2018

A two-stage classifier approach for network intrusion detection

Wei Zong
University of Wollongong, wz630@uowmail.edu.au

Yang-Wai Chow
University of Wollongong, caseyc@uow.edu.au

Willy Susilo
University of Wollongong, wsusilo@uow.edu.au

A two-stage classifier approach for network intrusion detection

Abstract

Network Intrusion Detection Systems (NIDS) are essential to combat security threats in network environments. These systems monitor and detect malicious behavior to provide automated methods of identifying and dealing with attacks or security breaches in a network. Machine learning is a promising approach in the development of effective NIDS. One of the problems faced in the development of such systems is that the datasets used in the construction of classifiers are typically imbalanced. This is because the classification categories do not have relatively equal representation in the datasets. This study investigates a two-stage classifier approach to NIDS based on imbalanced intrusion detection datasets by separating the training and detection of minority and majority intrusion classes. The purpose of this is to allow flexibility in the classification process, for example, two different classifiers can be used for detecting minority and majority classes respectively. In this paper, we performed experiments using the random forests classifier and the contemporary UNSW-NB15 dataset was used to evaluate the effectiveness of the proposed approach.

Disciplines

Engineering | Science and Technology Studies

Publication Details

Zong, W., Chow, Y. & Susilo, W. (2018). A two-stage classifier approach for network intrusion detection. Lecture Notes in Computer Science, 11125 329-340. Tokyo, Japan Information Security Practice and Experience: 14th International Conference, ISPEC 2018

A Two-Stage Classifier Approach for Network Intrusion Detection

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. Network Intrusion Detection Systems (NIDS) are essential to combat security threats in network environments. These systems monitor and detect malicious behavior to provide automated methods of identifying and dealing with attacks or security breaches in a network. Machine learning is a promising approach in the development of effective NIDS. One of the problems faced in the development of such systems is that the datasets used in the construction of classifiers are typically imbalanced. This is because the classification categories do not have relatively equal representation in the datasets. This study investigates a two-stage classifier approach to NIDS based on imbalanced intrusion detection datasets by separating the training and detection of minority and majority intrusion classes. The purpose of this is to allow flexibility in the classification process, for example, two different classifiers can be used for detecting minority and majority classes respectively. In this paper, we performed experiments using the random forests classifier and the contemporary UNSW-NB15 dataset was used to evaluate the effectiveness of the proposed approach.

Keywords: Machine learning; Network intrusion detection; Random forests

1 Introduction

For many people, the Internet has become a ubiquitous part of daily life and numerous online services and applications are used everyday. At the same time, the threat of cyber attacks is increasing and cyber security experts have undertaken extensive studies on methods of combating such security threats. Network Intrusion Detection Systems (NIDS) are potential automated solutions for protecting online environments [17]. While the most effective method for the development of NIDS remains a challenging and open question, machine learning is seen as a very promising approach as these techniques can perform real-time automated detection of potential threats [2, 17].

Misuse detection and anomaly detection are two major approaches adopted in NIDS. Misuse detection focuses on identifying the signatures or patterns of malicious records. When a new record is received, a misuse detection system

compares it with existing signatures to classify it as normal or malicious activity. One of the major problems of misuse detection is that it performs poorly against novel attacks, since the system cannot match it with signatures that have previously been classified as malicious activity [7, 17]. On the other hand, anomaly detection attempts to identify behaviors that differ significantly from regular network activity. Thus, accurate behavior profiles of normal behavior are important in such systems [7]. While anomaly detection systems outperform misuse detection systems when it comes to detecting novel attacks, they typically produce high false alarm rates, which is undesirable and researchers often attempt to reduce the number of false alarms [6].

Another problem faced in the development of NIDS based on machine learning, is that the datasets used in the construction of classifiers are typically extremely imbalanced. A dataset in which the classification categories are not approximately equally represented is considered to be imbalanced [3, 4]. The characteristic representation of malicious activity in datasets that are used for intrusion detection is usually extremely imbalanced, as certain attacks occur more often than others. The problem that this creates is that some machine learning intrusion detection approaches may perform well at the task of detecting frequent attacks, but are much less effective when it comes to the detection of infrequent attacks, due to the lack of sufficient training data for infrequent attacks [12].

