Dimensionality Reduction and Visualization of Network Intrusion Detection Data

Chapter *in* Lecture Notes in Computer Science · May 2019

DOI: 10.1007/978-3-030-21548-4_24

3 authors:

Wei Zong
University of Wollongong
16 PUBLICATIONS **167** CITATIONS
SEE PROFILE

Yang-Wai Chow
University of Wollongong
97 PUBLICATIONS **984** CITATIONS
SEE PROFILE

Willy Susilo
University of Wollongong
926 PUBLICATIONS **23,735** CITATIONS
SEE PROFILE

Dimensionality Reduction and Visualization of Network Intrusion Detection Data

Wei Zong, Yang-Wai Chow, Willy Susilo

Institute of Cybersecurity and Cryptology
School of Computing and Information Technology
University of Wollongong, NSW, Australia
wz630@uowmail.edu.au, {caseyc, wsusilo}@uow.edu.au

Abstract. Nowadays, network intrusion detection is researched extensively due to increasing global network threats. Many researchers propose to incorporate machine learning techniques in network intrusion detection systems since these techniques allow for automated intrusion detection with high accuracy. Furthermore, dimensionality reduction techniques can improve the performance of machine learning models, and as such, are widely used as a pre-processing step. Nevertheless, many researchers consider machine learning techniques as a black box because of its complex intrinsic mechanism. Visualization plays an important role in facilitating the understanding of such sophisticated techniques because visualization is able to offer intuitive meaning to the machine learning results. This research investigates the performance of two dimensionality reduction techniques on network intrusion detection datasets. In addition, this work also demonstrates visualizing the resulting data in 3-dimensional space. The purpose of this is to possibly gain insight into the results, which can potentially aid in the improvement of machine learning performance.

Keywords: dimensionality reduction; machine learning; network intrusion detection; visualization.

1 Introduction

The Internet is essential in daily life for almost everyone in contemporary society. Meanwhile, there has been extensive research conducted on Network Intrusion Detection Systems (NIDS) due to the increasing global threat of cyberattacks. Machine learning techniques have been proposed by cyber security experts as a promising solution for NIDS to combat cyberattacks. This is because machine learning can provide an automated approach to detecting intrusions with high accuracy [13].

To improve the intrusion detection performance of machine learning models, techniques for dimensionality reduction are widely used as one of the preprocessing steps [3] [11]. Wang et al. [16] investigated different dimensionality reduction techniques and concluded that the autoencoder technique outperforms

other dimensionality reduction techniques in certain situations. This technique has also been adopted for the purpose of network intrusion detection. As an example, the autoencoder technique was used in Javaid et al. [4] to learn new feature representation before using a soft-max regression for classification.

Although some machine learning models can provide adequate intrusion detection performances, the underlying reasons that affect accuracy are not usually analyzed. Furthermore, improvement of machine learning models usually rely on a time-consuming trial-and-error process due of the complex nature of machine learning mechanisms [7]. The reason for this is because machine learning is typically treated as a black box, and while the performance might be impressive, researchers may not know the theoretical link between a machine learning model and its performance [16].

Information visualization techniques can potentially bridge the gap between the performance of machine learning models and understanding factors that contribute to its performance. Visualization, whether in 2-dimensions (2D) or 3-dimensions (3D), also plays an important role in the cyber security domain [14]. In addition, previous work has shown that complex attack patterns in NIDS can be visualized in various forms [1] [9]. Previous research in this area includes a visual approach to analyzing the characteristics of network intrusion detection datasets in 3D space [17]. Visualization makes these characteristics more comprehendible and intuitive, while they may be difficult to perceive when using traditional statistical data analysis alone [7].

In this paper, we first investigate the performance of two dimensionality reduction techniques on the benchmark NSL_KDD and UNSW-NB15 network intrusion datasets. The results show the relationship between the number of dimensions and the intrusion detection performance. This allows us to identify the number of dimensions that will give rise to good performance for different classifiers. We then implement a method to visualize the data in 3D, in order to observe patterns in the data and to gain a better understanding of the machine learning results. In this visual form, the data is more intuitive and potential insight can be gained to improve machine learning performance.

Our Contributions. This paper investigates and compares the performance of two dimensionality reduction techniques, namely, the principal component analysis and autoencoder techniques, for network intrusion detection using three different classifiers. The classifiers that were used in this study were the k-nearest neighbors classifier, the multi-layer perceptron classifier and the decision tree classifier. Results of our experiments show the relationship between the number of dimensions and the intrusion detection performance for the respective classifiers. This paper also demonstrates how visually presenting the results in 3D space can facilitate the intuitive identifying of patterns in the data. This can potentially provide useful insight that can be used to understand and improve machine learning performance, rather than relying on the usual trial-and-error process.

2 Background

2.1 Dimensionality Reduction

Dimensionality reduction is used in a number of areas, including for machine learning. The research presented in this paper investigates the use of two of these techniques for the purpose of network intrusion detection.

Principal Component Analysis (PCA) is a commonly used dimensionality reduction technique for projecting data onto new axes which are orthogonal to each other [5]. In PCA, the first principal component captures the largest variance, while the second principal component capture the largest variance among the remaining orthogonal directions, etc. Therefore, each principal component captures the largest variance excluding the preceding principal components. To project data into 3D space, an approach is to only use data from the first 3 principal components.

