Majority Voting Ensemble Learning for Intrusion Detection using Recursive Feature Elimination

Kudakwashe Zvarevashe, Prudence Kadebu, Addlight Mukwazvure and Fungai Mukora
Faculty of Computer Engineering, Informatics and Communication
University of Zimbabwe
Harare, MI 48075, Zimbabwe
zvarevashek@schooloftechnology.ac.uz, kadebup@schooloftechnology.ac.uz,
mukwazvurea@schooloftechnology.ac.uz, mukoraf@schooloftechnology.ac.uz

Tatenda Trust Gotora
Computer Science Department
Midlands State University
Gweru, Zimbabwe
tatenda.gotora88@gmail.com

Abstract

Intrusion detection has become a popular solution of mitigating cyber-attacks in this technological era. Deep learning algorithm have become the go to solution in developing intelligent intrusion detection models and they have been successful. However, deep learning algorithms are too slow in training and this may not be effective for real time applications. Therefore, we propose to develop a faster intrusion detection model using a novel majority voting ensemble with Random Forest recursive feature elimination to improve computation time as well as the detection rate. By including the random forest recursive feature elimination in the proposed model, redundant features will be removed while maintaining the most discriminative features. We evaluated the efficacy of the proposed model on the CICIDS 2017 dataset. In addition, we compared our model with deep learning algorithms which include Resnet50, LSTM and AE-FCN. The experimental results showed that our proposed model outperformed the other deep learning algorithms in terms of accuracy, precision, recall and computation time.

Keywords

Intrusion detection, cyber-attacks, Machine learning , Deep learning and Ensemble.

1. Introduction

Communication has become extremely dynamic because of the ever-evolving computing resources (Vinayakumar et al. 2019). This has been recently spear-headed by the Covid 19 pandemic which has forced a lot of businesses to allow their employees to work from the comfort of their homes. This has presented hackers with opportunities to hijack and hack the computing systems of most people. Over the years, intrusion detection systems have been developed to protect systems from such malicious and nefarious activities. However, while these systems were developed and improved over time, hackers have not been taking vacations. They have also been developing new ways to exploit computer systems. Intrusion detection systems are intelligent software designed to handle cyber-attacks on the network (Rajagopal, Kundapur, and Hareesha 2020). In addition, these systems are responsible for inspecting packets of data entering into a system connected to the internet as either abnormal or normal. In this case, abnormal packets or behaviour are described as misuse detection or anomaly detection (He et al. 2019). Misuse detection involves the situation whereby the signature of a malicious transaction determines the attack model whereas in anomaly detection there is no relationship between the expected behaviour of network flow and the data involved (Thapa et al. 2020).

Machine learning has become a preferred option for developing intrusion detection systems because it provides the required intelligence of classifying unknown data or behaviours as either normal or abnormal (Manimurugan et al.

2020). Most researchers have proposed the use of various machine learning (ML) algorithms such as Decision Trees (DT), Naive Bayes (NB) and Support Vector Machines (SVM) for anomaly detection to adapt to new forms of cyber-attacks (Arslan, Gunduz, and Sagiroglu 2016). However, this has not been sufficient. Most ML algorithms proposed over the years have yielded high rates of false positives which results in the development of flawed intrusion detection systems (KASIM 2020). Another problem in developing intrusion detection systems is the high number of features found in the datasets used. This is computationally expensive in terms of processing time and results in intrusion detection systems that are not ideal for real time computing environments. Therefore, some researchers have proposed the use of transfer learning techniques to extract effective features (Musafer et al. 2020). This has yielded promising results at the expense of processing time. Therefore, this paper presents a method that achieves high detection rates without compromising the computation time. In summary, the key contributions of this body of work are as follows:

- Discarding redundant features using random forest feature elimination (RF-RFE) improves the computation time and detection rate of intrusion detection models.
- Majority voting ensembles are effective at detecting several types of malicious behaviours.
- Resnet50, LSTM and AE-FCN are efficient in detecting Distributed denial of service attacks

The remainder of this paper is succinctly organized as follows. Section 2 discusses the related studies in chronological order. Section 3 describes the experimental database and the study methods. The details of the results and concluding statements are given in Sections 4 and 5 respectively.

