INVESTIGATIONS OF AUTOENCODER HYPER-PARAMETERS ON ANOMALY DETECTION

Van Loi Cao[1], *Huu Noi Nguyen*[1], *Van Quan Nguyen*[1], *Viet Hung Nguyen*[1], *Van Thang Cao*[2]

Abstract

Most of anomaly detection techniques, such as density-based methods, often perform inefficiently on the high dimension of network data because *the curse of dimensionality phenomenon*. Our previous work presented a novel approach of using the feature space of AutoEncoders (AEs) as a new feature representation for addressing this problem. In this study, we attempt to investigate the characteristics of the latent representation of AEs. Thus, we first discuss the hypothesis of using the latent representation in more details, and extend several experiments showed in the previous work. Following this, we design three intensive examinations (an investigation on the middle hidden layer size, an evaluation on the performance of the hybrid and an exploration on latent data). These aim to get insight into the latent representations of AEs, which is fundamental for designing good latent representations in the future work. This paper closes with analysis and discussion on the experimental results.

Index terms

Anomaly detection, autoEncoders, latent representation, hyper-parameters.

1. Introduction

Most well-known anomaly detection methods, such as density/distance-based methods and one-class SVMs, tend to perform inefficiently in high feature spaces because *"the curse of dimensionality phenomenon"* [1], [2], [3], [4], [5]. This phenomenon results in a high proportion of *inappropriate* and *redundant* attributes concealing true anomalies, and the *concentration of distances*. As discussed in [6], [4], AE-based one-class classifications (OCCs) are very powerful for network anomaly detection. An AE with a bottleneck layer can learn to reproduce its the original input data at the last layer. The AE-based model learnt from normal data will represent properly normal instances, but poorly reconstruct anomalous data and produce large reconstruction errors (REs). REs have been typically used as "anomaly score", query points whose REs above a pre-determined threshold indicate anomalies, as presented in [6], [7], [8], [9]. This suggests a hypothesis that if normal and anomalous instances can be distinguished by

[1] Faculty of Information Technology, Le Quy Don Technical University, Hanoi, Vietnam.
[2] Telecommunications University, Nhatrang, Vietnam.

REs in the output layer of an AE, they should be separated in the compressed latent representation of the AE [10]. This means that the trained AE may allocate some areas (called *normal areas*) in its hidden feature space for normal data, and anomalies that deviate significantly from normal data will appear in other regions. If normal data in the hidden feature space is modeled by density-based learners, the normal regions tend to have high density, and anomalies may fall into low-density regions. Therefore, there is potential to apply density-based anomaly detection techniques on the hidden feature space to avoid the curse of dimensionality phenomenon.

Furthermore, in [2] one-class SVMs were built on the top of a Deep Belief Network (DBN) for dealing with the problem of identifying anomalies in high-dimension. In this hybrid, the performance of one-class SVMs improved significantly, it produced comparable or better performance (both classification accuracy and computational complexity) to stand-alone AEs and one-class SVMs. Based on their experimental results, they concluded that DBNs were useful as a feature reduction technique. Alternatively, [6] demonstrated that AEs with a narrow middle hidden layer will force it to compress redundant information whereas preserve and differentiate non-redundancies of input data in the hidden layer. This suggests that AEs can learn relevant and robust features in their latent representation. Based on these, we suppose that AEs can map the original input data into a lower feature space in which relevant and robust features are discovered, and redundancies are compressed. Therefore, it is desirable to investigate the characteristics of the latent data (called latent vectors) in the hidden feature space of AEs, and the behavior of anomaly detection methods on the latent feature space. This aims to combine the different advantages from AEs and density-based techniques to leverage the performance of network anomaly detection models.

In our previous work [10], the latent representation of AEs was used to leverage density-based anomaly detection performing well on high-dimensional network data. The evaluation on the performance of hybrid between AEs and density-based anomaly detection methods demonstrated that using the latent representation is very potential for addressing high-dimensional anomaly detection. However, the latent representation is often very sensitive to the hyper-parameters of AEs. This will result in the sensitivity on the performance of the subsequent learning density-base techniques. The investigations of the AE hyper-parameters were discussed in [10] but it was not comprehensive study.

In this work, we extend the study in [10] by further examining the characteristics of the bottleneck layer of AEs. This aims to get insight into the latent representation of AEs, which is fundamental to design better latent representations in future work. We first investigate the influence of the middle hidden layer size on the performance of the hybrid AE and density-based techniques. The latent representation is examined with different steepnesses of the activation function. Finally, the hybrid AEs and density-based techniques are assessed on the network security problems to illustrate its performance of a set of hyper-parameters.

The rest of the paper is organized as follows. Section 2 describes our approach using the latent representation of AEs for facilitating density-based anomaly detection

algorithms. Experiment setup, investigations and discussions are described in Section 3. The last section concludes with highlights and future directions.