Autoencoder is a type of artificial neural network that can be used for dimensionality reduction. This is because it can automatically learn feature representation of the data. The autoencoder technique consists of an encoder and a decoder. When the number of nodes in the hidden layer are made smaller than the input nodes, autoencoder can learn a compressed presentation of the data. In this manner, autoencoder is capable of reducing the dimensions of the input data. Compared with other dimensionality reduction techniques, autoencoder may produce more favorable results in certain situations and can detect repetitive characteristics in datasets [16].

2.2 Network Intrusion Detection Datasets

Network intrusion detection datasets are important when it comes to validating the performance of NIDS. Benchmark datasets for NIDS, namely, KDD98, KDD CUP99 and NSL_KDD, are widely used in research to compare results of intrusion detection methods. These datasets categorize network attacks into different types, e.g., Denial of Service (DoS) attacks, probe attacks, Remote to Local (R2L) and User to Root (U2R) attacks. However, it has been contended that these datasets are outdated since they were proposed more than a decade ago. In addition, they contain some flaws which negatively affect the performance of NIDS [8]. To reflect contemporary cyber traffic, the UNSW-NB15 dataset was proposed [10]. In addition to normal connections, this dataset contains 9 types of network attacks, including worm and shellcode attacks. Although in this paper, we utilize NSL_KDD and UNSW-NB15 datasets for our experiments, our proposed approach can be applied to other network intrusion detection datasets.

2.3 Related Work

Dimensionality reduction techniques can improve the performance of machine learning models. Among various techniques, PCA and autoencoder are widely used to reduce the high number of dataset dimensions before classification.

Moustafa et al. [11] used the PCA technique to reduce the high dimension of the network intrusion datasets before classifying cyberattacks. Hoz Correa et al. [3] also used the PCA technique to select useful features and to remove noise in network intrusion data. Javaid et al. [4] used the autoencoder technique to learn a feature representation of the NSL_KDD dataset. Then, they used softmax regression to do the classification and achieved competitive results. Wang et al. [16] compared autoencoder with other commonly used dimensionality reduction techniques, such as PCA and Isomap, on synthesized data and image datasets. Their study showed that results obtained from the use of autoencoder differed from other dimensionality reduction techniques, and concluded that the autoencoder technique is potentially suitable for detecting repetitive structures in datasets.

In the NIDS domain, machine learning approaches have been extensively studied as these are seen as promising solutions towards automating the detection of abnormal network connections with high accuracy [13]. For example, Lin et al. [6] considered the geometric relationship between data records and proposed a novel feature representation method. They then used a k-Nearest Neighbors (kNN) classifier to detect cyberattacks. Wang et al. [15] proposed a multi-step NIDS. They first divided the training set into subsets by fuzzy clustering. Subsequently, they trained an artificial neural network on each subset. Finally, the detection results were combined using a fuzzy aggregation module. Their method was reported to achieve high network intrusion detection performance. In addition, a two-stage approach for network intrusion detention has also been proposed, where different machine learning models can be used in the different stages [18]. An advantage of this approach is that it can deal with the extremely imbalanced characteristics of network intrusion datasets.

Although machine learning models can achieved satisfactory results, the underlying reasons affecting accuracy are still not well understood. As an example, Javaid et al. [4] demonstrated the competitive performance of their approach without analyzing the reasons for misclassification. Moustafa et al. [11] proposed a novel approach, called geometric area analysis based on trapezoidal area estimation for NIDS. Their approach effectively detected intrusions in the NSL_KDD and UNSW-NB15 datasets. However, they did not analyze misclassification in detail. It has been argued that without a comprehensive and intuitive understanding of the underlying reasons that cause misclassification, the improvement of machine learning models usually relies on a time-consuming trial-and-error process due to the complex nature of machine learning mechanisms [7].

Visualization techniques can be used to facilitate the development of machine learning models since these techniques can show characteristics that humans can understand intuitively. Rauber et al. [12] proposed to visualize relationships between learned representations of observations, and relationships between artificial neurons. They performed this projection using t-distributed Stochastic Neighbor Embedding (t-SNE), so that they could view the data in 2D space. In other work, Liu et al. [7] proposed a system for enabling users to perform

visual analysis to help understand, diagnose and refine deep convolution neural networks.

Visualization approaches have also been proposed in the field of network intrusion detection. Angelini et al. [1] described a cyber security visualization system that can facilitate user awareness of cyber security statuses and events. McKenna et al. [9] showed a cyber security dashboard that can help experts understand global attack patterns. An approach to visualizing network intrusion datasets in 3D space was presented in Zong et al. [17]. Results of this approach demonstrated that it can be used to identify visual characteristics in the datasets, which can potentially contribute to improving detection performance of machine learning models in NIDS.

3 Proposed Approach

In this section, we describe the details of our proposed approach. In essence, the purpose of this work is to examine dimensionality reduction and visualization for network intrusion detection. For this, we investigated the relationship between the number of dimensions and intrusion detection performance. This allowed for the identification of a good value for dimension reduction that will produce reasonably good performance for different classifiers. We then implemented a method to visualize the data in 3D, in order to examine the intrusion detection results from the visual representation. The various stages involved in the overall process is depicted in Fig. 1.

