Comparison of network intrusion detection performance using feature representation *

Daniel Pérez[0000−0002−2173−3364], Serafín Alonso[0000−0003−3467−4938], Antonio Morán[0000−0002−2762−6949], Miguel A. Prada[0000−0002−1563−1556], Juan José Fuertes[0000−0001−9023−0341], and Manuel Domínguez[0000−0002−3921−1599]

University of León. Campus de Vegazana s/n. 24007, León (Spain).
{dperl,saloc,a.moran,ma.prada,jj.fuertes,manuel.dominguez}@unileon.es

Abstract. Intrusion detection is essential for the security of the components of any network. For that reason, several strategies can be used in Intrusion Detection Systems (IDS) to identify the increasing attempts to gain unauthorized access with malicious purposes including those base on machine learning. Anomaly detection has been applied successfully to numerous domains and might help to identify unknown attacks. However, there are existing issues such as high error rates or large dimensionality of data that make its deployment difficult in real-life scenarios. Representation learning allows to estimate new latent features of data in a low-dimensionality space. In this work, anomaly detection is performed using a previous feature learning stage in order to compare these methods for the detection of intrusions in network traffic. For that purpose, four different anomaly detection algorithms are applied to recent network datasets using two different feature learning methods such as principal component analysis and autoencoders. Several evaluation metrics such as accuracy, F1 score or ROC curves are used for comparing their performance. The experimental results show an improvement for two of the anomaly detection methods using autoencoder and no significant variations for the linear feature transformation.

Keywords: Anomaly detection · Feature representation · Network intrusion detection.

1 Introduction

The great advances in network technologies entail a rise of the complexity of network attacks. For this reason, network intrusion detection plays a key role in the security of information systems and, therefore, it has become an active research area [1,5]. In this context, intrusion can be considered as an attempt to compromise the security of a computer or network elements. It can be of

* This research was supported by the Regional Government of Castilla y León and the European Regional Development Fund under project LE045P17.
The final publication is available at Springer via https://doi.org/10.1007/978-3-030-20257-6_40

two types: external, where unauthorized users try to gain access to the system, and internal, which are more frequent and users with different permission roles could have access to resources of the system. Moreover, different situations can be labelled as intrusions, ranging from worms that try to propagate through the network without authorization to denial of service (DoS) which focus on disrupting the resources of a system on a network. Intrusion detection systems (IDS) are devices that monitor a network in order to find any malicious activity. They are commonly classified in different types: Host-based IDS (HIDS) that analyses the internals of an individual system and Network-based IDS (NIDS) that monitors traffic between the devices of a network trying to find suspicious patterns [3]. The techniques for intrusion detection include misuse-based approaches that look for known malicious activity mostly using signatures and anomaly-based approaches which consider as an anomaly any intrusive action and would potentially detect unknown intrusions. Although both techniques have been extensively studied [5], misuse detectors are much commonly deployed in real systems.

Anomaly detection methods attempt to estimate a model of the normal behaviour of data according to a specific criteria and find patterns deviated from the resulting model [6]. These methods have been extensively applied to network intrusion detection [1,5,3]. However, their application in real scenarios has traditionally been unusual because it implies to deal with some issues [25]. For instance, network traffic presents large variability so anomalous behaviour can sometimes be related to performance, or high false positives also involve the evaluation of potential alarms which are actually normal situations.

Besides, there are other different aspects such as labelling or scaling the data that improve the success of these techniques. In addition, feature selection helps to reduce complexity and understand data interpretation. Although there are different existing strategies, representation learning and deep learning have provided enormous advances in several areas [2] such as computer vision or natural language processing.

In this work, a comparison of anomaly detection tasks is made using a feature representation of data for network intrusion detection. For that purpose, different methods of anomaly detection are compared in order to evaluate how feature transformation affects them, using four recent network datasets that provide real situations. The organization of the paper is structured as follows: in section 2, different anomaly detection approaches are described and some examples for their application in intrusion detection are mentioned; in section 3, the method used is illustrated; in section 4, the datasets, the configuration of the experiments and the results are discussed and, finally, the conclusions are summarized in section 5.