2. Investigation hyper-parameters

We would recall the hybrid model proposed in [10]. There are two phases in building such hybrid models. Firstly, normal data is used to train an AE with the objective of minimize its RE, typically *MSE*, with respect to AE parameters, W and $bias$, as illustrated in Figure 1(a) (training AE phase). Secondly, the decoder of the trained AE is discarded, and a density-based technique, such as Kernel Density Estimation (KDE) or Centroid (CEN), is stacked on the top of the AE encoder. The normal training data is then passed through the encoder again, and its latent data is used to construct a density-based learners as showed in Figure 1(b) (modeling density phase). A threshold of density can be set when training, possibly classifying 95%, 97.5% or 100% of the normal training data with density above the threshold. The choice of this threshold will depend largely on particular applications and scenarios. AE-KDE and AE-CEN denotes the hybrid models of AE and KDE, and AE and CEN, respectively. Once training is finished, the trained hybrid is employed to classify testing data, and a query data point whose density is below the threshold is identified as an anomaly as showed in Figure 1(c).

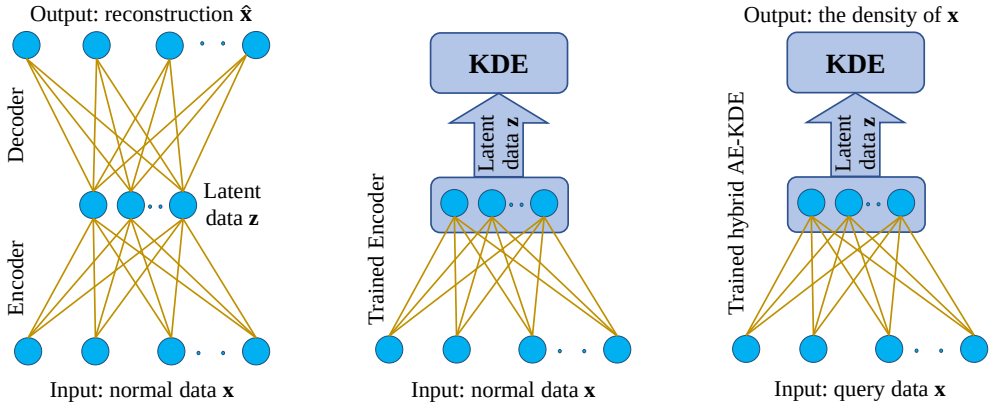


Fig. 1. The illustration of the hybrid AE and density-based methods (KDE) on training stage, training an AE (a) and training the hybrid (b), and on query stage (c).

The combination of AEs and density estimators obtains the advantages of their strengths. AEs can project the original data into a lower-dimensional feature space, and discover more robust/relevant features, while density estimators perform very efficiently in low-dimensional spaces. Any kinds of anomaly detection algorithms can be stacked on the top of the AE encoder in order to form a hybrid model. However, only KDE and

CEN are involved since this work aims to investigate the latent representation of AEs for further studies.

In the first phase of training AE-KDE, an AE learns from normal data to optimize its loss function (RE). Anomalies are assumed not involving in the cross-validation task for estimating hyper-parameters as in binary classification problems. How to estimate the hyper-parameters of AEs without using a validation set is still an open question for anomaly detection methods. This raises the first question of whatever RE and AUC_{AEKDE} have a strong association, and so RE could be used for tuning the AE hyper-parameters?. Secondly, we observe the influence of the hidden size of AEs on RE and the performance of AE-KDE. Following this, the distribution of latent vectors will be examined on different steepness values. Finally, the proposed hybrid is testified using four attack groups in NSL-KDD.

3. Experiments and Resulting Discussions

The section describes a set of investigations on the latent representation of AEs for getting insight into the characteristics of AEs. Thus, the first experiment is to investigate the influence the AE hyper-parameters on AUCs yielded by the hybrid AE-KDE. This aims to evaluate the rule of thumb for choosing a good middle hidden layer size (hz) proposed in [10]. The second investigation is to discover characteristics of the latent representation of AEs. Four UCI datasets are employed for these experiments, but they are not used for evaluating the proposed models later. Finally, the proposed models are assessed in comparison to stand-alone AE, CEN and KDE on the NSL-KDD dataset. These experiments will be presented in Subsections 3.2, 3.3 and 3.4.

The hyper-parameters of AEs and KDE will be set by common settings and using rules of thumb. The Gaussian kernel, a typical kernel machine methods, is used for KDE. Its bandwidth $h = \sqrt{\frac{n\_features}{2}}$ as a default setting in [11], where $n\_features$ is the number of original features (latent features if KDE is stacked on the top of an AE encoder). A non-linear sigmoid function is employed for hidden layers, whereas the output layer uses the identity function. AEs are trained by Back-propagation [12] together with the Adaptive Gradient Algorithm (Adagrad) [13]. The settings for the rest of the AE hyper-parameters will be discussed in the following subsections.

In practice, the choice of a classification threshold depends on specific applications. However, a number of classification thresholds are employed that attempts to evaluate AUC values. The implementation of our experiments uses Python 3.6.3, and the results are shown in Tables 2 to 5 and Figures 2 to 8.

3.1. Datasets

We select datasets that consist of main classes: one can used as normality and the other can be treated as anomaly data. The UCI datasets [14], such as WBC, WDBC, C-heart and ACA, and NS-KDD [15] are utilized for the experiments. Each of dataset in the UCI

datasets is randomly split into a training set (80%) and testing set (20%), respectively. Anomalies are discarded from the training set. Several of the categorical and discrete features in NSL-KDD are simply treated as real-valued features. This can provides good results shown in Section 3, but better prepossessing techniques are possible.