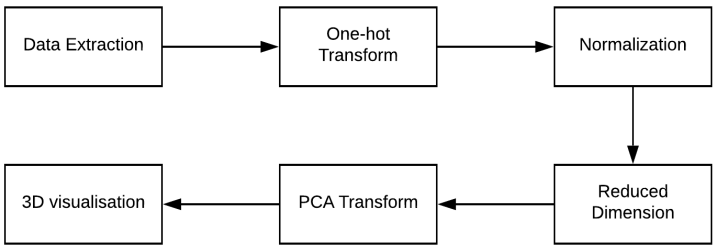


Fig. 1. Stages in the proposed approach.

The NSL_KDD and UNSW-NB15 network intrusion detection datasets were used in this study. The first step was to extract data from the original datasets. NSL_KDD and UNSW-NB15 are known as imbalanced datasets, because they contain minor classes that only occupy a relatively small proportion of the dataset, whereas the remainder of the dataset consists of major classes [18]. For example, worm attacks in the UNSW-NB15 occupy $< 1\%$ of the dataset,

similarly U2R attacks in the NSL_KDD only represents a minor portion of the dataset.

Methods to improve the intrusion detection performance of imbalanced datasets is to over-sample minor classes, to down-sample major classes, or both [2]. Therefore, in the data extraction stage, we extracted all minor classes from the dataset. Then, we randomly extracted other classes until a certain percentage, 30% in our experiments, of the dataset was extracted to establish our training set. Other than our training set, we also extracted data from the original training set which accounted for 10% of the data to establish a validation set. Since the training set includes all the minor classes, the minor classes in the validation set are repeated in the training set. However, other classes in the validation set are not repeated in the training set. In this way, we could use less computational power to achieve satisfactory detection results and the visual quality in 3D space was not adversely affected.

Subsequently, one-hot transform was applied to the categorical features in the datasets since the dimensionality reduction techniques adopted in our experiments, i.e. PCA and autoencoder, only operate on numeric data and are not suitable for categorical features. After one-hot transformation, only numeric data remains. It should be noted that, one-hot transform is applied to the training and test sets separately. Consequently, the training set may generate some features that do not exist in the test set and vice versa. This may happen because some categorical values may exist in only one set but not in both sets. To handle this situation, we only used features in the transformed training set. In this way, whenever the training set contains features that were missing in the test set, a value of zero would be used. On the other hand, if the test set contains some features that the training set did not contain, such features were ignored.

The next step was to normalize the data. Normalization was performed because the numeric range of the different features can vary significantly. For example, some features range between 0 to 100, while other features range from 0 to several million. Without normalization, this would negatively affect the dimensionality reduction results. The test set was normalized based on the training data. Specifically, only the maximum and minimum values of each feature in the training set were used to normalize both the training and test sets.

To examine the number of dimensions that would produce the best detection performance, we reduced the dimensions to a range of values. In our experiments, the number of dimensions ranged from 2 to 30. Then, we applied basic classifiers, such as k-nearest neighbors and decision trees to the data. The number of dimensions that gave rise to reasonably good performance for all classifiers was identified to be as the best value to use for dimensionality reduction.

Once this value was selected, dimensionality reduction was performed on the original data to transform the data into the specific number of dimensions. The PCA technique was then used to transform the data into 3D space in order to visualize the results. The reason why the PCA algorithm was used is because PCA transformation can be inversed. In this manner, when performing visual examination on certain areas of the data in 3D space, the data can be inversed

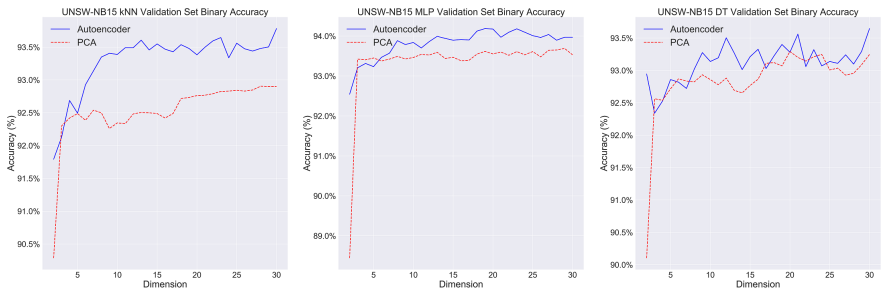
and examined in higher dimension space. Thus, allowing us to adequately analyze the detection performance using the visual form.

4 Results and Discussion

In this section, we describe our experiment results. Experiments using the proposed approach were performed on both the UNSW-NB15 and NSL_KDD datasets. First, we present results of the dimensionality reduction study using the autoencoder and PCA techniques, respectively. We project the extracted data to lower dimension spaces, ranging from 2 to 30, to find the number of dimensions that produced reasonably good performance for all classifiers. The classifiers that were used in the experiments were the k-Nearest Neighbors (kNN) classifier, the Multi-Layer Perceptron (MLP) classifier and the Decision Tree (DT) classifier. Subsequently, we present examples of results that demonstrate observable visual characteristics, which were obtained by projecting the data with the best number of dimensions into a 3D visual space.