2. Related work

Most researchers have proposed various techniques in an effort to mitigate the problem of low detection rates in the development of intrusion detection systems. Various IDS datasets have been developed over the past decades and these have been used to benchmark the efficacy of the proposed solutions. Lee at al (Lee and Park 2019) proposed the use of AE-CGAN (autoencoder-conditional GAN (generative adversarial networks)) model and Random forest algorithm. They evaluated their technique on the Canadian institute for cybersecurity intrusion detection system (CICIDS) 2017 dataset achieving accuracy score of 98.45%.

Instead of relying on long feature vectors, He et al (He et al. 2019) took a step forward in extracting different level features from a network connection. They further proposed a Multimodal-Sequential Approach wherein they used a combination of multimodal deep auto encoder (MDAE) and LSTM technologies. Their method achieved an average classification of 94% for binary classification and 88% for multi-class classification using the CICIDS2017 dataset. In the same vein, the Information gain (IG) feature selection algorithm was used to select the most influential features in CICIDS2017 and these were fed to the rule-based classifiers (Kshirsagar and Shaikh 2019). Their experimental results showed an average accuracy of 99.99% using the JRip rule-based classifier.

Musafer et al (Musafer et al. 2020)posited that the use of an enhanced design of sparse autoencoder for latent features extraction based on trigonometric simplexes improves the accuracy of detecting intrusions in intrusion detection systems. They evaluated the efficacy of their proposed system on the CICIDS2017 dataset achieving an accuracy score of 99.63%. Abdulhammed et al (Abdulhammed et al. 2019) conducted a research to investigate the most efficient dimensionality reduction techniques in developing intrusion detection systems. Furthermore, they also developed a Uniform Distribution Based Balancing (UDBB) to cater for imbalanced classes. Their experimental results achieved an overall accuracy of 98.8% on the CICIDS2017 dataset using the Random Forest ensemble algorithm.

In their quest to develop an intelligent intrusion detection system for a cloud environment, Chiba et al (2019) proposed the use of a hybrid optimization framework (IGASAA) based on Improved Genetic Algorithm (IGA) and Simulated Annealing Algorithm (SAA). To benchmark the efficacy of their proposed method, they used three intrusion detection system datasets namely, CICIDS2017, NSL-KDD version 2015 and CIDDS-001. Their method achieved an overall accuracy of 99.93% on the CICIDS2017 dataset. Vinayakumar et al (2019) developed a highly scalable and hybrid DNNs framework called Scale-Hybrid-IDS-AlertNet (SHIA)

Kasim (2020) proposed a combination of deep learning Auto-encoders and Support vector machines (SVM) to improve the detection of anormalies within a network. He evaluated their proposed method on the CICIDS dataset achieving 99.41% overall accuracy. To improve the accuracy and efficiency of flow-based network traffic attack detection, Chen et al (Chen et al. 2020) developed a Fuzzy Entropy Weighted Natural Nearest Neighbour (FEW-NNN) method. They used KDD99 and CICIDS-2017 datasets to evaluate the effectiveness of their novel method. Their experimental results showed that their method is efficient in detecting anomalies.

Furthermore, Injadat et al (2020) eloquently presented a multi-stage optimized Machine Learning-based network intrusion detection system framework. Their main aim was to develop an intrusion detection system that is computationally inexpensive yet improving the detection rate. The CICIDS 2017 and the UNSW-NB 2015 datasets were used to benchmark the performance of the proposed method. The experimental results showed an average accuracy of 99%.

Gu et al (2019) presented a semi-supervised weighted k-means method to improve the process of detecting intrusions in intrusion detection systems. In addition, they developed the Hadoop-based feature selection algorithm to select the most discriminative features. To evaluate the efficacy of their proposed method, the used four intrusion detection datasets namely DARPA DDoS dataset, CAIDA ‘‘DDoS attack 2007’’ dataset, CICIDS ‘‘DDoS attack 2017’’ dataset and real-world dataset. Their experimental results achieved an overall detection rate of 98.86%. Manimurugan et al (Manimurugan et al. 2020) developed a Deep Belief Network (DBN) model to detect intrusions in IoT (Internet of things) environments. The proposed method achieved 99.37% accuracy for detecting the intrusion-free class, 97.93% for Botnet class, 97.71% for Brute Force class, 96.67% for Dos/DDoS class, 96.37% for Infiltration class, 97.71% for Ports can class and 98.37% for Web attack. These experiments were done on the CICIDS2017 dataset. Table 1 shows the result of the comparative analysis of our method with the related methods in terms of the experimental dataset, classification method and maximum percentage accuracy results obtained.

