Intrusion Detection using Deep Learning Long Short-term Memory with Wrapper Feature Selection Method

Sana Al Azwari[1], Hamza Turabieh[2]
Department of Information Technology, Taif University, Taif, Saudi Arabia

Abstract—**Recently, many companies move to use cloud computing systems to enhance their performance and productivity. Using these cloud computing systems allows the execution of applications, data, and infrastructures on cloud platforms (i.e., online), which increase the number of attacks on such systems. As a resulting, building robust Intrusion detection systems (IDS) is needed. The main goal of IDS is to detect normal and abnormal network traffic. In this paper, we propose a hybrid approach between an Enhanced Binary Genetic Algorithms (EBGA) as a wrapper feature selection (FS) algorithm and Long Short-Term Memory (LSTM). A novel injection method to prevent premature convergence of the GA is proposed in this paper. An intelligent k-means algorithm is employed to examine the solution distribution in the search space. Once 80% of the solutions belong to one cluster, an injection method (i.e., add new solutions) is used to redistribute the solutions over the search space. EBGA will reduce the search space as a preprocessing step, while LSTM works as a binary classification method. UNSW-NB15, a real-world public dataset, is used in this work to evaluate the proposed system. The obtained results show the ability of feature selection method to enhance the overall performance of LSTM.**

Keywords—Intrusion detection; feature selection; long short-term memory; binary genetic algorithm

I. Introduction

With the exponential growth rates of volumes of data, both structured and unstructured, that are generated from a variety of sources, the need to provide protection and privacy becomes a challenging issue for intrusion detection systems (IDSs) in this big data environment. Intrusions are suspicious and unauthorized activities on a computer or network that threaten the security of these systems. IDSs are very crucial to ensure network and information security. These systems can be devices or software that monitor systems or networks for malicious activities or violations of security policies.

Intrusion detection systems detects unusual attacks based on two methods; signature-based detection and anomaly detection. In signature-based detection, IDS analyzes system activities to find patterns that are similar to previously detected and stored patterns in a database. Intrusion detection using an anomaly detection method which relies on machine learning to build models of patterns of normal behavior on the system or the network (i.e., cloud computing systems) to detect patterns of unusual behavior. Fig. 1 presents the main architecture for IDS for cloud computing systems.

There are many algorithms have been proposed to build a robust IDS based on machine learning and soft computing methods. Network traffic data is a high dimensional one, many papers investigated the ability of employing FS algorithms to enhance the overall performance of IDS [1]. For example, Almomani [2] applied four types of FS algorithms, namely, genetic algorithm (GA), particle swarm optimization (PSO), firefly optimization (FFA), and grey wolf optimizer (GWO). Almomani used two classifiers: Support Vector Machine (SVM) and decision tree (J48) to build a robust IDS. Thakkar and Lohiya [3] applied seven ML classifiers (i.e., Neural Networks (NN), Decision Tree (DT), Logistic Regression (LR), Support Vector Machine (SVM), k-nearest neighbours (kNN), Random Forest (RF), and Naïve Bayes (NB)) to build an intelligent IDS. Zhu et al. [4] introduced a multi-objective method for FS for building a robust IDS inside cloud computing systems.

Many contributions in the literature focus on traditional machine learning methods for IDS. However, these methods have high cost in terms of training time when working with big data sets. To overcome this issue, deep learning approach is used for effective learning mechanism in reducing the training time and increasing the accuracy of the obtained results from the IDS. Moreover, the main contribution of this work is to introduce a robust wrapper feature selection that is able to reduce the high dimensionality of the dataset.

This paper is organized as follow: Section II presents the related works of IDS. Section III presents the proposed method used in this paper (i.e., EBGA and LSTM). Section IV presents the data set used in this paper. Section V presents the obtained results and analysis. Finally, Section VI presents the conclusion and future works of this paper.

II. Related Work

The literature shows a number of traditional machine learning approaches methods have been proposed for intrusion detection systems which include Support Vector Machine, K-Nearest Neighbors, Decision Trees, Random Forests, Linear Regression, Naive Bayes, Artificial Neural Networks. Recently, deep learning-based approaches has emerged to overcome the challenges of developing an accurate high-detection rate IDSs. State of the art deep learning approaches that have been used for IDS include Deep Neural Networks (DNNs) [5], Deep Belief Networks (DBNs), Restricted Boltzmann Machines (RBMs), autoencoders and hybrid methods. For example, Zhao et al. [6], proposed an intrusion detection method based on deep belief networks and probabilistic neural network. The KDD CUP 99 data set was used for testing the performance of the proposed method. The result shows that their proposed method performs better than traditional