This paper investigates the use of a two-stage approach to the development of NIDS based on imbalanced intrusion detection datasets. The underlying notion behind this approach is to filter the dataset into majority and minority malicious activity classes, and to apply classification algorithm on them separately to produce different models for detection. The purpose of this is to improve the overall detection rate of minority classes and to reduce the error rate. The proposed approach is flexible in that a different classifier can be used for each stage of the NIDS. This study examines the two-stage approach using the random forests classifier and also evaluates the effectiveness of the proposed approach on the contemporary UNSW-NB15 dataset.

Our Contributions. In this study we examine an innovative two-stage classifier approach to NIDS. The main purpose of this approach is to be able to handle imbalanced intrusion detection datasets, by separating the intrusion detection data into majority and minority classes, and training two separate classifiers for each category respectively. In this manner, different classifiers can be used to detect the majority and minority classes, with the overall aim of improving the detection rate, especially of minority classes and to reduce the error rate. While this paper examines the two-stage classifier approach using the random forests classifier, note that different classifiers can be used for each of the two-stages.

2 Background

This section introduces related work on machine learning and the various techniques for dealing with imbalanced intrusion detection datasets. In addition, it also provides a background description of different datasets that are typically used for the development of NIDS.

2.1 Related Work

Over the years, researchers have proposed a variety of different machine learning approaches for intrusion detection, including artificial neural networks, Bayesian networks, support vector machines, etc. [2]. The random forests classifier is an approach that combines decision trees and ensemble learning into an ensemble classifier that consists of multiple decision trees, where each tree grows to the largest possible extent without pruning [1]. The advantages of using random forests include its resistance to over-fitting, and its low number of control and model parameters [2].

Zhang et al. [22] proposed a random forests based NIDS for both misuse detection and anomaly detection. For misuse detection, their approach applies sampling techniques and feature selection algorithms to improve the overall detection performance. Conversely for anomaly detection, an unsupervised outlier detection approach was adopted by first building patterns of network services, then using this to determine anomalies in network traffic.

Intrusion detection datasets are typically imbalanced, as some attacks occur at higher frequencies compared with others. The random forests algorithm attempts to minimize the overall classification errors by lowering the error rate on majority classes while increasing the error rate on minority classes [1, 22]. Therefore, imbalanced datasets will adversely affect the overall performance of accurately classifying minority classes. One of the approaches for dealing with imbalance datasets and to improve the detection rate of minority intrusions, is to over-sample minority intrusions or to down-sample majority intrusions, or to implement both methods [4].

Chawla et al. [3] proposed a method for over-sampling the minority classes by creating synthetic minority classes to achieve better classifier performance in imbalanced datasets. They named this method the Synthetic Minority Oversampling Technique (SMOTE) and showed its improved performance when used in conjunction with down-sampled majority classes using C4.5, Ripper and a Naïve Bayes classifier. The SMOTE method has also been used in other work on machine learning classification models for intrusion detection [12, 13, 19].

Feature selection is an important step in building NIDS, as only certain features may be essential to distinguish intrusions from normal activity. Unessential features may increase the computation cost and error rate [22]. While in many NIDS methods the features are designed by security experts, it would be ideal to have an automated approach to selecting important features. Information gain can be used as a criterion for feature selection, where features with low infor-

mation gain can be eliminated because they have relatively small relevance on classification [15].

The following is a formal definition of information gain [15]:

Definition 1. *Let* X *and* Y *be discrete variables representing sample attributes* $(x_1, x_2, ..., x_m)$ *and class attributes* $(y_1, y_2, ..., y_n)$*, respectively. Then, the information gain,* IG*, of a given attribute* X *regarding a class attribute* Y *is calculated as:*

$IG(Y, X) = Entrophy(Y) - Entrophy(Y|X)$

where

- $Entrophy(Y) = -\sum_{i=1}^{n} P(Y = y_i) log_2 P(Y = y_i)$*, where* $P(Y = y_i)$ *is the probability that* y_i *occurs, and*
- $Entrophy(Y|X) = -\sum_{i=1}^{m} P(X = x_j) Entrophy(Y|X = x_j)$.