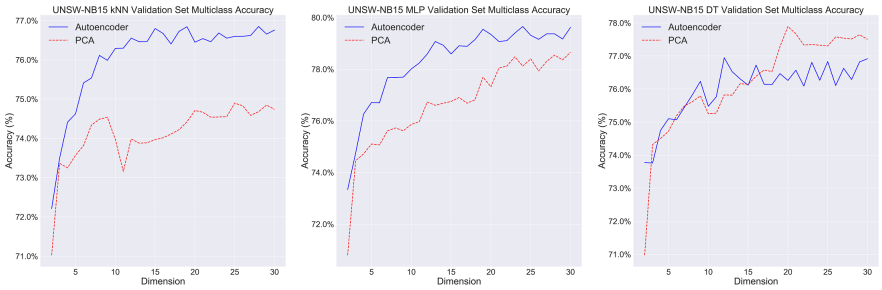
4.1 Results for the UNSW-NB15 Dataset

Fig. 2(a)-(c) and Fig. 3(a)-(c), depict results of accuracy trends that were obtained when the three different classifiers were applied to the validation set for binary classification and multiclass classification, respectively. The difference between binary classification and multiclass classification is that in binary classification, network traffic instances were either classified as normal traffic or abnormal traffic. Whereas in multiclass classification, the machine learning model was used to classify all categories of network traffic (e.g., normal traffic, DoS attacks, worm attacks, U2R attacks, exploits, etc.).



(a) (b) (c)

Fig. 2. UNSW-NB15 binary classification accuracy trends on the validation set using the (a) kNN classifier; (b) MLP classifier; (c) DT classifier.



(a) (b) (c)

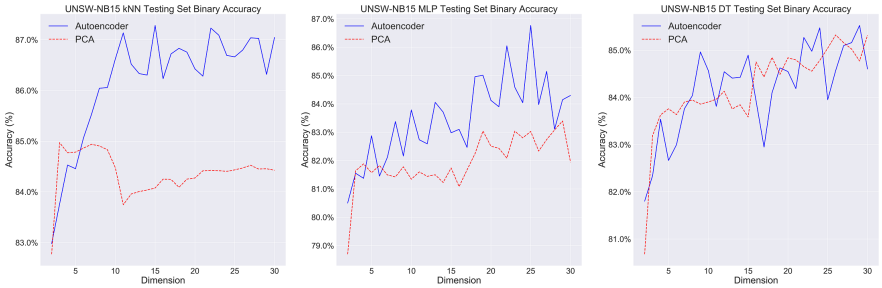
Fig. 3. UNSW-NB15 multiclass classification accuracy trends on the validation set using the (a) kNN classifier; (b) MLP classifier; (c) DT classifier.

From Fig. 2 and Fig. 3, we can see that accuracy trends in the kNN and MLP results are more stable than accuracy trends in the DT results. In general, accuracy increases with the number of dimension. It can also be seen that the performance of dimensionality reduction based on autoencoder outperforms PCA for the kNN and MLP classifiers, since they achieve higher accuracy results when autoencoder is used. However, for the DT classifier it is less obvious as to which dimensionality reduction technique is better. Overall, autoencoder performs better than PCA in relation to dimensionality reduction and accuracy. From the figures, one can see that the trend is such that the intrusion detection accuracy typically increases as the dimensions increase, then remains relatively stable once the number of dimensions reaches a certain value.

From these results, our purpose is to find the number of dimensions at which all three classifiers perform reasonably well for both the autoencoder and PCA techniques. From Fig. 2 and Fig. 3, we can see that the value of 20 is a reasonable choice for the number of dimensions because at this value almost all classifiers are near their peak accuracy for PCA and autoencoder. To confirm our choice of the intrinsic number of dimensions, we also present results showing accuracy trends when experiments were conducted on the test set. This is shown in Fig. 4(a)-(c) and Fig. 5(a)-(c) for binary classification and multiclass classification, respectively.

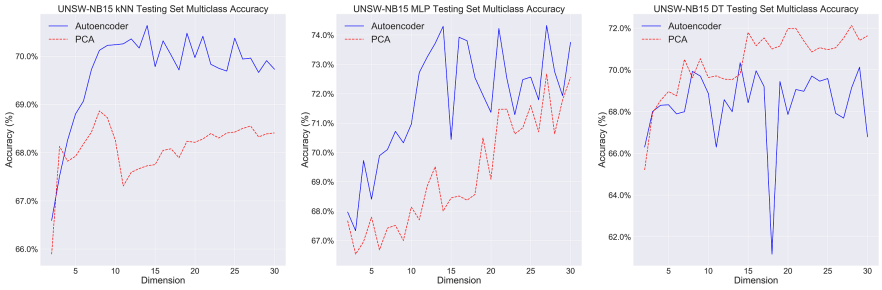
From Fig. 4(a)-(c) and Fig. 5(a)-(c), it can be observed that although there is a greater degree of fluctuation, accuracy trends in the test set show similar characteristics to those in the validation set for both binary and multiclass classification. Overall, autoencoder still performs better than PCA for dimensionality reduction. In Fig. 5(c), there is an abrupt drop in accuracy when the number of dimensions is 18. This may be due to over-fitting of the DT classifier. Nevertheless, the other accuracy trends as shown in the figures are reasonable.