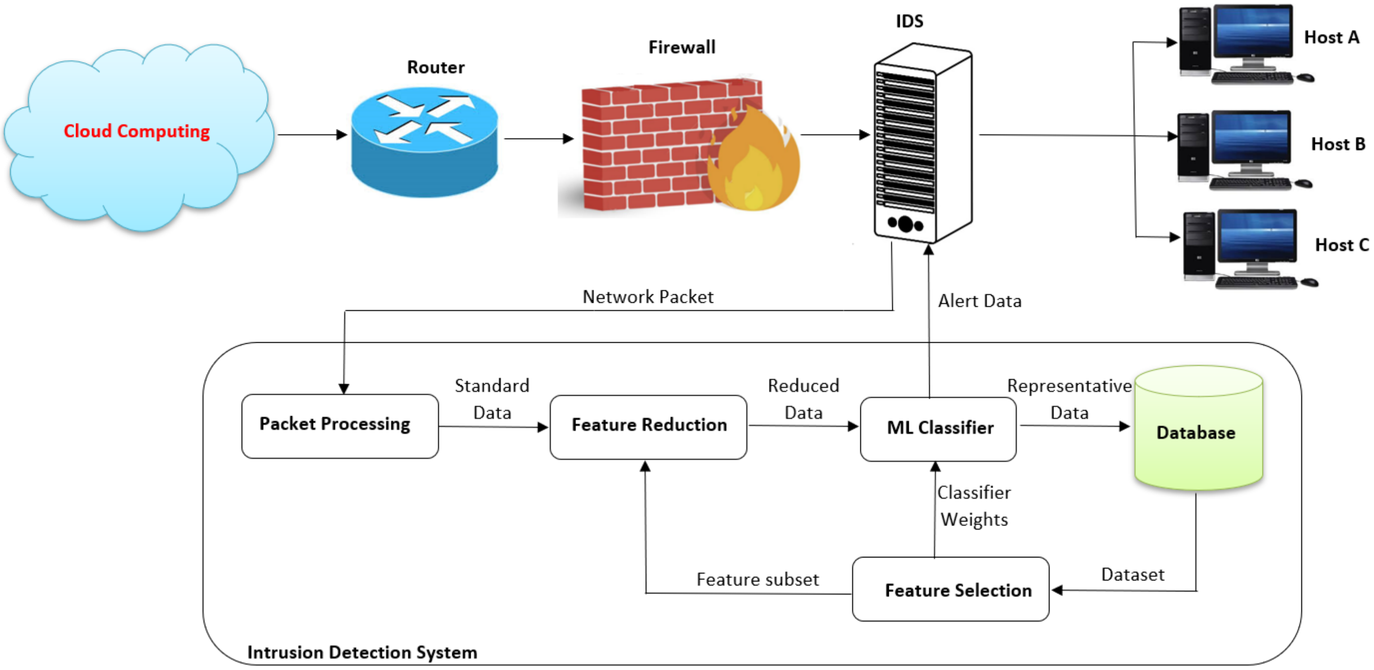

Fig. 1. IDS Architecture in a Cloud Computing.

machine learning techniques with accuracy of 99.1%, precision of 93.25% and FAR of 0.615%.

In [7] Erfani et al. presented a hybrid approach for IDS by combining DBNs with a linear one-class SVM and was applied using several data sets. Their experimental results show that their proposed model is scalable and computationally efficient and when compared to an autoencoder it executes 3 times faster in the training phase and 1000 times faster in the testing phase.

In [8], the authors proposed an approach for IDS based on deep learning using self-taught learning on NSL-KDD, a benchmark data set, with only six features selected out from the forty one features of the data set. results of their experiments and comparisons with other machine learning algorithms; Naive Bayes, SVM and Decision Tree show that using deep learning algorithm is promising as it performs better than the other algorithms with higher accuracy rate and lower false positive rate.

Javaid et al. [5] proposed a network intrusion detection system based on deep learning approach. They used self-taught learning technique (STL) on NSL-KDD benchmark data set. They compared the performance of their approach with the soft-max regression (SMR). their results show that the proposed approach outperforms SMR with accuracy rate more than 98%.

In [9] proposed an approach for network traffic identification using Artificial Neural Networks (ANN) and Stacked AutoEncoder (SAE) based on Deep learning using a real data set of TCP data collected from an internal network. Results of their work show that their proposed approach can classify any flow data to a predefined protocol with accuracy enough to be applied in real applications.