2.2 Network Intrusion Detection Datasets

Network intrusion detection datasets are vital for evaluating the effectiveness of NIDS. It has been contended that the commonly used KDD98, KDD_CUP99 and NSL_KDD benchmark datasets for intrusion detection were generated more than a decade ago, and many studies have highlighted flaws in these datasets [8, 18]. Furthermore, it has been argued that these datasets no longer reflect the current network threat environment. The UNSW-NB15 dataset was created as a hybrid of real modern normal and contemporary synthesized attack activities of network traffic [10]. As such, this modern dataset was used in this study for evaluating the effectiveness of the proposed approach.

Table 1 shows the characteristics of a part of the UNSW-NB15 data set, where the training and testing sets have been divided into an approximately 60% to 40% ratio. There were no redundant records among the training and testing set [11]. It can clearly be seen that the different categories are unequally represented in the dataset. For example, the analysis, backdoor, shellcode and worms categories are minority classes that collectively only make up $< 3\%$ of the sets. This imbalance creates problems for classifiers and results in poor detection performance of these minority classes.

3 Proposed Approach

The method proposed in this study adopts a two-stage classification approach for majority and minority classes. From Table 1, it can be seen that majority classes like normal, exploits, generic, etc. occur frequently and there is an abundance of such training samples. On the other hand, minority classes like analysis, backdoor, shellcode and worms only make up less than 3% of the overall dataset. This imbalance typically adversely affects classifier performance, and the purpose of the proposed approach is to increase the performance of minority class detection.

Table 1. Categories and their distribution in part of the UNSW-NB15 dataset [11].

Category	**Training Set**	**Testing Set**
Normal	56,000	37,000
Analysis	2,000	677
Backdoor	1,746	583
DoS	12,264	4,089
Exploits	33,393	11,132
Fuzzers	18,184	6,062
Generic	40,000	18,871
Reconnaissance	10,491	3,496
Shellcode	1,133	378
Worms	130	44
Total Records	175,341	82,332

An overview of dividing the dataset into majority and minority classes for the two stages is depicted in Fig. 1. During the first stage, majority classes, which occupy a major proportion of a training set, are classified as "others" and a model is trained to identify the minority classes using a classifier, the random forests approach was used in this study. In the second stage, the minority classes are removed and another model is trained to identify the majority classes. This results in two different intrusion detection models for identifying the minority and majority classes respectively. While this study uses the random forests classifier for both stages, other classifiers can also be used. In fact, it is possible to use different classifiers for each of the stages.

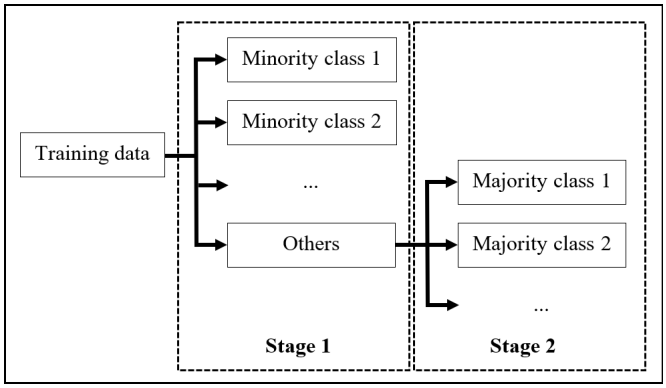

Fig. 1. Overview of the two-stage classification approach.

Fig. 2 shows a more detailed depiction of the processes involved in the proposed approach. The processes are divided into a training phase and a detection phase. It can be seen that the training phases is divided into two stages for the majority and minority classes respectively. Stage 1 involves the training of all the minority classes that are extracted from the full training set, while the majority classes are grouped together into another category for training in the second stage.

