu, J. Wu, and Z. H. Zhou, "Exploratory undersampling for class learning," *IEEE Transactions on System, Man, and Cybernetics, Part B (Cybernetics)*, vol. 39, no. 2, pp. 539-550, 2009.

[18] M. A. Salama, H. F. Eid, R. A. Ramadan, A. Darwish, and A. E. Hassanien, "Hybrid intelligent intrusion detection scheme," *Soft Computer Industry Application 96*, pp. 293303, 2011

[19] J. Kim, J. Kim, H. L. T. Thu, and H. Kim, "Long short term memory recurrent neural network classifier for intrusion detection," *International Conference on Platform Technology and Service*, pp. 1-5, 2016.

[20] S. M. Danish, A. Nasir, H. K. Qureshi, A. B. Ashfaq, S. Mumtazy, and J. Rodriguez, "Network Intrusion Detection System for Jamming Attack in LoRaWAN join procedure," in *ICC*, 2018.

[21] NSL-KDD dataset. [Online] Available: http://nsl.cs.unb.ca/NSL-KDD/ [Accessed: 10- Apr- 2018].

[22] N. Moustafa and J. Slay, "UNSW-NB15: A comprehensive data set for network intrusion detection systems (UNSW-NB15 network data set)," *Military Communications and Information Systems Conference*, 2015.

[23] D. P. Kingma & J. Ba. Adam: A Method for Stochastic Optimization. *International Conference on Learning Representations*. arXiv preprint arXiv:1412.6980. 2015.

[24] Sklearn tutorial. [Online] Available: http://scikit-learn.org/stable/.

[25] D. M. W. Powers, "Evaluation: from precision, recall and F-measure to ROC, informedness, markedness and correlation," *Journal of Machine learning Technologies*, vol. 2, pp. 37-63, 2011.