After investigating the hyper-parameters with the UCI datasets, we testify the hybrids of AEs and density-based methods on NSL-KDD. This aims to demonstrate how efficient performance our proposed model produces on particular attack groups. We aim to transfer the hyper-parameter knowledge on the UCI problems to the NSL-KDD problem. A similar sized of training data as those in the UCI problem is a good good choice. Thus, only 10% (6734) normal instances of *KDDTrain*$^+$ is randomly sampled for training model while all *KDDTest*$^+$ is used for evaluating the resulting model. The details of data are descibed in Table 1.

Table 1. Datasets for investigating and evaluating

Dataset	Features	Training set	Testing set	
		Normal	Normal	Anomaly
C-heart	13	128	32	28
ACA	14	306	77	62
WBC	9	355	89	48
WDBC	30	285	72	43
DoS	41	6734	9711	7458
R2L	41	6734	9711	2887
U2R	41	6734	9711	67
Probe	41	6734	9711	2421

3.2. Investigate the correlation amongst hidden size, RE and AUC

In this subsection, we will investigate (1) the influence of RE on the performance of AE-KDE, (2) the influence of the middle hidden layer size (hz) on RE, and on the performance (AUC) of AE-KDE w.r.t a set of hyper-parameters. These hyper-parameters such as number of epochs (ep), learning rate (α), the steepness (k) of the sigmoid function [1] and hz are set by wide ranges of values as follows: $ep \in \{3000, 5000, 10000\}$, $\alpha \in \{0.01, 0.05, 0.1, 0.2\}$, $hz \in \{2, 3, 4, 5, 6, 7, 8, 9, 10\}$, and $k \in \{0.01, 0.05, 0.1, 0.5, 1.0\}$. We take $\log(k)$ and $\log(\alpha)$ since k and α are really log-scale parameters.

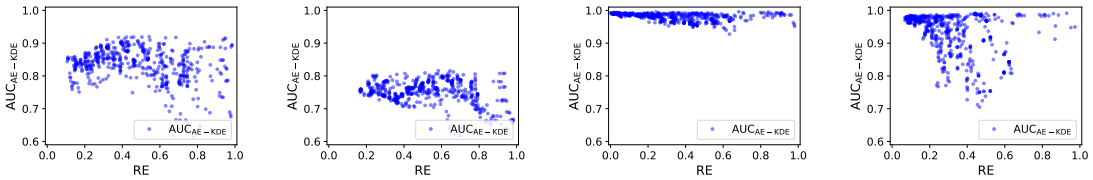
3.2.1. Influence of RE on the performance of AE-KDE: In the first phase of training AE-KDE, normal training data is used to trained an AE with the object of minimizing its loss function (RE). This raises the question of whatever RE and $\mathrm{AUC}_{\mathrm{AE-KDE}}$ have a strong association, and so RE could be used for tuning the AE hyper-parameters?. Therefore, we measure the association by Pearson's correlation coefficient (r) with respect to ep, $\log(\alpha)$, $\log(k)$ and hz as shown in Table 2, and demonstrated in Figure 2. The results show that the correlation is very weak (around -0.30) on the four UCI datasets. Figure 2 also illustrates the very weak correlation between AUC and RE because AUC values tend to not depend on the RE values. This suggests that RE can

[1] $f(z) = \frac{1}{1+e^{(-kz)}}$, where z is the input of the sigmoid function, $k \in \mathbb{R}^+$

not be used for tuning hyper-parameters. Thus, the influences of hz on RE and AUC will be further examined in the following.

Table 2. The Pearson correlation between RE and $\mathrm{AUC}_{\mathrm{AE-KDE}}$.
All measures are highly statistically significant (p-value < 0.01*).*

		RE			
		ACA	**C-heart**	**WBC**	**WDBC**
AUC	r	-0.281	-0.315	-0.332	-0.300
	p-value	3.1×10^{-11}	6.9×10^{-14}	2.2×10^{-15}	1.1×10^{-12}



(a) (b) (c) (d)

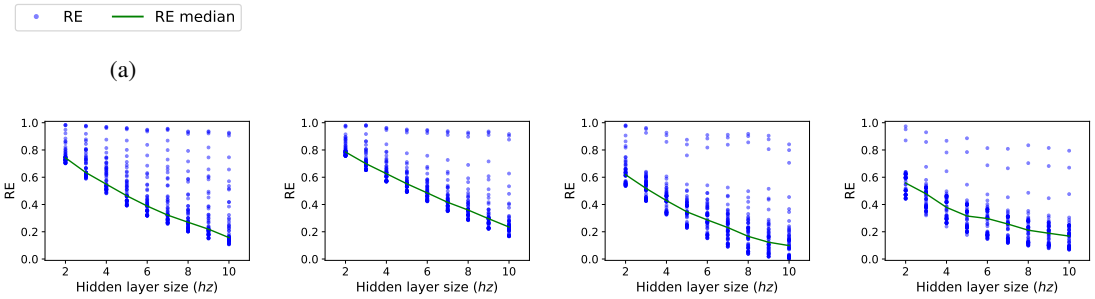
Fig. 2. Investigations of the relationship between $\mathrm{AUC}_{\mathrm{AE-KDE}}$ *and RE with respect to hyper-parameters* ep, $\log(\alpha)$, hz *and* $\log(k)$ *on the UCI datasets*