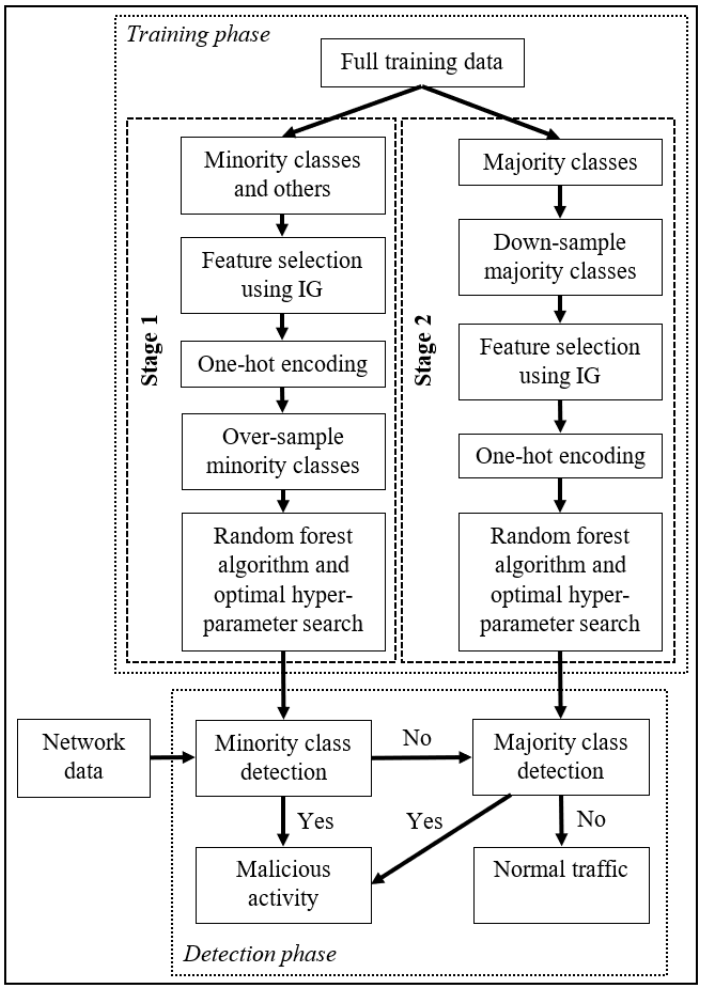

Fig. 2. Processes in the proposed approach.

3.1 Training Phase

In stage 1, after extracting the minority classes, feature selection is performed using the information gain method that was previously defined in Definition 1, and all categorical features are then converted into binary features using one-hot encoding to produce a set of numeric values. The SMOTE method is then used to over-sample the minority classes. The purpose of over-sampling the minority classes is to alleviate the imbalance in the minority classes. From Table 1, it can be seen that even though classes like analysis, backdoor, shellcode and worms are grouped into minority classes, samples for the worms category are extremely under represented. Hence, over-sampling attempts to bring this closer to the other categories.

The resulting set is then used for the training, in which optimal hyper-parameters are found for the random forests algorithm. Three hyper-parameters are considered for fine tuning the model, namely, the maximum depth of a tree in the forest, the number of trees and the number of features considered when looking for the best split. During the training phase, the random forests algorithm calculates the out-of-bag (oob) error. Since the oob error rate can be taken as an indication of whether the model is well trained, a random search is performed to find the lowest oob error rate, and the corresponding hyper-parameters are obtained from this.

Stage 2 undergoes a relatively similar process to obtain a trained model for identifying the majority classes. Only the majority classes are used in the training set, the minority classes are removed, since this was handled in stage 1. Down-sampling is performed to balance the majority classes using a random selection method. This is done for the same reason as over-sampling the minority classes. The distribution of network traffic within the majority classes in itself is unbalances, hence, down-sampling is performed to balance certain categories. Information gain is again used for feature selection, followed by one-hot encoding. This is subsequently used for training, and the optimal hyper-parameters search is performed for stage 2 random forest optimization.

It should be noted that while the random forests approach was used for both stages in this study, the proposed approach is flexible in that other classifiers can also be used for each stage respectively. For example, other classifiers like decision tree approach, logistic regression, artificial neural network, etc. can also be used and may potentially result in better performance.