Table 1. A comparison of our method with related methods.

Reference	Dataset	Classification method	Result
Lee & Park, 2019	CICIDS2017	AE-CGAN + Random Forest	98.45% (Accuracy)
He et al., 2019	CICIDS2017	MDAE + LSTM	88% (Accuracy)
Kshirsagar & Shaikh, 2019	CICIDS2017	Information gain + JRip rule-based classifier	99.99% (Accuracy)
Musafer, Abuzneid, Faezipour, & Mahmood, 2020	CICIDS2017	enhanced design of sparse autoencoder for latent features extraction based on trigonometric simplexes	99.63% (Accuracy)
Abdulhammed, Musafer, Alessa, Faezipour, & Abuzneid, 2019	CICIDS2017	Uniform Distribution Based Balancing (UDBB) + Random Forest	98.8% (Accuracy)
Chiba, Abghour, Moussaid, El omri, & Rida, 2019	CICIDS2017	hybrid optimization framework (IGASAA) based on Improved Genetic Algorithm (IGA) and Simulated Annealing Algorithm (SAA)	99.93% (Accuracy)
Kasim, 2020	CICIDS2017	Auto-encoders + SVM	99.41% (Accuracy)
Injadat, Moubayed, Nassif, & Shami, 2020	CICIDS2017	optimized RF classifier with Bayesian Optimization using Tree Parzen Estimator (BO-TPE-RF)	99% (Accuracy)
Gu, Li, Guo, & Wang, 2019	CICIDS2017	k-means Hadoop-based feature selection algorithm	98.86% (Accuracy)
Manimurugan et al., 2020	CICIDS2017	Deep Belief Network	99.37% (intrusion-free class) 97.93% (Botnet), 97.71% (Brute

			Force), 96.67% (Dos/DDoS), 96.37% (Infiltration), 97.71% (Ports can) 98.37% (Web attack)
Proposed Model	CICIDS2017	MVC + RF-RFE	99.96%

3. Materials and Methods

In this section, we present the material used in this experimental study, which is the CICIDS2017 intrusion detection dataset. The methodology of this body of work follows two phases, which are feature selection and classification. The primary purpose of the feature selection phase was to select the most discriminative feature while discarding redundant features. According to research posited by various authors, feature selection significantly improves the computation time of machine learning algorithms. This phase was subsequently followed by the classification stage where the selected features were fed into a majority voting ensemble algorithm. The performance of the proposed majority voting ensemble algorithm was evaluated against other learning algorithms, which are Majority Voting Classifier (MVC), Resnet50, Long short-term memory (LSTM) and Auto-Encoder Full Convolution network (AE-FCN). The flow diagram of the methodology is illustrated in Figure 1.

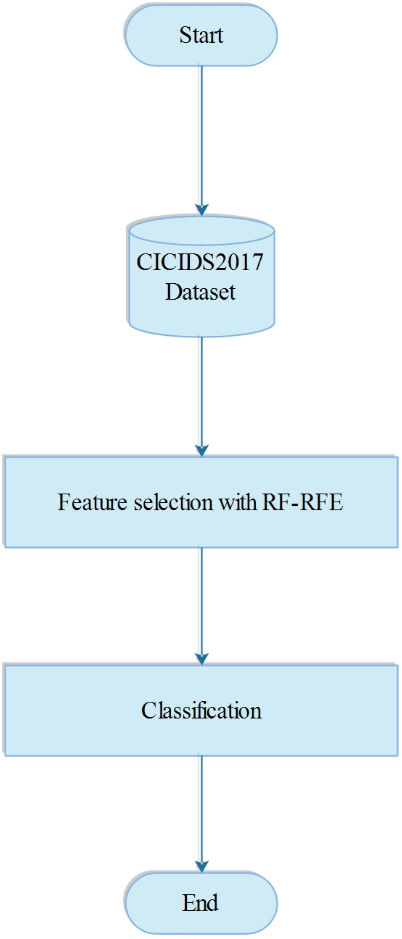


Figure 1. Proposed system flow chart

3.1 Dataset

Unlike some research fields such as speech processing, there are few public research datasets used for conducting experiments to solve intrusion detection problems. The most popular datasets in this field include ISCX2012 dataset, NSL-KDD dataset, Kyoto2006 dataset, KDD99 dataset, and NSL-KDD dataset. Research has regarded theses datasets as inefficient even though they have thrived positively over the years. This is because some of them are devoid of the required diversity required to develop a modern solution and most of them are archaic and unreliable (Lee and Park 2019). CICIDS2017 dataset and UBSW-VB15 dataset are two of the modern intrusion detection datasets that have the required diversity which include current popular attacks. However, UBSW-VB15 has fewer number of attacks compared to CICIDS2017. In addition, CICIDS2017 comprises of realistic background traffic that represents the network events (Abdulhammed et al. 2019). Therefore, it is against this background that we chose to use CICIDS2017 in this experimental study.