3.2.2. Influence of hz *on RE and the performance of AE-KDE:* The associations between hz and RE, and between hz and $\mathrm{AUC}_{\mathrm{AE-KDE}}$ are evaluated by Pearson's correlation coefficient as presented in Table 3. The first row in the table presents a very strong negative correlation (r around -0.70) between hz and RE on the four datasets. This would indicate that the larger hz an AE have, the easier the AE learn normal behavior and reproduce the original data in the last layer, and the smaller RE the AE creates. This is also illustrated in Figure 3 that RE decreases as hz increases. However, hz and $\mathrm{AUC}_{\mathrm{AE-KDE}}$ seem to have no association since Pearson's correlation coefficient r is very small (except on WBC) as illustrated in the 2^{nd} row of Table 3. Note that RE and $\mathrm{AUC}_{\mathrm{AE-KDE}}$ are computed with respect to wide ranges of hype-parameter settings. Thus, we cannot use cross-validation on RE, with normal data only, to tune hyper-parameters such as α, k, ep and hz.

Table 3. The Pearson correlation between the middle hidden size and RE, $\mathrm{AUC}_{\mathrm{AE-KDE}}$.
All measures are highly statistically significant (p-value < 0.01*).*

		Hidden layer size (hz)			
		ACA	**C-heart**	**WBC**	**WDBC**
RE	r	-0.802	-0.707	-0.707	-0.682
	p-value	1.2×10^{-122}	6.4×10^{-83}	4.8×10^{-83}	3.4×10^{-75}
AUC	r	0.177	0.136	0.570	0.164
	p-value	3.6×10^{-5}	1.6×10^{-3}	6.3×10^{-48}	1.2×10^{-4}

3.2.3. Rule of Thumb to select the hidden size of AEs: In order to introduce a rule of thumb to estimate hz, we further examine the relationship between hz and $\mathrm{AUC}_{\mathrm{AE-KDE}}$



(a) (b) (c) (d) (e)

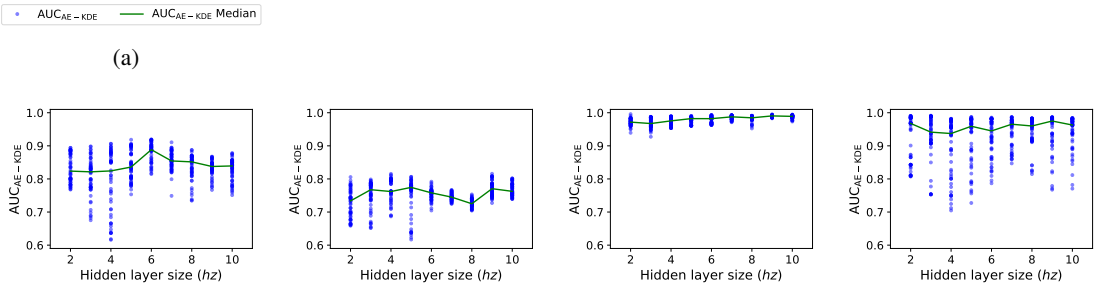
Fig. 3. Investigation the relationship between hz and RE with respect to different values of hyper-parameters ep, $\log(\alpha)$ and $\log(k)$) on the UCI datasets: C-heart (b), ACA (c), WBC (d) and WDBC (e).

with respect to a wide range values of the hyper-parameters ep, α, and k. Note that the relationship between hz and $\mathbb{AU}_{\mathbb{AE-KDE}}$ investigated in our previous work [10], but the values of the hyper-parameters ep, α, and k were fixed. Figure 4 illustrates $\mathrm{AUC}_{\mathrm{AE-KDE}}$ and its median value against ten different values of hz. When observing the curve of the median $\mathrm{AUC}_{\mathrm{AE-KDE}}$, there is no common pattern amongst the four figures. For instance, at a large value $hz = 8$ the $\mathrm{AUC}_{\mathrm{AE-KDE}}$ median reaches the lowest value on ACA, but on WBC and C-heart it is relatively high. The number of original features of C-heart, ACA, WBC are not much different (13, 14 and 9 respectively). The hidden layer size is related to how much information will be compressed in the hidden layer. The best value of hz may vary according to the number of original dimentions and the distribution of given datasets. Based on the experiments, we observe that the rule of thumb proposed in [10], $hz = [1 + \sqrt{n}]$ with n refers to the number of original dimensions, is not the best one, but it is still acceptable. Thus, we use the rule of thumb proposed in [10] for the followed experiments in this paper.

Based on the rule, the hyper-parameter hz can be calculated for each dataset, $hz = 4$ for ACA, WBC and C-heart, and $hz = 6$ for WBCD, and 7 for NSL-KDD. When observing the performance of AE-KDE on the hz settings, the hybrid produces very high or reasonable $\mathrm{AUC}_{\mathrm{AE-KDE}}$ median values.