3.2 Detection Phase

During the next phase of the proposed approach, which is the detection phase, network data is input into the system. When used for intrusion detection, the model for identifying minority classes is used first to determine whether an activity is malicious. If it is not identified as one of the minority classes, the second model is then applied to identify whether the activity is a majority intrusion. Otherwise, it is determined to be normal network traffic.

4 Results and Discussion

To evaluate the effectiveness of the proposed approach, an experiment was performed on the UNSW-NB15 dataset. The UNSW-NB15 training dataset was used to train the two intrusion detection models, and the full testing dataset was used to evaluate the performance of the proposed approach.

Table 2 shows results of the minimum oob error (MoE) rates and their corresponding hyper-parameters for the respective stages. In the table, the hyper-parameters are the maximum depth (MD), which refers to the maximum depth of a tree in the forest, the number of trees (NoT), and the number of features (NoF) for best split after one-hot encoding. The total number of features (TNoF) refers to the number of features remaining after one-hot encoding and feature selection.

Table 2. Minimum oob error rates and the corresponding hyper-parameters.

	MoE	**MD**	**NoT**	**NoF**	**TNoF**
Stage 1	0.119	29	179	109	138
Stage 2	0.167	23	248	80	170

A comparison of the proposed approach with the five different techniques (i.e. Decision Tree (DT), Logistic Regression (LR), Naïve Bayes (NB), Artificial Neural Network (ANN) and Expectation-Maximization (EM) clustering) as presented in Moustafa and Slay [11] is shown in Table 3. From the table, it can be seen that the resulting accuracy of the proposed approach is higher than the other techniques, while the False Alarm Rate (FAR) is lower. This suggests that the overall performance of the proposed approach is better than most of the other techniques and comparable with the DT technique. Fig. 3 shows the confusion matrix depicting the performance results of the proposed approach for the individual categories.

Table 3. Comparison with the different techniques from [11].

Technique	**Accuracy (%)**	**FAR (%)**
DT [20]	85.56	15.78
LR [21]	83.15	18.48
NB [16]	82.07	18.56
ANN [21]	81.34	21.13
EM clustering [14]	78.47	23.79
Proposed Approach	85.78	15.64

		Predicted										
		Normal	Fuzzers	Reconnaissance	DoS	Exploits	Generic	Analysis	Backdoor	Shellcode	Worms	Recall (%)
Actual	Normal	26024	8602	10	144	344	3	1271	23	565	14	70.3
	Fuzzers	541	3680	12	1026	299	1	171	139	183	10	60.7
	Reconnaissance	11	21	2927	267	154	2	14	57	42	1	83.7
	DoS	36	72	36	2834	540	5	159	348	56	3	69.3
	Exploits	119	219	408	2743	6756	10	306	464	80	27	60.7
	Generic	10	61	2	149	399	18205	1	15	25	4	96.5
	Analysis	13	1	0	462	13	0	118	70	0	0	17.4
	Backdoor	0	4	1	384	13	0	88	93	0	0	16.0
	Shellcode	3	44	6	25	34	0	0	4	262	0	69.3
	Worms	0	0	0	0	3	1	0	0	0	40	90.9
	Precision (%)	97.3	29.0	86.0	35.3	79.0	99.9	5.5	7.7	21.6	40.4	

Fig. 3. Confusion matrix.

While the attacks represented in the minority classes are typically infrequent, they are nevertheless potentially dangerous. However, most of these attacks (i.e. analysis, shellcode and worms) could not be detected using the NB and EM clustering approaches as reported in Moustafa and Slay [9]. Only backdoor attacks could be detected by the NB approach with a low accuracy of 20%.

In other recent work, these attacks could be detected at low rates using a random forests with stratified cross-validation method [5]. Fig. 4 and 5 provide a comparison of the recall and precision performances, respectively, between the results reported in Janarthanan and Zargari [5] and the proposed approach. It can be seen from Fig. 4 that the recall results of the proposed two-stage approach performs better in comparison. However, the precision performance in Fig. 5 is lower. Nevertheless, for minority classes a higher recall rate is more important than precision, because these attacks are potentially more dangerous than other attacks. Therefore, higher recall values prevent these attacks from escaping detection.