The value of 20 is still a reasonably good choice for the best number of dimensions, when considering all the accuracy trends in Fig. 4(a)-(c) and Fig. 5(a)-(c). Since in our experiments autoencoder performs better than PCA for



(a) (b) (c)

Fig. 4. UNSW-NB15 binary classification accuracy trends on the test set using the (a) kNN classifier; (b) MLP classifier; (c) DT classifier.



(a) (b) (c)

Fig. 5. UNSW-NB15 multiclass classification accuracy trends in testing set using the (a) kNN classifier; (b) MLP classifier; (c) DT classifier.

dimensionality reduction, we used the autoencoder data that was reduced to 20 dimensions for 3D visualization. For projecting to 3D space, we used the PCA technique for the visualization. The reason for this is because unlike the autoencoder technique, PCA transformation can be inversed. Hence, when examining certain areas of data in 3D space, this data can be inversed and examined in higher dimensional space.

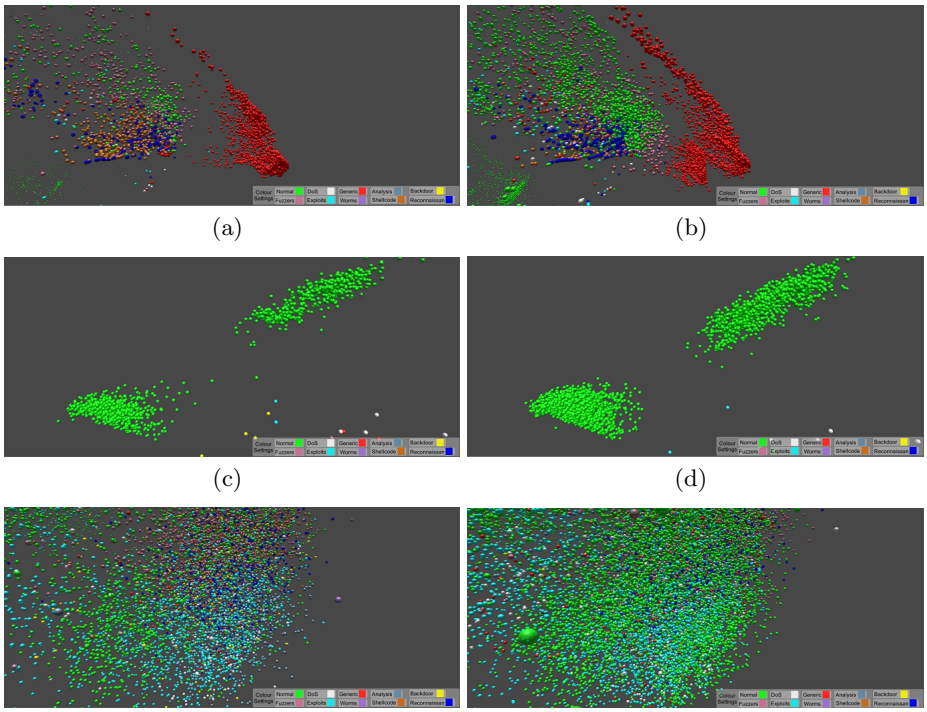
From the visualization results, we show that key visual features of the UNSW-NB15 datasets are comparable with those presented in related work [17]. Zong et al. [17] showed that most generic attacks are visually clustered together in both the training and test sets. In addition, there are some clusters that contain only normal connections in both the training and test sets. Their results also showed that the main difficulty encountered by machine learning intrusion detection methods using the UNSW-NB15 dataset, comes from clusters where different categories of traffic are densely mixed. These three features can also be observed in our visualization experiment as shown in Fig. 6.

From the visual representations shown in Fig. 6(a) and Fig. 6(b), it can clearly be seen in the visual representation that most generic attacks are grouped together in the training and test sets. We can also find homogeneous clusters of normal connections in the training and test sets as shown in Fig. 6(c) and Fig. 6(d), respectively. In addition, Fig. 6(e) and Fig. 6(f) respectively show sections that contain a mixture of network traffic in the training and test sets. Despite the visualization results in our experiment differing from the results in [17], the visual features are similar. This affirms the validity of our 3D visualization results. An obvious visual characteristic of UNSW-NB15 is that the training set and the test set have similar characteristics in 3D space. This implies that the original data in the training and test sets are similar in nature. This characteristic is the reason why we can choose the best dimension that can produce relatively good results in both validation and test sets.