3.3. Investigate Latent Vectors

3.3.1. Influence of steepness on RE and AUC$_{AE-KDE}$: This subsection is to study the characteristics of the latent representation of AEs. Firstly, the associations of $\log(k)$ with $\mathrm{AUC}_{\mathrm{AE-KDE}}$, and with RE are examined with respect to other hyper-parameters (shown in Subsection 3.2). These are measured by Pearson's correlation coefficients as presented in Table 4, and also illustrated in Figures 5 and 6. The Pearson correlation coefficients in the first row are very small, which indicates that $\log(k)$ and RE have a weak association. However, $\mathrm{AUC}_{\mathrm{AEKDE}}$ and $\log(k)$ has a strong negative correlation on the WBC and WDBC datasets (about -0.4 and -0.8 respectively). This means that



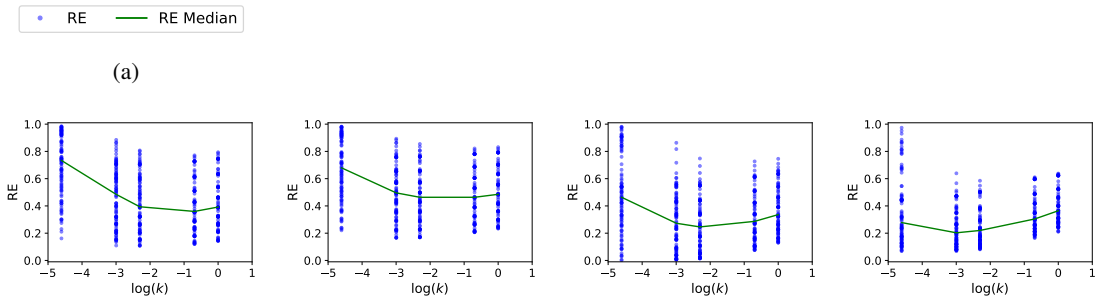
(a) (b) (c) (d) (e)

Fig. 4. Investigation the relationship between $\mathrm{AUC}_{\mathrm{AE-KDE}}$ *and* hz *with respect to different values of hyper-parameters* $epoch$, α *and* k *on the UCI datasets: C-heart (b), ACA (c), WBC (d) and WDBC (e)*

the smaller value k is, the better performance AE-KDE tends to produce on WBC and WDBC. This motivates me to see what the distribution of latent vectors of WBC and WDBC looks like when k varies.

Table 4. The Pearson correlation between $\log(k)$ *and RE,* $\mathrm{AUC}_{\mathrm{AE-KDE}}$*. All measures are highly statistically significant (p-value* < 0.01*) except the correlation between* $\log(k)$ *and* $\mathrm{AUC}_{\mathrm{AE-KDE}}$ *on C-heart.*

		log(k)			
		ACA	**C-heart**	**WBC**	**WDBC**
RE	r	-0.249	-0.379	-0.150	0.119
	p-value	4.5×10^{-9}	7.1×10^{-20}	4.7×10^{-4}	5.6×10^{-3}
AUC	r	0.322	0.009	-0.394	-0.773
	p-value	1.7×10^{-14}	8.4×10^{-1}	1.6×10^{-21}	2.7×10^{-108}



(a) (b) (c) (d) (e)

Fig. 5. Investigation the relationship between $\log(k)$ *and RE with respect to hyper-parameters* ep, $\log(\alpha)$ *and* hz *on the four UCI datasets: C-heart (b), ACA (c), WBC (d) and WDBC (e)*

3.3.2. Visualize latent vectors: We select four AE-KDE models with regard to four different values of k such as $0.05, 0.1, 0.5$ and 1.0 for created the latent vectors of WBC and WDBC. The purpose is to show the behaviour of latent vectors when the value of

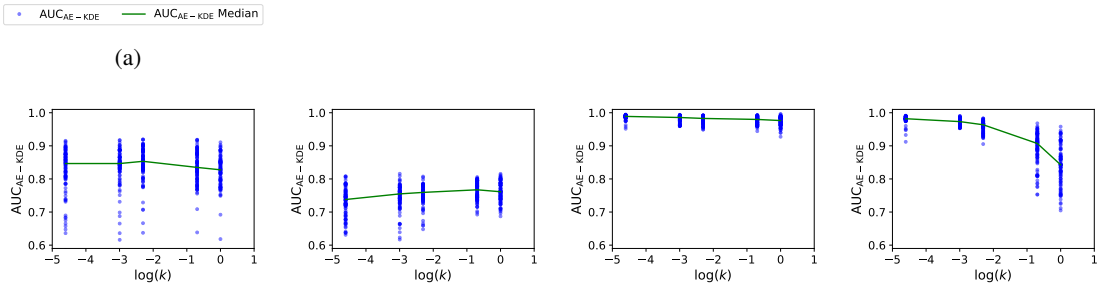
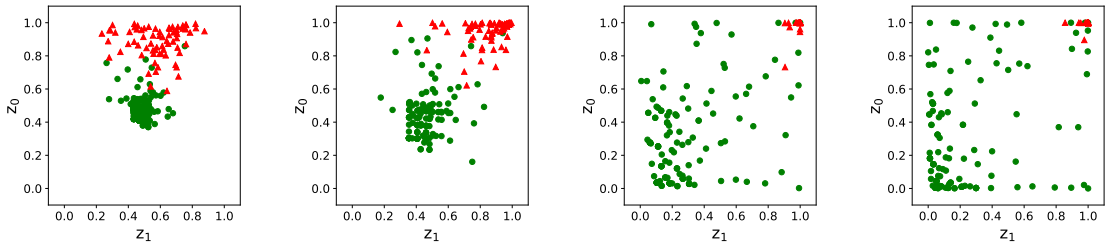


Fig. 6. Investigation the relationship between $\mathrm{AUC}_{\mathrm{AE-KDE}}$ *and* $\log(k)$ *with respect to hyper-parameters* ep, $\log(\alpha)$ *and* hz *on the UCI datasets: C-heart (b), ACA (c), WBC (d) and WDBC (e)*

k increases. Thus, we would choose a large step of k, four values of k in the range of [0, 1]. Other hyper-parameters of these models are set by a single value, $ep = 5000$, $\alpha = 0.01$ and $hz = 4$ (WBC) or $hz = 6$ (WDBC). The first two features, z_0 and z_1, of the latent vectors of testing sets are visualized as shown in Figures 7 and 8.