2 Related work

There have been numerous efforts to survey available techniques for the implementation of anomaly detection tasks [6], specifically applied to network intrusion detection [5,3,1]. Common approaches can be grouped into categories

depending on how the method detects outliers. Next, they are briefly reviewed and some related works about network intrusion detection are mentioned.

Generally statistical-based approaches assume normal data points are generated from a Gaussian distribution. The estimation of their parameters can be sensitive to outliers so that robust estimators were proposed, like minimum covariance determinant [22]. There are examples that use statistical approaches for intrusion detection systems such as HIDE [29] which uses statistical modelling along with neural network classifiers or PAYL [28] that computes statistical parameters of the application payload, estimated from normal behaviour using a 1-gram model and then evaluated in terms of Mahalanobis distances. Other strategy to detect anomalies is based on distance-based approaches considering data instances with N features as a N-dimensional vector. For instance, One-Class Support Vector Machines (OC-SVM) constructs a hyperplane that aims to separate with a maximum margin the normal instances from the anomalous ones. Moreover, clustering methods [10] like K-means can also be used where the anomaly score is evaluated using the distance between new data points and computed centroids. Related examples for intrusion detection include Khan et al. [12], who proposed a combination of a hierarchical clustering with a SVM classification or Muda et al. [20] that computes initially K-means cluster centroids and then applies Naive Bayes classification in the final stage to distinguish between five different classes. Other proposals use ensemble-based methods such as bootstrap aggregation (bagging) or boosting that combine individual results of multiple classifiers to achieve a final decision. Similarly, Isolation Forest [15] creates an ensemble of decision trees isolating anomalies instances.

Since the performance of machine learning methods is generally affected by the number of the data dimensions, there are algorithms to select and transform data features providing another representation of the data. On one hand, irrelevant features can be eliminated in terms of information redundancy removal and accuracy improvement. Several feature selection methods have been proposed in the intrusion detection domain [7]. Some algorithms use an optimization criteria (wrapper), others compute independent features (filter) and hybrid methods try to combine both approaches for a better performance. On the other hand, feature transformation algorithms estimates a latent space that provides a new representation of the data. Dimensionality reduction techniques can be used for that purpose, like Principal Component Analysis (PCA) which computes linearly the principal components with largest variance. The use of autoencoders as dimensionality reduction tool was proposed in [9] whose low-dimensional representation can improve the performance of different tasks. Although there are other dimensionality reduction techniques, for instance those based on neighbour embeddings or spectral methods [14], the active recent research in deep learning has provided an increasing interest in approaches related to representation learning [2].

There are similar works that propose approaches related to neural networks, deep learning and anomaly detection. Previous examples include the combination of deep belief network with linear one-class SVM [8] for unsupervised

anomaly detection of high-dimensional data, discussing the performance of the hybrid model. On the other hand, a model composed by a deep autoencoder and a variant of one-class SVM using random Fourier features is introduced in [21]. An ensemble of autoencoders, called Kitsune [18], is proposed for network intrusion detection to differentiate between normal and abnormal traffic patterns. Finally, a NIDS is proposed in [11] that uses two-stage process with a sparse auto-encoder for learning features and soft-max regression, using labelled data for classification. Although these methods use networks not only for anomaly detection but also for dimensionality reduction of data features, in this work the performance of auto-encoder is studied with respect to different approaches for anomaly detection to evaluate which one is more suitable and it also uses recent datasets to test more realistic scenarios.