4.2 Results for the NSL_KDD Dataset

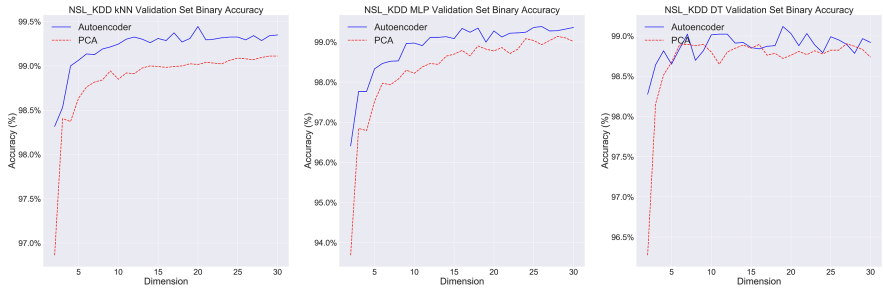
The same experiments that were performed on the UNSW-NB15 dataset were also done on the NSL_KDD dataset. These results are presented here.

From Fig. 7(a)-(c) and Fig. 8(a)-(c), it can clearly be seen that results of accuracy trends for the NSL_KDD dataset, using all three classifiers for binary and multiclass classification, share similar characteristics with the UNSW-NB15 dataset. Accuracy initially increases with the number of dimensions, then remains relatively stable after a certain number of dimensions. In relation to dimensionality reduction for the NSL_KDD dataset, autoencoder is still better in terms of performance compared with PCA. The DT classifier again has more fluctuations than the other two classifiers. Similar to the UNSW-NB15 dataset results, the kNN and MLP classifiers favor autoencoder when it comes to reducing the dimensionality of data. Considering the results in Fig. 7(a)-(c) and Fig. 8(a)-(c), the value of 25 is a reasonable choice as the best number of dimensions to achieve good performance for the NSL_KDD dataset. In an attempt to verify this, accuracy trends of the test sets are shown in Fig. 9 and Fig. 10.



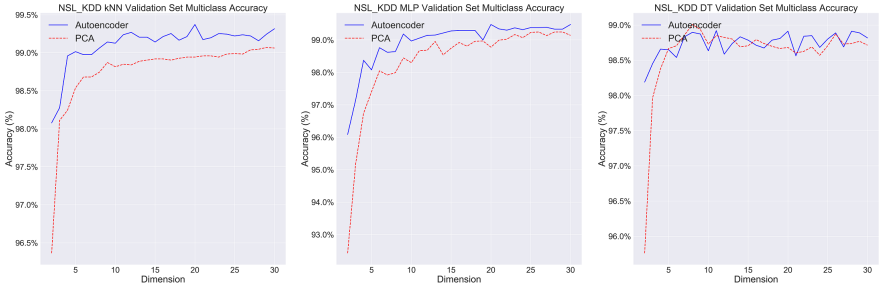
(a) (b) (c) (d) (e) (f)

Fig. 6. 3D visualization results from the UNSW-NB15 dataset showing (a) clusters of generic attacks in the training set; (b) clusters of generic attacks in the test set; (c) homogeneous clusters containing only normal connections in the training set; (d) homogeneous clusters containing only normal connections in the test set; (e) clusters containing mixed traffic in the training set; (f) clusters containing mixed traffic in the test set.



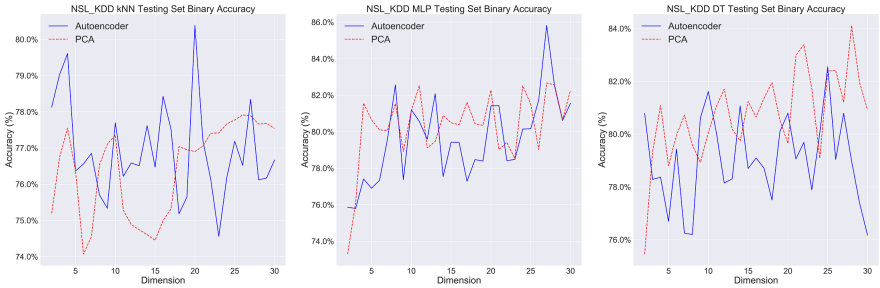
(a) (b) (c)

Fig. 7. NSL_KDD binary classification accuracy trends on the validation set using the (a) KNN classifier; (b) MLP classifier; (c) DT classifier.



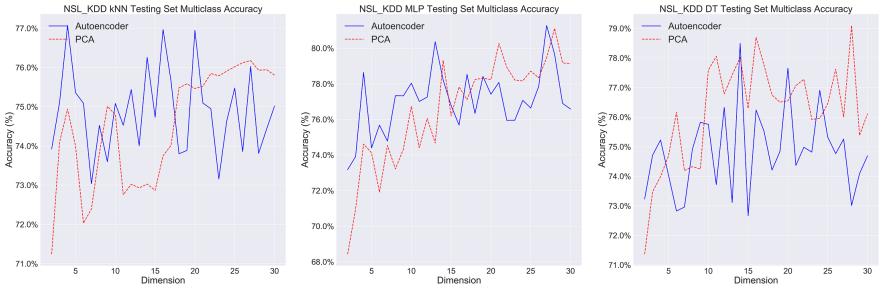
(a) (b) (c)

Fig. 8. NSL_KDD multiclass classification accuracy trends on the validation set using the (a) KNN classifier; (b) MLP classifier; (c) DT classifier.



(a) (b) (c)

Fig. 9. NSL_KDD binary classification accuracy trends in the test set using the (a) kNN classifier; (b) MLP classifier; (c) DT classifier.