These figures shows that for small k (0.05 and 0.1), normal data is approximately Gaussian, and it appears to be close to the origin of the hidden unit outputs (typically 0.5 for sigmoid units and 0.0 for $\tanh$ units) [10]. This is each coordinate of the latent vectors (given by the output of the hidden unit activation), such as z_0 and z_1, is very close to the most non-saturating value of the activation outputs. However, both normal data and anomalies seem to be distributed along the borders of a hyper-box for large values of k (0.5 and 1.0). Each coordinate tends to bloat to the most saturating values of the hidden unit outputs (0.0 and 1.0). For highly non-saturating regions of the hidden unit output, a small variance on its input will result in a large change on the output, whereas the output tends to be insensitive to changes in the input on highly saturating regions. Therefore, these evidences could be useful for explaining why the AE-KDE model prefers small values k on WBC and WDBC as shown in Table 4. This means that for small k (associated with large non-saturating regions of the activation outputs), each hidden unit tends to behave very similar on normal data, while anomalies that deviate significantly from normality tend to be much more different on the outputs. If normal data and anomalies are separated on each hidden unit, they will be highly distinguished on the combination of all the hidden unit outputs.

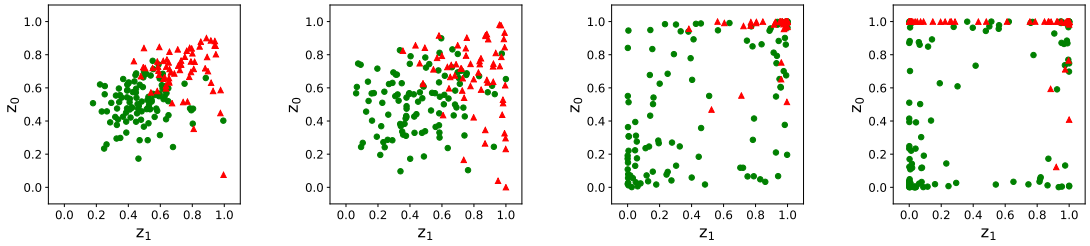
Using the hyper-parameter such as k is not an effective approach since it will increase the number of hyper-parameters, and how to choose k for a specific dataset is problematic. Alternatively, the latent vectors of normal data could be constrained towards highly non-saturating regions of the hidden unit outputs. A possible solution for this is to propose new regularization terms, to be added to the loss function of AEs. The use of regularization terms can avoid additional hyper-parameters, and be flexible to apply for many kinds of activation functions. In the followed subsection, k will be chosen equal to 1.0 for evaluating the proposed models on the NSL-KDD dataset.



(a) $k = 0.05$ (b) $k = 0.1$ (c) $k = 0.5$ (d) $k = 1.0$

(e)

Fig. 7. Illustration of hidden data (the 1^{st} two dimensions,, z_0 and z_1) of an AE with respect to four different values of k, and $hz = 4$ on WBC.



(a) $k = 0.05$ (b) $k = 0.1$ (c) $k = 0.5$ (d) $k = 1.0$

(e)

Fig. 8. Illustration of hidden data (the 1^{st} two dimensions, z_0 and z_1) of an AE with respect to four different values of k, and $hz = 6$ on WDBC.

3.4. Evaluate Proposed Models

3.4.1. Experimental Setup: This experiment is to testify the proposed model performance, such as AE-CEN and AE-KDE in comparison to that of stand-alone AE, CEN and KDE on NSL-KDD. The hyper-parameters of KDE are set up as presented above. The Adagrad [13] with common values of learning rate and smoothing term ($\alpha = 0.01$ and $\varepsilon = 10^{-8}$) are configured for training AEs. The number of epochs is equal to 5000. The mini-batch size can vary from 10 to 100 depending on datasets as presented in [16], in this work, the size of a mini-batch is set to 20. The middle hidden layer size is set by using the rule of thumb proposed in Subsection 3.2, thus $hz = 7$.

3.4.2. Results and Discussion: Table 5 illustrates our experimental results. The table presents the AUC values created by stand alone models (AE, CEN and KDE) in the first row, and by hybrids (AE-CEN and AE-KDE) in the second row. The values in both,

*Table 5. The AUC created by hybrid and stand alone models on NSL-KDD. The cases where the AUC of the hybrids is improved in comparison to CEN are indicated by bold text, and where improved in comparison to KDE are indicated by *.*

Anomaly detection models	**NSL-KDD dataset**			
	DoS	**R2L**	**U2R**	**Probe**
AE	0.948	0.910	0.942	0.969
CEN	0.947	0.883	0.928	0.973
KDE	0.950	0.885	0.932	0.975
AE-CEN	0.946	0.864	0.903	**0.979***
AE-KDE	**0.957***	**0.887**	**0.940**	**0.987***

and in * indicate the cases where the AUC of the hybrids is improved in comparison to those of stand-alone CEN and KDE, and AE respectively.