3 Proposed method

In this work, a feature learning stage is combined with well-known anomaly detection methods to detect network intrusions. Taking into account the complexity of the application area, real traffic data should be considered in order to provide more realistic scenarios for the analysis. The variables included in data are usually essential features such as protocol, service, flags, bytes between source and destination or their IP addresses and, in some cases, additional ones including statistical or aggregation measures like sum of connections or mean values. In this case, an intrusion is considered as an individual point labelled in data which is a simplification of the consequences provoked by a network attack.

A preparation stage for preprocessing data should be done. In that stage, the transformation of categorical attributes like protocol type into numeric values is performed and also normalization of data provides scaling of the features so that they are between similar ranges of values. Besides, the variables with a few unique values can be transformed into binary values using one hot encoding. Finally, data are split into train, validation and test sets.

The feature representation estimates a reduced latent space of the data by means of unsupervised learning, that is, without using data labels of the status of the network. As a baseline, the widely-used method PCA is used for reducing the dimensionality of the input data by computing a feature representation. Also, a deep auto-encoder is used with training data to compute the latent representation in the bottleneck so that the encoder provides the representation of new data. The number of the low-dimensional space is considered taking into account a trade-off between a significant reduction of the dimensionality of data without an excessive loss of information. The same number is used in the transformation of the reduced features for comparison purposes of the methods.

Once the feature transformation is done, the anomaly detection methods are trained using normal instances from the labels of the data for train and validation sets. Then the prediction of test data is performed after a data transformation into the resulting latent space. A flowchart diagram for the architecture of the method is represented in Fig. 1.

The comparison of the resulting performance is evaluated with several measures commonly used for binary classification. The well-known evaluation metrics are accuracy, precision, recall, F1 score, and Receiver Operating Characteristic (ROC) curves. The accuracy measures the fraction of the instances correctly categorized; precision denotes the proportion of true positives between all the positive predicted ones, and recall refers to the ratio of true positive between the real positive instances. Furthermore, F1 score helps to consider both precision and recall to evaluate a model computing their harmonic average. ROC curve shows false positive rates between true positive rates for several thresholds and area under curve (AUC) measures the capacity of distinguish between the two classes.

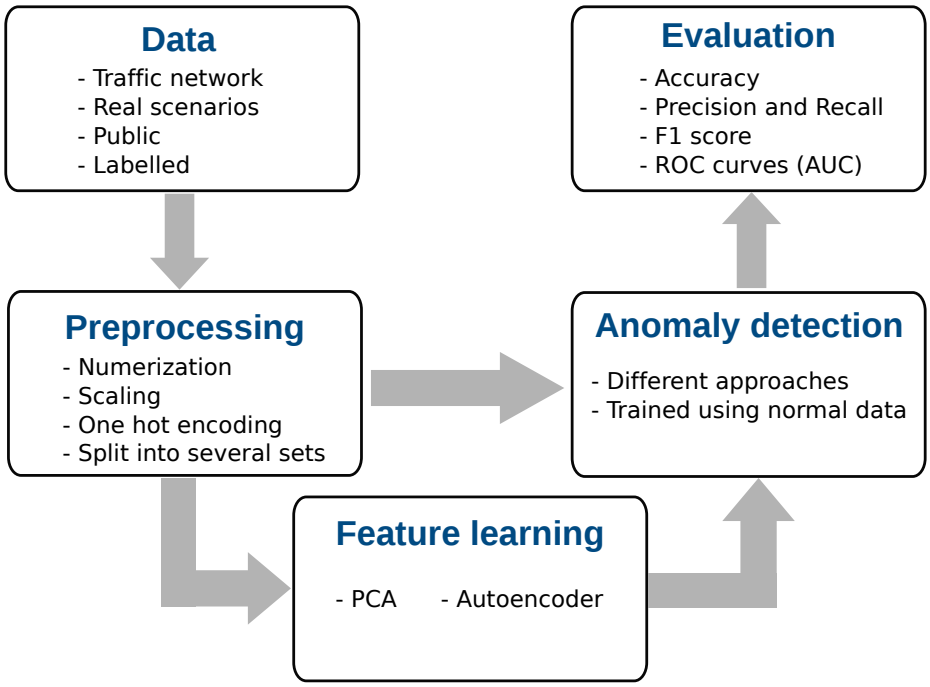

Fig. 1. Description of the architecture for the proposed method.