(a) (b) (c)

Fig. 10. NSL_KDD multiclass classification accuracy trends in the test set using the (a) kNN classifier; (b) MLP classifier; (c) DT classifier.

As can be seen from the results in Fig. 9(a)-(c) and Fig. 10(a)-(c), the accuracy obtained from the NSL_KDD test data do not show obvious trends, because the values fluctuate wildly with respect to the number of dimensions. Hence, the best value for the number of dimensions that was selected in the validation set cannot be verified from results of the test set. Therefore, for the purpose of our experiment as long as the dimension was not too small, i.e. larger than 5, there was no significant difference in choosing the best number of dimensions. The reason why this situation occurs can be explained from the 3D visualization results. In particular, the difficulty in intrusion detection when using the NSL_KDD dataset lies in the fact that the test set contains previously unknown attacks [17]. In view of the accuracy trends in the validation set, we first reduce the number of dimensions to 25 using autoencoder and then use PCA to visualize the data. Examples of visualization results are shown in Fig. 11.

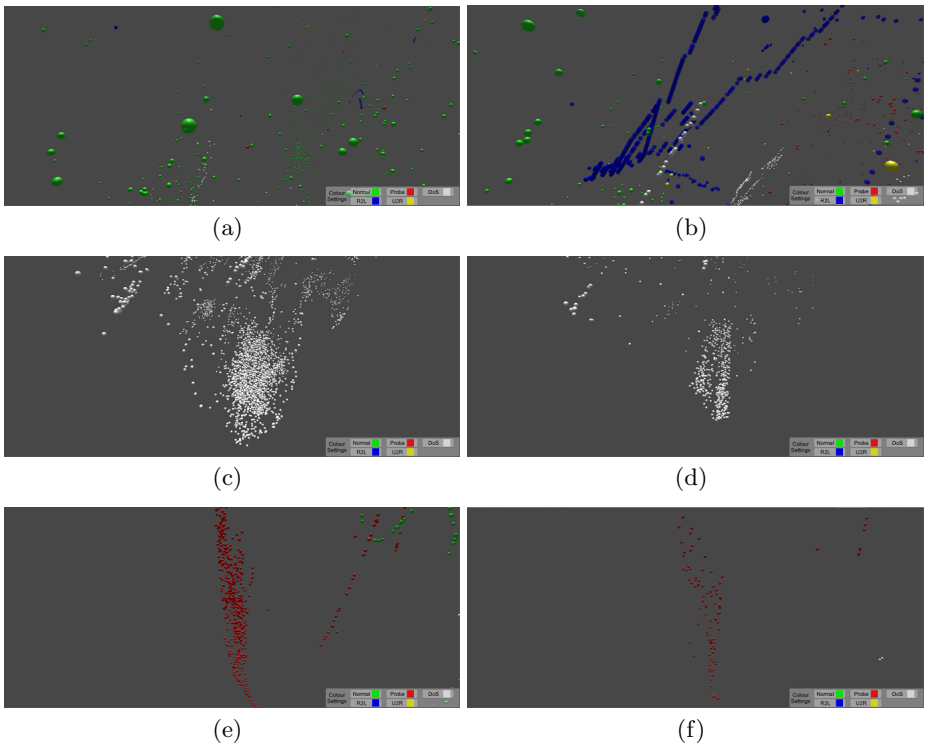


Fig. 11. 3D visualization results from the NSL_KDD dataset showing (a) various attacks in the training set; (b) previously unknown attacks in the test set; (c) homogeneous clusters of DoS attacks in the training set; (d) homogeneous clusters of DoS attacks in the test set; (e) clusters of probe attacks in the training set; (f) clusters of probe attacks in the test set.

From visual inspection of the 3D visualization results in Fig. 11, we can see that there are attacks that only exist in the test set but are not in the training set. This can be seen when comparing the visual results in Fig. 11(a) and Fig. 11(b), as the attack characteristics in Fig. 11(b) contain previously unknown attacks when compared with Fig. 11(a). This difference is the main reason why there are obvious fluctuations in the results presented in Fig. 9 and Fig. 10. Consequently, for the NSL_KDD test set, no good value for the number of dimensions to produce optimal performance could be identified. This situation is different from the UNSW-NB15 dataset and shows that there are obvious differences in the datasets, which can easily be seen in the visual representation. From the visualization results, we can also find highly homogeneous clusters that contain the same type of network traffic in both the training and test sets. For example, it can be seen that both Fig. 11(c) and Fig. 11(d) contain clusters with only DoS attacks, and also Fig. 11(e) and Fig. 11(f) which show sections that contain mainly probe attacks. Similar visual characteristics in the NSL_KDD dataset have also been reported in Zong et al. [17].

5 Conclusion

This paper investigates the effects of two dimensionality reduction techniques on network intrusion detection datasets. The experiment results show that the autoencoder technique typically performs better than the PCA technique for both the UNSW-NB15 and NSL_KDD datasets. For UNSW-NB15 dataset, we were able to identify a specific number of dimensions at which the classifiers produced relatively good results in both the validation and test sets. This is likely due to high similarity between data in the training and test sets. On the other hand, we could not easily identify such a value for the NSL_KDD dataset, despite clear accuracy trends in the validation set. From visual inspection of the 3D visualization results, the reason for this is likely due to the fact that data in the training and test sets of the NSL_KDD dataset contain significant differences, e.g., previously unknown attacks which were not in the validation set are present in the test set. As such, this paper also demonstrates how 3D visualization can facilitate the understanding of intrusion detection results, as visual patterns in a dataset can be identified through visual inspection of the data.