Upon closer inspection of the overall results, it was found that most of the misclassification was to do with fuzzing activity. Table 4 provides the rate of normal traffic that was misclassified as malicious activity. It can be seen from the table at a large portion of the misclassification are for fuzzers. Fuzzers are attacks where the attacker attempts to discover security loopholes in a pro-

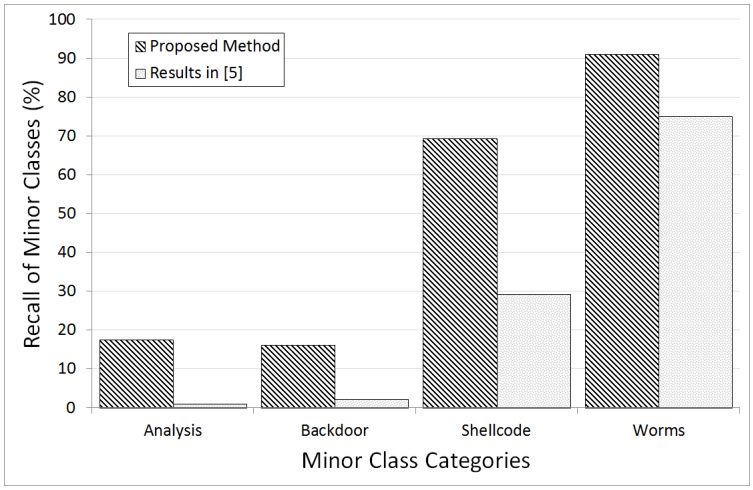

Fig. 4. Comparison of minority classes recall performance.

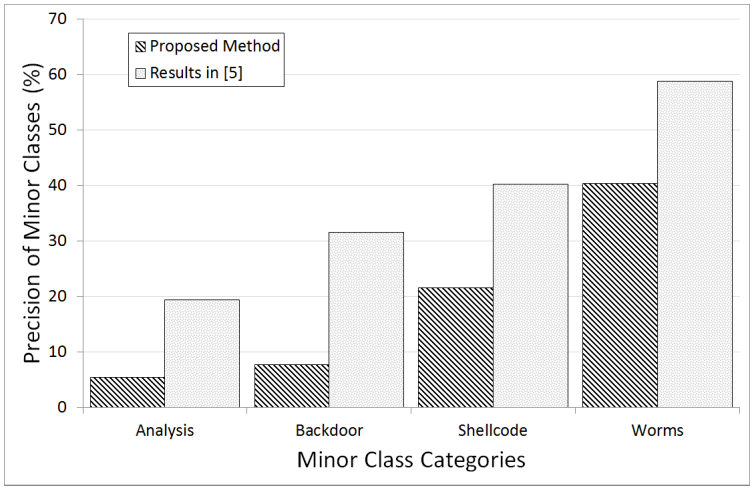

Fig. 5. Comparison of minority classes precision performance.

gram, operating system or network [11]. They are not necessarily dangerous in themselves when compared with other attacks. Fuzzing activity has to do with inputting lots of random data. As such, they do not have a fixed pattern and are more difficult to distinguish from normal network traffic. Nevertheless, as future work it would be ideal to be able to reduce the misclassification rate of this category of activity.

Table 4. Normal activity misclassified as malicious.

Categories	**Misclassification (%)**
Analysis	3.4
Backdoor	0.1
DoS	0.4
Exploits	0.9
Fuzzers	23.2
Generic	0.0
Reconnaissance	0.0
Shellcode	1.5
Worms	0.0

It should be noted that even though this study uses the random forests approach for both stages of the proposed approach, each stage can potentially use a different classification technique. For example, for the minority classifier, other techniques like a decision tree approach, logistic regression, or artificial neural network can be used to potentially increase the detection precision. As such, this will the subject of future work.

5 Conclusion

This study has demonstrated a two-stage classifier approach to NIDS based on imbalanced intrusion detection datasets. The purpose is to address the problem faced in the development of NIDS, which is that the datasets used in the construction of classifiers are typically imbalanced. The primary notion is to separate the training and detection of minority and majority intrusion classes to improve the overall detection rate of minority classes and to reduce the error rate. The effectiveness of the proposed approach was evaluated using the contemporary UNSW-NB15 dataset and was shown to produce favorable results when compared with other approaches. Future work will focus on examining the proposed approach with other classifiers in the two-stages.