4 Experimental methodology

Several experiments were performed using publicly available datasets based on real traffic for network intrusion detection. A previous feature learning stage was applied and various evaluation metrics were used in order to compare the performance.

4.1 Datasets

In the majority of the previous works reviewed (see section 2) about network intrusion detection, only a couple of datasets are widely used for the assesssment of the detection systems [1,3], i.e. DARPA 98 and 99 from MIT's Lincoln Laboratory and KDDCup'99. However, these datasets have several shortcomings which have already been identified in the literature [17,16,27]. This leads to consider that other datasets might be more suited to evaluate the detection of contemporary network attacks. For that reason, the following recent datasets have been used in the experiments in order to consider more realistic situations:

- **UNSW-NB15** [19] possess a hybrid of the real modern normal traffic and synthesized attack activities. It was generated using an attack automatic generation tool called IXIA PerfectStorm.
- **NSL-KDD** [27] was created in order to improve the KDDCup'99 dataset. Although the dataset still suffers some problems to be considered a complete representative of modern networks, it can be used as a reference for comparison purposes because of its wide use.
- **CIC-IDS-2017** [24] covers updated attacks with more than 80 features and labelled for benign and intrusive flows. Concretely, the data used here correspond exactly to working hours of Wednesday.
- **Kyoto** [26], built on 3 years of real traffic data (Nov. 2006–Aug. 2009) which were obtained from different kinds of honeypots.

All datasets include labels about normal and different types of attacks occurred in the network which are used for training and evaluation of anomaly detection tasks. A description of the datasets such as number of instances, the attributes or dimensions obtained after one hot encoding of some of the features and the percentage of anomalies is detailed in Table 1.

Table 1. Description of the network datasets.

Data	Instances	Attributes	Dimensions	% of anomalies
UNSW-NB15	257673	45	230	63.9
NSL-KDD	148517	43	143	48.1
CIC-IDS-2017	691695	83	83	36.4
Kyoto	364725	24	46	82.9

Preprocessing The categorical features included in data were transformed to numerical values, in some cases using a one hot encoding to obtain a set of binary variables for those features with few categories. For that reason, the number of dimensions can be augmented with respect to original attributes of data. The transformed features depend on the dataset, but there are some in common for

the majority of datasets, for example service, protocol type or flag. Additionally, a min-max scaling were made so that each feature is scaled to a range of values between [0,1] on the training set and then also transform the validation and test data. Despite the variety of the intrusions labelled in data, they are all grouped only into one category, that is, are considered only two classes (normal and anomaly) in the analysis.

4.2 Experiments

First, four methods are applied for anomaly detection where only normal instances are used for training the different methods. These methods that were used are:

- **Local Outlier Factor (LOF)**: [4] assigns to each object a degree about how it is isolated with respect to a specific neighbourhood. The number of the k-nearest neighbours selected after several tests is set to 60 for all datasets used in the experiments.
- **One-Class Support Vector Machine (OC-SVM)**: only uses one class for estimating a model and detects new data different from that class as outliers [23]. The kernel used in this work is a radial basis function (RBF) with $\gamma = 0.1$, fixed experimentally.
- **Isolation Forest (IF)**: creates an ensemble of trees that isolate anomalies instead of fitting normal instances, which is a different approach for outlier detection [15].
- **Robust Covariance (RC)**: implements a minimum covariance determinant which is a highly robust algorithm for estimating covariance matrix in multivariate data [22].