References

1. M. Angelini, N. Prigent, and G. Santucci. PERCIVAL: proactive and reactive attack and response assessment for cyber incidents using visual analytics. In L. Harrison, N. Prigent, S. Engle, and D. M. Best, editors, *2015 IEEE Symposium on Visualization for Cyber Security, VizSec 2015, Chicago, IL, USA, October 25, 2015*, pages 1–8. IEEE Computer Society, 2015.
2. N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer. SMOTE: synthetic minority over-sampling technique. *J. Artif. Intell. Res.*, 16:321–357, 2002.

3. E. de la Hoz Correa, E. de la Hoz Franco, A. Ortiz, J. Ortega, and B. Prieto. PCA filtering and probabilistic SOM for network intrusion detection. *Neurocomputing*, 164:71–81, 2015.
4. A. Y. Javaid, Q. Niyaz, W. Sun, and M. Alam. A deep learning approach for network intrusion detection system. *ICST Trans. Security Safety*, 3(9):e2, 2016.
5. A. Lakhina, M. Crovella, and C. Diot. Diagnosing network-wide traffic anomalies. In R. Yavatkar, E. W. Zegura, and J. Rexford, editors, *Proceedings of the ACM SIGCOMM 2004 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communication, August 30 - September 3, 2004, Portland, Oregon, USA*, pages 219–230. ACM, 2004.
6. W. Lin, S. Ke, and C. Tsai. CANN: an intrusion detection system based on combining cluster centers and nearest neighbors. *Knowl.-Based Syst.*, 78:13–21, 2015.
7. S. Liu, X. Wang, M. Liu, and J. Zhu. Towards better analysis of machine learning models: A visual analytics perspective. *Visual Informatics*, 1(1):48–56, 2017.
8. J. McHugh. Testing intrusion detection systems: a critique of the 1998 and 1999 DARPA intrusion detection system evaluations as performed by lincoln laboratory. *ACM Trans. Inf. Syst. Secur.*, 3(4):262–294, 2000.
9. S. McKenna, D. Staheli, C. Fulcher, and M. D. Meyer. Bubblenet: A cyber security dashboard for visualizing patterns. *Comput. Graph. Forum*, 35(3):281–290, 2016.
10. N. Moustafa and J. Slay. UNSW-NB15: a comprehensive data set for network intrusion detection systems (UNSW-NB15 network data set). In *2015 Military Communications and Information Systems Conference, MilCIS 2015, Canberra, Australia, November 10-12, 2015*, pages 1–6. IEEE, 2015.
11. N. Moustafa, J. Slay, and G. Creech. Novel geometric area analysis technique for anomaly detection using trapezoidal area estimation on large-scale networks. *IEEE Transactions on Big Data*, pages 1–1, 2018.
12. P. E. Rauber, S. G. Fadel, A. X. Falcão, and A. C. Telea. Visualizing the hidden activity of artificial neural networks. *IEEE Trans. Vis. Comput. Graph.*, 23(1):101–110, 2017.
13. R. Sommer and V. Paxson. Outside the closed world: On using machine learning for network intrusion detection. In *31st IEEE Symposium on Security and Privacy, S&P 2010, 16-19 May 2010, Berleley/Oakland, California, USA*, pages 305–316. IEEE Computer Society, 2010.
14. D. Staheli, T. Yu, R. J. Crouser, S. Damodaran, K. Nam, B. D. O'Gwynn, S. McKenna, and L. Harrison. Visualization evaluation for cyber security: trends and future directions. In K. Whitley, S. Engle, L. Harrison, F. Fischer, and N. Prigent, editors, *Proceedings of the Eleventh Workshop on Visualization for Cyber Security, Paris, France, November 10, 2014*, pages 49–56. ACM, 2014.
15. G. Wang, J. Hao, J. Ma, and L. Huang. A new approach to intrusion detection using artificial neural networks and fuzzy clustering. *Expert Syst. Appl.*, 37(9):6225–6232, Sept. 2010.
16. Y. Wang, H. Yao, and S. Zhao. Auto-encoder based dimensionality reduction. *Neurocomputing*, 184:232–242, 2016.
17. W. Zong, Y. Chow, and W. Susilo. A 3d approach for the visualization of network intrusion detection data. In A. Sourin, O. Sourina, C. Rosenberger, and M. Erdt, editors, *2018 International Conference on Cyberworlds, CW 2018, Singapore, October 3-5, 2018*, pages 308–315. IEEE, 2018.
18. W. Zong, Y. Chow, and W. Susilo. A two-stage classifier approach for network intrusion detection. In C. Su and H. Kikuchi, editors, *Information Security Practice and Experience - 14th International Conference, ISPEC 2018, Tokyo, Japan,*

September 25-27, 2018, Proceedings, volume 11125 of *Lecture Notes in Computer Science*, pages 329–340. Springer, 2018.