The CICIDS2017 dataset was developed using abstract behaviors from traffic collected from 25 users. The dataset consists of different types of attacks malicious and normal traffic, which is also referred to as the benign class. The datasets consists of 2,830,108 entries and brief descriptions of the different kinds of attacks found in CICIDS2017 are shown in Table 2.

Table 2. CICIDS 2017 dataset analysis.

Traffic type	**Number**	**Percentage**
Benign	2,358,036	80.3004%
HeartBleed	11	0.0004%
Web Attack: SQL Injection	21	0.0008%
DoS Hulk	231,073	8.1630%
Port Scan	158,930	5.6441%
DoS Slow Loris	5796	0.2048%
DoS Slow HTTP Test	5,499	0.1943%
Botnet	1966	0.0695%
Web Attack: Brute Force	1507	0.0532%
Web Attack: XSS	625	0.0221%
Infiltration	36	0.0013%
DDoS	41,835	1.478%
DoS GoldenEye	10,293	0.3636%
FTP Patator	7938	0.2804%
SSH Patator	5,897	0.2083%

3.2 Feature selection

Feature selection is an essential process that involves the selection of highly discriminative features to improve classification rates (Kuhn 2012; Kursa and Rudnicki 2010). In this experimental study, feature selection was used to select the features that play a prominent role in distinguishing various attacks. Since CICIDS2017 consists of 83 features, some of them may be redundant and this has a negative impact on the computation time of the models used. Various feature selection techniques have been used in developing intrusion detection systems and these include Information gain, autoencoders and many more. The Random forest recursive feature elimination RF-RFE algorithm has not been used in this regard. It has performed well in gender identification (Zvarevashe and Olugbara 2018) and speech cross language recognition (Zvarevashe and Olugbara 2020) and this inspired us to explore it further in this body of work. Figures 2 and 3 depict the iterative cycle involved in the application of the RF-RFE algorithm.