For computing the latent representation, the dimensionality of data is reduced using either Principal Component Analysis (PCA) as the linear baseline method and the encoder obtained from an autoencoder. The design of the neural network can be esentially considered as a common deep autoencoder. The input layer has a size equal to the dimensionality of input data which is reduced using several hidden layers using rectifier linear unit (ReLU) as activation functions except for the last layer where a sigmoid function is used. The optimization stage is performed using the Adam algorithm [13]. The selected batch size is 256 and epochs for training have been set to 700, they are experimentally fixed according to the datasets used. The details for the representation learning stage are described in Table 2. The latent dimension computed using PCA has the same dimension of the bottleneck of the autoencoder for comparison purposes. The layers of the encoding were selected in order to obtain a significant reduction of the dimensionality of data.

4.3 Results and discussion

The results of the experiments are presented in this section. The evaluation measures of the anomaly detection methods applied to the network datasets are

Table 2. Details of the feature transformation.

Data	Encoding layers	PCA dim.
UNSW-NB15	{230, 120, 60, 20}	20
NSL-KDD	{143, 100, 80, 20}	20
CIC-IDS-2017	{83, 80, 40, 20}	20
Kyoto	{46, 40, 20, 5}	5

detailed in Table 3. In this table, the accuracy, precision, recall and F1 score indicate the performance of the corresponding method for each dataset, also including the previous feature learning stage using PCA and encoder network. The best resulting F1 score for each dataset is highlighted between all the methods used. Furthermore, area under curve (AUC) and ROC curves are shown in Fig. 2 in a matrix form where the rows correspond to each dataset and the columns the method applied. In case of equal F1 scores, the AUC value is considered for selecting the best one.

Several changes can be observed in the performance of anomaly detection tasks as a result of feature learning stage. The most significant improvement is produced using One-Class SVM method, where the use of the auto-encoder computing a feature representation shows better evaluation metrics for all datasets used in the experiments. In addition, the auto-encoder representation also produces small enhancements in the results using Local Outlier Factor, as it is shown in the Fig. 2.

However, the feature representation barely affects the effectiveness for anomaly detection using the Isolation Forest and Robust Covariance methods. There are only improvements for both methods using CIC-IDS-2017 dataset, shown by the values of F1 scores (see Table 3). Moreover, in some cases it is preferable the application of these two methods using the original data without any feature learning.

On the other hand, PCA transformation produces generally similar results to original data and, in some cases even worse than original features. There are only a few cases where the representation computed by PCA overcomes the rest. In these cases, the method used is Robust Covariance which seems to be the most suitable one to a previous PCA feature learning. This can reflect that linear techniques could only work in specific scenarios and they might be insufficient for a general type of analysis. Finally, it is remarkable that results from the experiments show in some cases a poor performance, for example Kyoto data using Local Outlier Factor.

5 Conclusions

Network intrusion detection is an active research area in a continuous development. Although there have been numerous efforts to address several challenges, anomaly-based approaches are sometimes difficult to be applied in real systems for intrusion detection.

Table 3. Performance of the proposed methods for intrusion detection.