It can be seen from Table 5 that the performance of hybrid AE-KDE is very good in terms of the AUC values, and higher than those of stand-alone CEN and KDE on the four groups of attacks. In comparison to AE, the hybrid shows improvements in AUC on only DoS and Probe while AE out-performs on R2L and U2R. Moreover, AE-KDE out-performs AE-CEN on all attack groups in the NSL-KDD dataset, although the performance of KDE and CEN is competitive on the original input. These can be explained that the density of normal data in the hidden feature space (normal latent vectors) may be higher than those in the input feature space since normal hidden vectors are put close together. The distribution of normal latent vectors can also have an arbitrary shape. KDE can model the density of normal data without any assumption about the underlying the normal data distribution [17], thus the hidden representation can benefit the density-based method such as KDE. However, CEN assume that the training data distributes in a spherical Gaussian shape. Thus, it is not facilitated from the latent representation of AEs, and the performance of AE-CEN does not improve (except on the Probe group) because the normal latent vectors are not Gaussian. These conclusions can be also supported by the experimental results and analysis from Subsection 3.3when $k = 1$, the normal latent vectors have high density in the regions near the borders, and with an arbitrary shape.

Overall, the experimental results suggest that AE-KDE is more efficient than CEN, KDE and AE-CEN in identifying anomalies from the four attack groups. This is because the latent normal vectors that are represented in a lower dimension and higher-density regions are easy for KDE to model. However, the latent representation tends to not facilitate CEN since the normal latent vectors are not in a good shape such as Gaussian distribution. Besides this, AE-KDE performs competitively AE in terms of classification accuracy.

4. Conclusion

This work is an extension of our previous study [10] on using the latent representation of AEs for improving network anomaly detection. The motivation is to achieve the good characteristics of the latent representation: representing original data in a lower feature space; revealing relevant/robust features; and compressing normal data into some regions with high density. These potentially enhance the ability of density-based anomaly detection in dealing with high-dimensional network data. Thus, we have done a set of investigations on the latent representation, and achieved promising results.

Firstly, the association between $\mathrm{AUC}_{\mathrm{AE-KDE}}$ and RE, and the influence of the middle hidden layer size hz on the hybrid AE-KDE are measured. The results testify that they have very weak correlations, meaning that RE can not be used for doing cross-validation. Thus, we have proposed a rule of thumb for choosing a proper hz. Secondly, we further examine the steepness of the sigmoid function k, and observe the distribution of latent vectors with regards to different values of k. The results suggest that $\mathrm{AUC}_{\mathrm{AE-KDE}}$ tends to prefer small k (on WBC and WDBC). In this case, normal latent vectors reside close to the origin of the hidden unit outputs that is the most non-saturating region on the activation curves. Finally, the proposed models are evaluated on NSL-KDD. The experimental results suggest that the hybrid AE-KDE often out-performs stand-alone KDE and CEN, and performs competitively to AE. The hybrid AE-KDE also performs better than AE-CEN. This is because the normal latent vectors have an arbitrary shape that does not facilitate simple method such as CEN.

In the future work, the Bayesian Optimization will be employed to find the good hyper-parameters for the tested models. We will also aim to tailor good latent representations that potentially benefit to wide range of machine learning methods for anomaly detection including simple one such as CEN.

References

[1] A. Zimek, E. Schubert, and H.-P. Kriegel, "A survey on unsupervised outlier detection in high-dimensional numerical data," *Statistical Analysis and Data Mining*, vol. 5, no. 5, pp. 363–387, 2012.

[2] S. M. Erfani, S. Rajasegarar, S. Karunasekera, and C. Leckie, "High-dimensional and large-scale anomaly detection using a linear one-class SVM with deep learning," *Pattern Recognition*, vol. 58, pp. 121–134, 2016.

[3] H. Song, Z. Jiang, A. Men, and B. Yang, "A hybrid semi-supervised anomaly detection model for high-dimensional data," *Computational Intelligence and Neuroscience*, vol. 2017, 2017.

[4] G. Pang, C. Shen, L. Cao, and A. V. D. Hengel, "Deep learning for anomaly detection: a review," *ACM Computing Surveys (CSUR)*, vol. 54, no. 2, pp. 1–38, 2021.

[5] S. Bulusu, B. Kailkhura, B. Li, P. K. Varshney, and D. Song, "Anomalous example detection in deep learning: A survey," *IEEE Access*, vol. 8, pp. 132330–132347, 2020.

[6] N. Japkowicz, C. Myers, and M. Gluck, "A novelty detection approach to classification," in *IJCAI*, pp. 518–523, 1995.

[7] U. Fiore, F. Palmieri, A. Castiglione, and A. De Santis, "Network anomaly detection with the Restricted Boltzmann Machine," *Neurocomputing*, vol. 122, pp. 13–23, 2013.

[8] S. Hawkins, H. He, G. Williams, and R. Baxter, "Outlier detection using replicator neural networks," in *Data warehousing and knowledge discovery*, pp. 170–180, Springer, 2002.