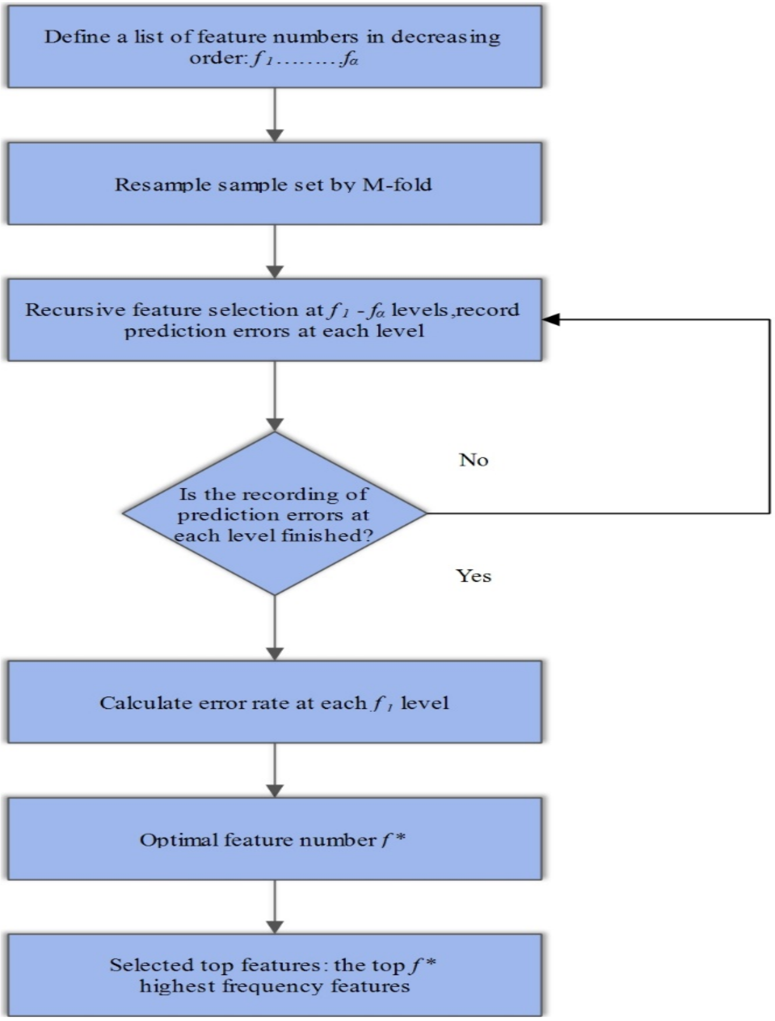
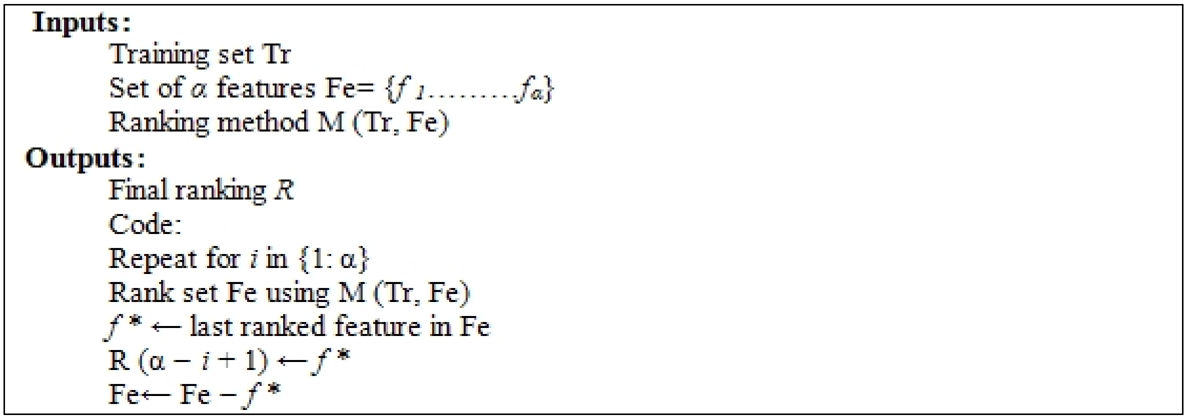

Define a list of feature numbers in decreasing order: f_1........f_α
Resample sample set by M-fold
Recursive feature selection at f_1 - f_α levels,record prediction errors at each level
Is the recording of prediction errors at each level finished?
No
Yes
Calculate error rate at each f_1 level
Optimal feature number f^*
Selected top features: the top f^* highest frequency features

Figure 2. Workflow of Random Forest Recursive Feature Elimination



```
Inputs:
        Training set Tr
        Set of $\alpha$ features Fe= {$f_1$........$f_\alpha$}
        Ranking method M (Tr, Fe)
Outputs:
        Final ranking $R$
        Code:
        Repeat for $i$ in {1: α}
        Rank set Fe using M (Tr, Fe)
        $f^* \leftarrow$ last ranked feature in Fe
        R ($\alpha - i + 1$) $\leftarrow f^*$
        Fe$\leftarrow$ Fe $- f^*$
```

Figure 3. Summarized RF-RFE Algorithm

3.3 Classification

Classification was the final phase of this experimental study and it involved the use of Resnet50, LSTM and AE-FCN in classifying the different attacks found in the CICIDS2017 dataset. We developed an ensemble-learning algorithm using the hard voting technique. Research has shown that ensemble algorithms perform better than inducers in solving classification problems (Bhavan et al. 2019; Geurts, Ernst, and Wehenkel 2006; Kotsiantis and Pintelas 2004) and this is the reason MVC was developed for this research study. MVC was developed using the majority voting based mechanism, which is also referred to as hard voting. Furthermore, the technique is also known as plurality voting. In majority voting, every individual classifier votes for a class, and the majority wins as shown in equation 1. We used four learning algorithms to develop the ensemble classifier and these are linear discriminant analysis, logistic regression, extra trees and decision trees.

$$Y = mode\ \{C_1(x), C_2(x),.., C_n(x)\} \quad \text{Equation 1}$$

where Y is the class label while C stands for each individual classifier applied to a set of features (x). Three performance metrics were used to evaluate the effectiveness of our proposed method and these are accuracy, precision and f1-score.