		LOF				OC SVM				Isolation Forest				Robust Cov.			
		Acc.	Precision	Recall	F1	Acc.	Precision	Recall	F1	Acc.	Precision	Recall	F1	Acc.	Precision	Recall	F1
UNSW-NB15	-	0.83	0.96	0.72	0.82	0.57	0.97	0.22	0.36	0.55	0.96	0.19	0.31	0.45	0.41	0.01	0.01
	PCA	0.83	0.95	0.72	0.82	0.72	0.99	0.51	0.67	0.44	0.1	2e-3	4e-3	0.48	0.77	0.08	0.14
	Encoder	0.82	0.96	0.7	**0.82**	0.79	0.96	0.64	0.77	0.46	0.72	0.03	0.06	0.46	0.81	0.03	0.05
NSL-KDD	-	0.43	0.1	1e-3	1e-3	0.76	0.93	0.63	0.75	0.75	0.97	0.58	0.73	0.63	0.99	0.35	0.52
	PCA	0.42	1e-5	1e-5	1e-5	0.75	0.93	0.61	0.73	0.5	0.96	0.13	0.23	0.43	0.73	5e-3	0.01
	Encoder	0.43	0.04	3e-4	6e-4	0.92	0.92	0.95	**0.93**	0.62	0.98	0.35	0.51	0.43	0.25	2e-3	4e-3
CIC-IDS-2017	-	0.68	0.81	0.74	0.77	0.64	0.76	0.74	0.75	0.67	0.96	0.56	0.71	0.36	0.93	0.13	0.22
	PCA	0.63	0.79	0.68	0.73	0.62	0.73	0.75	0.74	0.59	0.94	0.46	0.62	0.45	0.99	0.24	0.39
	Encoder	0.72	0.76	0.89	**0.82**	0.6	0.7	0.8	0.75	0.74	0.87	0.75	0.81	0.72	0.87	0.72	0.79
Kyoto	-	0.46	0.62	0.31	0.41	0.54	0.8	0.33	0.47	0.5	0.99	0.19	0.32	0.49	0.99	0.16	0.28
	PCA	0.49	0.82	0.22	0.34	0.53	0.81	0.3	0.44	0.59	0.99	0.33	0.5	0.6	0.99	0.36	0.53
	Encoder	0.57	0.93	0.33	0.49	0.68	0.81	0.61	**0.7**	0.47	0.99	0.13	0.23	0.4	0.96	0.03	0.06