References

1. L. Breiman. Random forests. *Machine Learning*, 45(1):5–32, 2001.

2. A. L. Buczak and E. Guven. A survey of data mining and machine learning methods for cyber security intrusion detection. *IEEE Communications Surveys and Tutorials*, 18(2):1153–1176, 2016.
3. 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.
4. C. Chen, A. Liaw, and L. Breiman. Using random forest to learn imbalanced data. Technical report, University of California, Berkeley, 2004.
5. T. Janarthanan and S. Zargari. Feature selection in UNSW-NB15 and KDD-CUP'99 datasets. In *2017 IEEE 26th International Symposium on Industrial Electronics (ISIE)*, pages 1881–1886, June 2017.
6. S. Ji, B. Jeong, S. Choi, and D. H. Jeong. A multi-level intrusion detection method for abnormal network behaviors. *J. Network and Computer Applications*, 62:9–17, 2016.
7. J. Kevric, S. Jukic, and A. Subasi. An effective combining classifier approach using tree algorithms for network intrusion detection. *Neural Computing and Applications*, 28(S-1):1051–1058, 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. N. Moustafa and J. Slay. The significant features of the UNSW-NB15 and the KDD99 data sets for network intrusion detection systems. In *2015 4th International Workshop on Building Analysis Datasets and Gathering Experience Returns for Security (BADGERS)*, volume 00, pages 25–31, Nov. 2015.
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 and J. Slay. The evaluation of network anomaly detection systems: Statistical analysis of the UNSW-NB15 data set and the comparison with the KDD99 data set. *Information Security Journal: A Global Perspective*, 25(1-3):18–31, 2016.
12. H. H. Pajouh, G. Dastghaibyfard, and S. Hashemi. Two-tier network anomaly detection model: a machine learning approach. *J. Intell. Inf. Syst.*, 48(1):61–74, 2017.
13. M. R. Parsaei, S. M. Rostami, and R. Javidan. A hybrid data mining approach for intrusion detection on imbalanced nsl-kdd dataset. *International Journal of Advanced Computer Science and Applications*, 7(6):20–25, 2016.
14. M. Salem and U. Buehler. Mining techniques in network security to enhance intrusion detection systems. *International Journal of Network Security & Its Applications*, 4(6), 2012.
15. P. Sangkatsanee, N. Wattanapongsakorn, and C. Charnsripinyo. Practical real-time intrusion detection using machine learning approaches. *Computer Communications*, 34(18):2227–2235, 2011.
16. M. Shyu, K. Sarinnapakorn, I. Kuruppu-Appuhamilage, S. Chen, L. Chang, and T. Goldring. Handling nominal features in anomaly intrusion detection problems. In *15th International Workshop on Research Issues in Data Engineering (RIDE-SDMA 2005), Stream Data Mining and Applications, 3-7 April 2005, Tokyo, Japan*, pages 55–62. IEEE Computer Society, 2005.
17. 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.
18. M. Tavallaee, E. Bagheri, W. Lu, and A. A. Ghorbani. A detailed analysis of the KDD CUP 99 data set. In *2009 IEEE Symposium on Computational Intelligence for Security and Defense Applications, CISDA 2009, Ottawa, Canada, July 8-10, 2009*, pages 1–6. IEEE, 2009.
19. A. Tesfahun and D. L. Bhaskari. Intrusion detection using random forests classifier with smote and feature reduction. In *2013 International Conference on Cloud Ubiquitous Computing Emerging Technologies*, pages 127–132, Nov 2013.
20. The Bro Project. *The Bro Network Security Monitor*, 2014. https://www.bro.org/.
21. I. H. Witten, E. Frank, and M. A. Hall. *Data Mining: Practical Machine Learning Tools and Techniques*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 3rd edition, 2011.
22. J. Zhang, M. Zulkernine, and A. Haque. Random-forests-based network intrusion detection systems. *IEEE Trans. Systems, Man, and Cybernetics, Part C*, 38(5):649–659, 2008.