4. Results and Discussion

All experiments were conducted on an i7 2.3GHz processor with ab 8 GB of random access memory (RAM). As we conducted these experiments, our purpose was motivated by a hypothesis, which states that discarding redundant features has a positive impact on the computation time of intrusion detection system models. Table 3 shows the computation time in ms for each individual classifier. MVC with RF-RFE was the fasted classifier (190 ms) followed by Resnet50 (230 ms). AE-FCN had the longest computation time followed by LSTM. These results imply that even though MVC+RF-RFE had the second best processing time, it is highly promising for developing intrusion detection models

Table 3. Training time (ms) of individual classifiers

Classifier	MVC+RF-RFE	Resnet50	LSTM	AE-FCN
Computation time	190	230	379	1050

The weighted precision, recall and F1 scores are shown in Table 4. The experimental results show that the proposed MVC+RF-RFE is an excellent classifier for developing intrusion detection models. This is because it achieved the highest performance scores across all metrics. A high precision score of 99.13% shows that MVC+RF-RFE is indeed a promising classifier for intrusion detection since the classes were skewed. However, LSTM proved to be equally good since it achieved a precision score of 97.5%. Resnet50 was the lowest performing classifier across all performance metrics. It achieved a precision score of 85.5%.

Table 4. Weighted precision, recall and accuracy

Classifier	Precision	Recall	F1-score	Accuracy
MVC+RF-RFE	99.13	97.2	98.07	99.96
Resnet50	85.5	83.7	88.3	90.03
LSTM	97.5	99.5	98.6	98.73
AE-FCN	95.80	99.20	97.50	97.70

Table 5 shows the accuracy, precision and recall for the individual classes and the key to the symbols are is shown in Table 6. The experimental results show that MVC+RF-RFE outperforms the deep learning classifiers in detecting most of the intrusions such as DoS Hulk, Port Scan, DoS Slow Loris, DoS Slow HTTP Test, Botnet, Web Attack: Brute Force, Web Attack: XSS, DoS GoldenEye and Infiltration since it achieved perfect precision scores (100%). In addition, LSTM detected a couple of attacks with high precision and these include Benign, HeartBleed, Port Scan, DoS Slow Loris, Botnet, Web Attack: Brute Force, Web Attack: XSS, Infiltration and DDoS. Overall, all the classifiers struggled to detect Web Attack: SQL Injection. The results show that the proposed model is comparable to the models proposed by various authors shown in Table 1.

Table 5. Accuracy, precision and recall for each individual class

Classes	**MVC+RF-RFE**			**Resnet50**			**LSTM**			**AE-FCN**		
	A	**P**	**R**	**A**	**P**	**R**	**A**	**P**	**R**	**A**	**P**	**R**
Benign	100%	100%	100%	97.67%	94%	93%	99.26%	100%	94%	99.47%	100%	96%
HeartBleed	100%	100%	82%	100%	100%	100%	100%	100%		100%	100%	100%
Web Attack: SQL Injection	100%	89%	76%	99.22%	77%	100%	99.26%	77%	91%	99.66%	88%	100%
DoS Hulk	99.96%	100%	100%	99.01%	77%	83%	99.52%	90.5%	100%	99.76%	95%	100%
Port Scan	100%	100%	100%	99.99%	70%	67%	99.99%	100%	100%	99.99%	100%	100%
DoS Slow Loris	100%	100%	100%	98.23%	73%	80%	99.53%	100%	100%	98.58%	81.00%	96%
DoS Slow HTTP Test	100%	100%	100%	99.19%	92%	70%	99.96%	99%	99%	99.52%	92.00%	100%
Botnet	100%	100%	100%	98.41%	100%	94%	99.95%	100%		99.52%	100%	94%
Web Attack: Brute Force	100%	100%	100%	98.06%	100%	100%	100%	100%	100%	100%	100%	100%
Web Attack: XSS	100%	100%	100%	97.67%	100%	100%	100%	100%	100%	100%	100%	100%
Infiltration	100%	100%	100%	95.72%	75%	100%	99.99%	100%	100%	98.99%	100%	91%
DDoS	99.96	98%	100%	97.67%	100%	100%	100%	100%	100%	100%	100%	100%
DoS GoldenEye	100%	100%	100%	96.3%	100%	100%	100%	100%	100%	100%	100%	100%
FTP Patator	100%	100%	100%	100%	100%	100%	100%	100%	100%	100%	100%	100%
SSH Patator	100%	100%	100%	99.22%	80%	100%	100%	100%	100%	100%	100%	100%

Table 6. Key to Table 5.