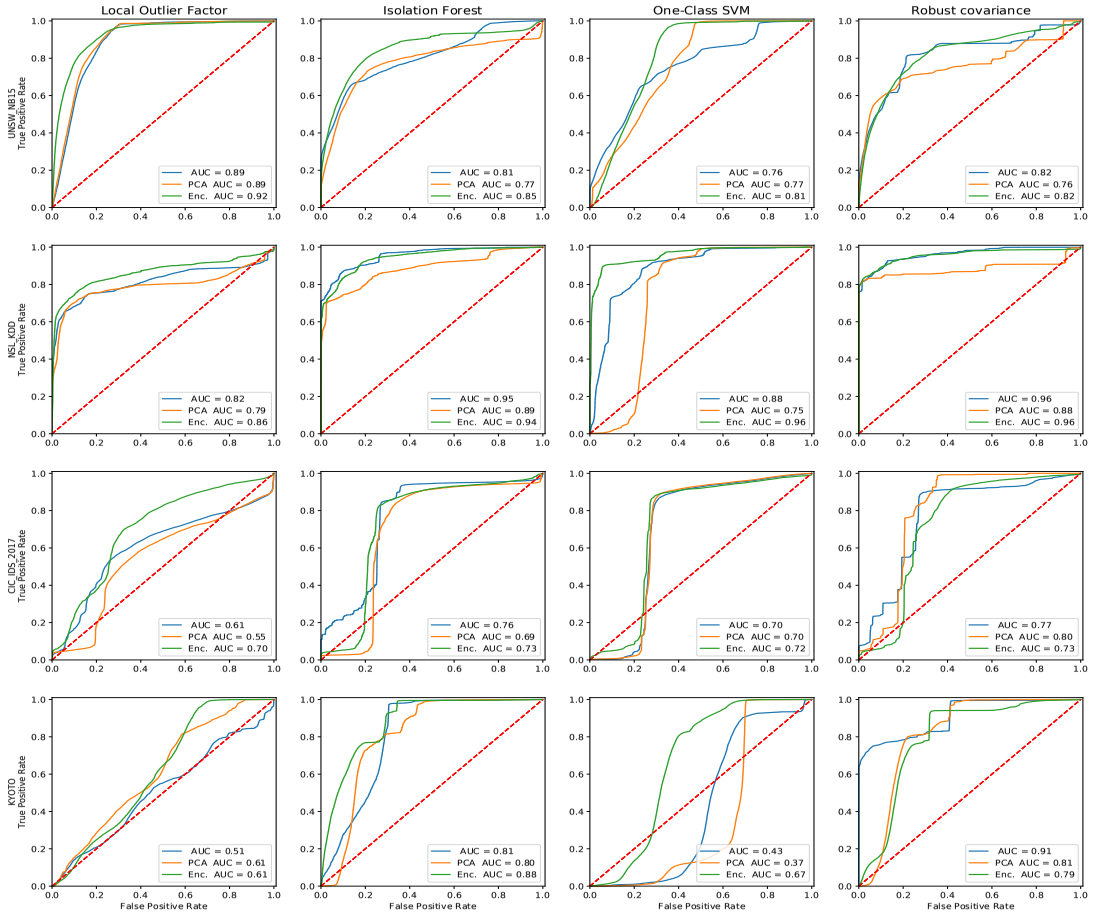

Fig. 2. ROC curves obtained from each method and datasets using the direct method and feature transformation using PCA and encoder network.

In this work, feature learning is used for network intrusion detection through its application as a previous stage to four different anomaly detection techniques applied to recent datasets. The methods used for computing the latent representation of data are PCA and the encoder part of an auto-encoder that introduces non-linearity. The main improvement for the datasets is shown for One-Class SVM method using the latent space computed by the auto-encoder. In contrast, PCA transformation does not show relevant enhancement in order to be applied as a previous feature learning stage.

Future work includes the study of other types of auto-encoders and techniques including different feature selection methods combined with more algorithms for anomaly detection that can help to improve the identification of intrusions.

References

1. Ahmed, M., Mahmood, A.N., Hu, J.: A survey of network anomaly detection techniques. Journal of Network and Computer Applications **60**, 19–31 (2016)
2. Bengio, Y., Courville, A., Vincent, P.: Representation learning: A review and new perspectives. IEEE transactions on pattern analysis and machine intelligence **35**(8), 1798–1828 (2013)
3. Bhuyan, M.H., Bhattacharyya, D.K., Kalita, J.K.: Network anomaly detection: methods, systems and tools. IEEE Communications Surveys & Tutorials **16**(1), 303–336 (2014)
4. Breunig, M.M., Kriegel, H.P., Ng, R.T., Sander, J.: LOF: identifying density-based local outliers. In: ACM sigmod record. vol. 29, pp. 93–104. ACM (2000)
5. Buczak, A.L., Guven, E.: A survey of data mining and machine learning methods for cyber security intrusion detection. IEEE Communications Surveys & Tutorials **18**(2), 1153–1176 (2015)
6. Chandola, V., Banerjee, A., Kumar, V.: Anomaly detection – a survey. ACM Computing Surveys **41**(3), 15:1–15:44 (July 2009). https://doi.org/10.1145/1541880.1541882
7. Chen, Y., Li, Y., Cheng, X.Q., Guo, L.: Survey and taxonomy of feature selection algorithms in intrusion detection system. In: International Conference on Information Security and Cryptology. pp. 153–167. Springer (2006)
8. Erfani, S.M., Rajasegarar, S., Karunasekera, S., Leckie, C.: High-dimensional and large-scale anomaly detection using a linear one-class SVM with deep learning. Pattern Recognition **58**, 121–134 (2016)
9. Hinton, G.E., Salakhutdinov, R.R.: Reducing the dimensionality of data with neural networks. Science **313**(5786), 504–507 (2006). https://doi.org/10.1126/science.1127647
10. Jain, A.K., Murty, M.N., Flynn, P.J.: Data clustering: a review. ACM computing surveys (CSUR) **31**(3), 264–323 (1999)
11. Javaid, A., Niyaz, Q., Sun, W., Alam, M.: A deep learning approach for network intrusion detection system. In: Proceedings of the 9th EAI International Conference on Bio-inspired Information and Communications Technologies (formerly BIONETICS). pp. 21–26. ICST (Institute for Computer Sciences, Social-Informatics and (2016)
12. Khan, L., Awad, M., Thuraisingham, B.: A new intrusion detection system using support vector machines and hierarchical clustering. The VLDB journal **16**(4), 507–521 (2007)

13. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. CoRR **abs/1412.6980** (2014), http://arxiv.org/abs/1412.6980
14. Lee, J.A., Verleysen, M.: Nonlinear dimensionality reduction. Springer Science & Business Media (2007)
15. Liu, F.T., Ting, K.M., hua Zhou, Z.: Isolation forest. In: In ICDM 08: Proceedings of the 2008 Eighth IEEE International Conference on Data Mining. IEEE Computer Society. pp. 413–422 (2008)
16. Mahoney, M.V., Chan, P.K.: An analysis of the 1999 DARPA/lincoln laboratory evaluation data for network anomaly detection. In: International Workshop on Recent Advances in Intrusion Detection. pp. 220–237. Springer (2003)
17. McHugh, J.: Testing intrusion detection systems: a critique of the 1998 and 1999 DARPA intrusion detection system evaluations as performed by Lincoln laboratory. ACM Transactions on Information and System Security (TISSEC) **3**(4), 262–294 (2000)
18. Mirsky, Y., Doitshman, T., Elovici, Y., Shabtai, A.: Kitsune: an ensemble of autoencoders for online network intrusion detection. arXiv preprint arXiv:1802.09089 (2018)
19. Moustafa, N., Slay, J.: UNSW-NB15: a comprehensive data set for network intrusion detection systems (UNSW-NB15 network data set). In: Military Communications and Information Systems Conference (MilCIS), 2015. pp. 1–6. IEEE (2015)
20. Muda, Z., Yassin, W., Sulaiman, M., Udzir, N.I., et al.: A k-means and naive bayes learning approach for better intrusion detection. Information technology journal **10**(3), 648–655 (2011)
21. Nguyen, M.N., Vien, N.A.: Scalable and interpretable one-class svms with deep learning and random fourier features. arXiv preprint arXiv:1804.04888 (2018)
22. Rousseeuw, P.J., Driessen, K.V.: A fast algorithm for the minimum covariance determinant estimator. Technometrics **41**(3), 212–223 (1999)
23. Schölkopf, B., Platt, J.C., Shawe-Taylor, J., Smola, A.J., Williamson, R.C.: Estimating the support of a high-dimensional distribution. Neural computation **13**(7), 1443–1471 (2001)
24. Sharafaldin, I., Lashkari, A.H., Ghorbani, A.A.: Toward generating a new intrusion detection dataset and intrusion traffic characterization. In: ICISSP. pp. 108–116 (2018)
25. Sommer, R., Paxson, V.: Outside the closed world: On using machine learning for network intrusion detection. In: 2010 IEEE symposium on security and privacy. pp. 305–316. IEEE (2010)
26. Song, J., Takakura, H., Okabe, Y., Eto, M., Inoue, D., Nakao, K.: Statistical analysis of honeypot data and building of kyoto 2006+ dataset for NIDS evaluation. In: Proceedings of the First Workshop on Building Analysis Datasets and Gathering Experience Returns for Security. pp. 29–36. ACM (2011)
27. Tavallaee, M., Bagheri, E., Lu, W., Ghorbani, A.A.: A detailed analysis of the KDD CUP 99 data set. In: Proceedings of the Second IEEE Symposium on Computational Intelligence for Security and Defence Applications 2009 (2009)
28. Wang, K., Stolfo, S.J.: Anomalous payload-based network intrusion detection. In: International Workshop on Recent Advances in Intrusion Detection. pp. 203–222. Springer (2004)
29. Zhang, Z., Li, J., Manikopoulos, C., Jorgenson, J., Ucles, J.: HIDE: a hierarchical network intrusion detection system using statistical preprocessing and neural network classification. In: Proc. IEEE Workshop on Information Assurance and Security. pp. 85–90 (2001)