Yin et al. [10] compared the performance of their IDS which is based on recurrent neural network, a deep learning approach, with a number of traditional machine learning techniques. Results from their experiments on NSL-KDD benchmark data set show that the proposed system outperforms traditional machine learning methods in both binary and multi-class classification with high accuracy.

The above work studied the emergence of deep learning in the performance of IDS. However, to date, A few number of existing studies in the literature have addressed the integration of deep learning approaches and Big Data for improving the performance of IDSs. Faker and Dogdu [11] integrated Big Data and deep learning approach to enhance the performance of intrusion detection system using three classifiers to classify attacks in both binary and multi-class classification; Deep Feed-Forward Neural Network (DNN), Random forest and Gradient Boosting Tree (GBT) on UNSW-NB15 and CICIDS2017 data sets. on UNSW-NB15, DNN gives high accuracy results in both binary and multi-class classification of 99.19% and 97.04%, respectively with low prediction times. However, on CICIDS2017, GBT achieved the best accuracy, of 99.99%, in binary classification. Researches in [12] suggested the implementation of Deep Neural Network model (DNN) for IDS to detect and classify unforeseen and unpredictable cyberattacks. They provide a comprehensive evaluation of experiments of DNN and other traditional machine learning models using various benchmark IDS data sets such as KDDCup99, NSL-KDD, UNSW-NB15, Kyoto, WSN-DS and CICIDS2017. Their proposed model exceeded in performance the other classical machine learning classifiers. A recent work by [13] addressed the detection of intrusions through the use of deep learning in big data environment. They proposed a hybrid deep learning model based on convolutional neural network (CNN) and a weight-dropped, long short-term memory network (WDLSTM). CNN is used to extract features from IDS big data and WDLSTM network for learning dependencies among the extracted features to solve the overfitting problem. Their experimental results show a good performance with 97.1% accuracy.

III. Proposed Method

A. Enhanced Binary Genetic Algorithm

One of the most population evolutionary algorithms that mimics the nature selection is Genetic Algorithm (GA) [14]. GA is a population-based algorithm, where the best solution obtained after a predefined number of iterations. In simple, GA starts by generating a set of solutions called population. All these solutions are evaluated based on a fitness function. A set of genetic operations (i.e., selection, crossover, and mutation) are applied on the population at each iteration. This process is repeated iteratively until stop condition is met and return the best solution [15]. Fig. 2 explores the standard GA algorithm.

```
Given:
    -nP: base population size.
    -nI: number of iterations.
    -rC: rate of crossover.
    -rM: rate of mutation.
Generate initial population of size nP.
Evaluate initial population according to the fitness function.
While ($current\_iteration \leq nI$)
    //Breed $rC \times nP$ new solutions.
    Select two parent solutions from current population.
    Form offspring's solutions via crossover.
   IF($rand(0.0, 1.0) <$ rM)
      Mutate the offspring's solutions.
    end IF
   Evaluate each child solution according to the fitness function.
   Add offspring's to population.
   //population size is now MaxPop=nP× (1+rC).
   Remove the rC× nP least-fit solutions from population.
end While
Output the global best solution
```

Fig. 2. Standard Genetic Algorithm.

To enhance the performance of GA, we proposed a novel injection method based on solution distribution in the search space. At each iteration, we examined the solution distribution using intelligent k-means clustering algorithm, if 80% of the solutions located in one cluster, we redistribute the solution by injecting the population with new solutions to redistribute the solutions over the search space and prevent the premature convergence. This enhancement will enhance the exploration process of GA. Fig. 3 explores the flow chart of enhanced GA.

B. Long Short-Term Memory (LSTM) Networks

A deep learning method (i.e., CNN-LSTM) is employed to detect intrusions. Fig. 4 explores the main structure of CNN-LSTM. In simple, LSTM uses an internal memory to memorise the temporal sequence of the input feature vectors.

LSTM maps the input i (i.e., features) with output o (i.e., abnormal/normal packet), while forget f gate to memorize the store features. The hidden state h cell state c are used for memorizing. All calculations of LSTM are shown in Eqs.(1, 2, and 3).