Symbol	**Description**
A	Accuracy
P	Precision
R	Recall

5. Conclusion

Intrusion detection has and still is a popular problem in this digital age. This is because the nature of attacks are always evolving making it difficult to develop intrusion detection systems that detect them with the precision of a surgeon. The proposed model was highly focused on the reduction of the dimensionality of the features to improve both accuracy and computation time. The results showed that the proposed model is an excellent tool for developing intrusion detection models because of its superior performance compared to deep learning classifiers. The proposal also supported our hypothesis, which states that feature selection can improve the computation time since it was the fastest in comparison with the deep learning models. The results of the experimental study also showed that Resnet50, LSTM and AE-FCN are highly effective in detecting DDoS attacks. The main limitation of this body of work was the unbalanced structure of the CICIDS 2017 dataset. Therefore, we would like to create a balanced dataset as part of our future work.

References

Abdulhammed, Razan, Hassan Musafer, Ali Alessa, Miad Faezipour, and Abdelshakour Abuzneid. Features Dimensionality Reduction Approaches for Machine Learning Based Network Intrusion Detection. *Electronics (Switzerland)* 8, vol. 3, pp 1-18, 2019.

Arslan, Bilgehan, Sedef Gunduz, and Seref Sagiroglu. A Review on Mobile Threats and Machine Learning Based Detection Approaches. *Proceedings of the 4th International Symposium on Digital Forensics and Security,* 2016, 7–13.

Bhavan, Anjali, Pankaj Chauhan, Hitkul, and Rajiv Ratn Shah. Bagged Support Vector Machines for Emotion Recognition from Speech. *Knowledge-Based Systems* 184, pp 23-43, 2019.

Chen, Liangchen, Shu Gao, Baoxu Liu, Zhigang Lu, and Zhengwei Jiang. FEW-NNN: A Fuzzy Entropy Weighted Natural Nearest Neighbor Method for Flow-Based Network Traffic Attack Detection. *China Communications* 17, vol. 5, 151–67, 2020.

Chiba, Zouhair, Noreddine Abghour, Khalid Moussaid, A. El omri, and Mohamed Rida. Intelligent Approach to Build a Deep Neural Network Based IDS for Cloud Environment Using Combination of Machine Learning Algorithms. *Computers and Security* vol 86, no 2, pp. 291–317, 2019.

Geurts, Pierre, Damien Ernst, and Louis Wehenkel. Extremely Randomized Trees.*Machine Learning* 63, no. 1, pp. 3–42, 2006.

Gu, Yonghao, Kaiyue Li, Zhenyang Guo, and Yongfei Wang. Semi-Supervised k-Means Ddos Detection Method Using Hybrid Feature Selection Algorithm. *IEEE Access* 7, vol. 87, no 3, pp. 46–65, 2019.

He, Haitao, Xiaobing Sun, Hongdou He, Guyu Zhao, Ligang He, and Jiadong Ren. A Novel Multimodal-Sequential Approach Based on Multi-View Features for Network Intrusion Detection. *IEEE Access* 7, vol. 87, no 4, pp. 1–21, 2019.

Injadat, MohammadNoor, Abdallah Moubayed, Ali Bou Nassif, and Abdallah Shami. Multi-Stage Optimized Machine Learning Framework for Network Intrusion Detection. *IEEE Transactions on Network and Service Management* . vol. 56, no. 4, pp. 1–23, 2020.

Kasim, Ömer. An Efficient and Robust Deep Learning Based Network Anomaly Detection against Distributed Denial of Service Attacks. *Computer Networks,* vol 180, no 3, pp. 45-61, 2020.

Kotsiantis, S B, and P E Pintelas. Combining Bagging and Boosting. *Computational Intelligence*. vol. 1, no. 4 , 324–33, 2004.

Kshirsagar, Deepak, and Jahed Momin Shaikh. Intrusion Detection Using Rule-Based Machine Learning Algorithms. *Proceedings of the 5th International Conference on Computing, Communication Control and Automation, ICCUBEA 2019*, 2019.

Kuhn, Max. Variable Selection Using The Caret Package. *Caret Vignettes*, 2012, 1–24.

Kursa, Miron B., and Witold R. Rudnicki. Feature Selection with the **Boruta** Package. *Journal of Statistical Software,*vol. 36, no. 11, pp. 78-91, 2010.