[9] M. Sakurada and T. Yairi, "Anomaly detection using autoencoders with nonlinear dimensionality reduction," in *Proc MLSDA*, p. 4, ACM, 2014.

[10] V. L. Cao, M. Nicolau, and J. McDermott, "A hybrid autoencoder and density estimation model for anomaly detection," in *Parallel Problem Solving from Nature*, pp. 717–726, Springer, 2016.
[11] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.
[12] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning internal representations by error propagation," tech. rep., DTIC Document, 1985.
[13] J. Duchi, E. Hazan, and Y. Singer, "Adaptive subgradient methods for online learning and stochastic optimization," *Journal of Machine Learning Research*, vol. 12, no. Jul, pp. 2121–2159, 2011.
[14] M. Lichman, "UCI machine learning repository," 2013.
[15] M. Tavallaee, E. Bagheri, W. Lu, and A. A. Ghorbani, "NSL-KDD dataset," *http://www.iscx.ca/NSL-KDD*, 2012.
[16] G. E. Hinton, "A practical guide to training Restricted Boltzmann Machines," in *Neural networks: Tricks of the trade*, pp. 599–619, Springer, 2012.
[17] M. P. Wand and M. C. Jones, *Kernel smoothing*. CRC Press, 1994.

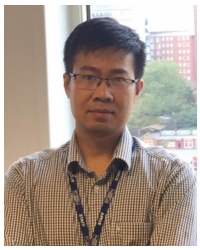
Manuscript received 02-05-2021; Accepted 25-06-2021.

■

Van Loi Cao received the B.Sc. and M.Sc. degree in computer science from Le Quy Don Technical University (LQDTU), Hanoi, Vietnam, and the Ph.D degree from University College Dublin (UCD), Dublin, Ireland. He is currently the Deputy Head of Information Security Department, the Faculty of Information Technology, LQDTU. His current research interests include Deep Learning, Machine Learning, Anomaly Detection, IoT Security, and Information Security. Email: loi.cao@lqdtu.edu.vn.

Huu Noi Nguyen received the B.Sc. degree in applied mathematics and informatics from Lipetsk State University, Lipetsk, Russia. He is currently studying the Ph.D. program in Computer Science at Le Quy Don Technical University. His current research interests include Machine Learning, Anomaly Detection, IoT and Information Security.
Email: noi.nguyen@lqdtu.edu.vn.

Van Quan Nguyen is currently working in Information Security - IT Faculty at Le Quy Don Technical University. He received an engineering degree and a master's degree from Bauman National University - Russia in 2012. Current main research directions are machine learning, deep learning, cybersecurity, digital forensics... Email: quannv@lqdtu.edu.vn.



Viet Hung Nguyen is currently working in the Department of Information Security – IT Faculty at Le Quy Don Technical University. He received his BSc, MSc and PhD degrees in Computer Science from Moscow Institute of Physics and Technology in 2006, 2008 and 2012 respectively. His research interests include neural networks, malware detection, Intrusion detection and cyber security. Email: hungn@lqdtu.edu.vn.

Van Thang Cao is currently working in Telecommunications University, Nhatrang, Vietnam. He received the B.Sc degree in Electronics and Telecommunications from Telecommunications University in 2005, and the M.Sc degree from Post and Telecommunications Institute of Technology, HCM, Vietnam in 2013. His current research interests include Machine Learning, Deep Learning, and Data communication. Email: thangcaosqtt@gmail.com.

KHẢO SÁT CÁC SIÊU THAM SỐ CỦA MẠNG AUTOENCODER CHO PHÁT HIỆN BẤT THƯỜNG

Cao Văn Lợi, Nguyễn Hữu Nội, Nguyễn Văn Quân, Nguyễn Việt Hùng, Cao Văn Thắng

Tóm tắt

Hầu hết các kỹ thuật phát hiện bất thường, chẳng hạn các phương pháp dựa trên mật độ, thường hoạt động không hiệu quả trên dữ liệu mạng có số chiều lớn do hiện tượng "the curse of dimensionality phenomenon". Một nghiên cứu trước đây của chúng tôi đã trình bày một cách tiếp cận mới trong việc sử dụng không gian biến ẩn của AutoEncoders (AEs) như một không gian biểu diễn mới để giải quyết vấn đề này. Trong nghiên cứu này, chúng tôi sẽ khảo sát các đặc điểm của không gian biểu diễn biến ẩn của các AE. Vì vậy, trước tiên chúng tôi sẽ trình bày chi tiết về giả thuyết sử dụng biểu diễn ẩn của AEs, và mở rộng một số thí nghiệm được trình bày trong nghiên cứu trước đây. Sau đó, chúng tôi sẽ thực hiện ba khảo sát chuyên sâu (khảo sát về kích thước lớp ẩn giữa, đánh giá hiệu suất của mô hình lai ghép và khám phá dữ liệu ẩn). Những thí nghiệm này nhằm mục đích cung cấp một cái nhìn sâu sắc về không gian biểu diễn ẩn của AEs, đây là cơ sở để thiết kế các không gian biểu diễn ẩn tốt hơn trong tương lai. Dựa trên kết quả thí nghiệm, bài báo đã đưa ra những phân tích, thảo luận và gợi mở hướng tiếp cận phát triển không gian biểu diễn biến ẩn của AEs trong tương lai.