Lee, Joo Hwa, and Kee Hyun Park. AE-CGAN Model Based High Performance Network Intrusion Detection System. *Applied Sciences (Switzerland)* 9, no. 20, 2019.

Manimurugan, S., Saad Al-Mutairi, Majed Mohammed Aborokbah, Naveen Chilamkurti, Subramaniam Ganesan, and Rizwan Patan. “Effective Attack Detection in Internet of Medical Things Smart Environment Using a Deep Belief Neural Network.” *IEEE Access* 8, vol 81, no. 4 pp. 396–404, 2020.

Musafer, Hassan, Abdelshakour Abuzneid, Miad Faezipour, and Ausif Mahmood. An Enhanced Design of Sparse Autoencoder for Latent Features Extraction Based on Trigonometric Simplexes for Network Intrusion Detection Systems. *Electronics (Switzerland)* vol. 9, no. 2 , pp. 1–12, 2020.

Rajagopal, Smitha, Poornima Panduranga Kundapur, and Katiganere Siddaramappa Hareesha. A Stacking Ensemble for Network Intrusion Detection Using Heterogeneous Datasets. *Security and Communication Networks* 2020, vol. 12, no. 2, pp. 1–9.

Thapa, Niraj, Zhipeng Liu, Dukka B Kc, Balakrishna Gokaraju, and Kaushik Roy. Comparison of Machine Learning and Deep Learning Models for Network Intrusion Detection Systems. *Future Internet*,vol. 16, no. 3, pp. 1–16, 2020.

Vinayakumar, R., Mamoun Alazab, K. P. Soman, Prabaharan Poornachandran, Ameer Al-Nemrat, and Sitalakshmi Venkatraman. “Deep Learning Approach for Intelligent Intrusion Detection System.” *IEEE Access* 7, no. 5 , pp. 31–50, 2019.

Zvarevashe, Kudakwashe, and Oludayo O. Olugbara. Gender Voice Recognition Using Random Forest Recursive Feature Elimination with Gradient Boosting Machines. *Proceedings of the 2018 International Conference on Advances in Big Data, Computing and Data Communication Systems, IcABCD 2018*, pp. 1–6. IEEE, 2018.

Zvarevashe, Kudakwashe, and Oludayo O. Olugbara. Recognition of Cross-Language Acoustic Emotional Valence Using Stacked Ensemble Learning. *Algorithms* 13, no. 10, pp. 1-23, 2020

Biographies

Kudakwashe Zvarevashe is the head of Analytics and Informatics department at the University of Zimbabwe. Kudakwashe holds an MTech in Information Technology from Jawaharlal Nehru Technological University (JNTUH, India) and a BSc in Information Systems from Midlands State University (MSU, Zimbabwe). He has more than 9 years University experience in Teaching, Research and Administration. His research interests are in Cyber Security, Computer Vision, Natural Language Processing, Cloud Computing and Big Data Analytics. He is also a professional member of ACM.

Prudence Kadebu is the head of Computer Engineering department at the University of Zimbabwe. She is a Software Engineer experienced in team leading, Software Engineering, lecturing at various levels, Research and Development, Curriculum Development and Review, Security, Software Quality Assurance, and is an Artificial Intelligence and Machine Learning enthusiast. She has more than 10 years University experience. Prudence is passionate about empowering women and girls in STEM. She is also involved in projects on Solar PV.

Addlight Mukwazvure is a lecturer in the Computer Engineering department at the University of Zimbabwe. She holds an MTech in Computer Science from Jawaharlal Nehru Technological University (JNTUH, India) and a BSc in Computer Science from National university of Science and Technology. Her research interests lie in the domain of Natural language processing, ontologies and machine learning.

Tatenda Trust Gotora is a lecturer in the Computer Science department at Midlands State University (MSU, Zimbabwe). He holds an MTech in Software Engineering from Jawaharlal Nehru Technological University (JNTUH, India) and a BSc in Computer Science from Midlands State University (MSU, Zimbabwe. His research interests lie in the domain of Fog computing, Network engineering and machine learning.

Fungai Nora Mukora is the acting Dean in the Computer Engineering, Informatics and Communication at the University of Zimbabwe. She holds an MSc in Computer Science from University of Zimbabwe and a BSc in Computer Science from University of Zimbabwe. Her research interests lie in the domain of Cyber security, Hardware engineering and machine learning.