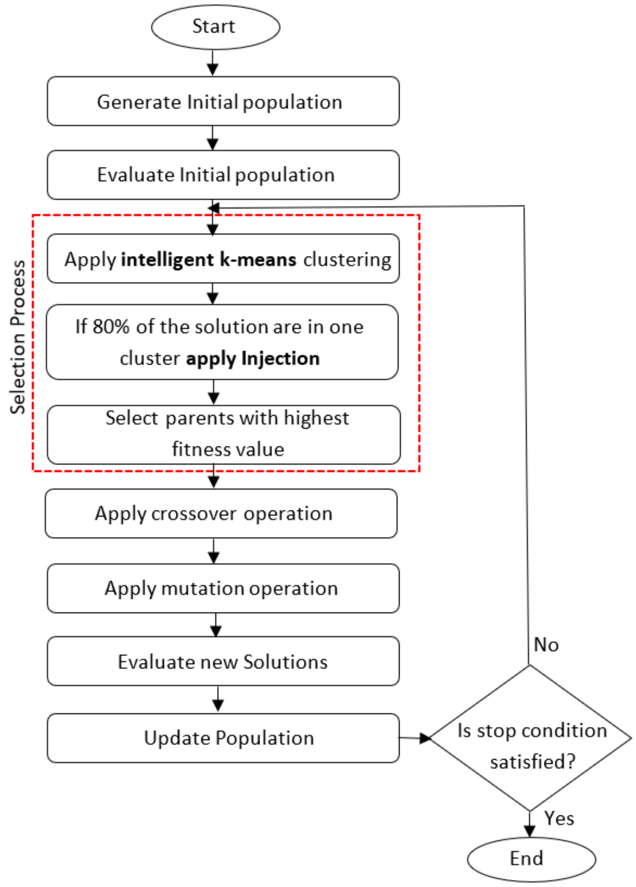
Start
Generate Initial population
Evaluate Initial population
Selection Process
Apply intelligent k-means clustering
If 80% of the solution are in one cluster apply Injection
Select parents with highest fitness value
Apply crossover operation
Apply mutation operation
Evaluate new Solutions
Update Population
Is stop condition satisfied?
No
Yes
End

Fig. 3. Enhanced Genetic Algorithm.

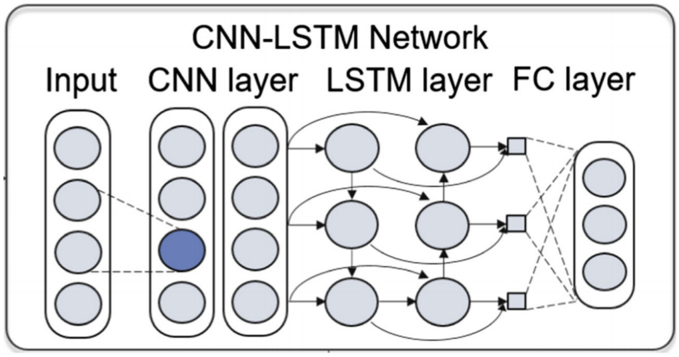

Fig. 4. The Main Structure of CNN-LSTM.

$$\begin{pmatrix} i\\ f\\ o\\ g\end{pmatrix} = \begin{pmatrix} sigmoid\\ sigmoid\\ sigmoid\\ tanh\end{pmatrix} w^t \begin{pmatrix} h_t^{l-1}\\ h_{t-1}^{l-1}\end{pmatrix} + \begin{pmatrix} b_i\\ b_f\\ b_o\\ b_g\end{pmatrix} \tag{1}$$

$$c_t = f_t \circ c_{t-1} + i_t \circ g \tag{2}$$

$$h_t = o_t \circ \sigma(c_t) \tag{3}$$

The calculation of fully connected layer and softmax process are shown in Eq. (4), and Eq.(5), respectively. In this work, we employed the softmax to classify the input user's role. While the output of the fully connected layer is presented by the softmax layer in a range [0,1]. Nc refers to the number of rules, and L presents the activity class probability.

$$d_i^l = \sum_i \sigma(W_{ji}^{l-1}(h_i^{l-1}) + b_i^{l-1}) \tag{4}$$

$$P(c|d) = argmax_{c \in C} \frac{exp(d^{L-1}w^L)}{\sum_{k=1}^{N_c}(d^{L-1}w_k)} \tag{5}$$

C. EBGA-LSTM

The proposed hybrid approach works by combining EBGA with LSTM. Here, EBGA works as a wrapper FS to remove the redundant/irrelevant data from the original dataset. while LSTM works as a binary classifier to detect normal and abnormal network traffic.

IV. Dataset

This paper evaluates the proposed hybrid approach over a public intrusion data set called UNSW-NB1. The data set is generated using a tool called IXIA PerfectStorm by Moustafa et al. [16]. The data set has 9 different types of attacks. The data set has 49 features. In this work, only 44 features are used. Table I explores 44 features of the data set. Moreover, this data set has 9 different attacks as shown in Table II.

UNSW-NB data set is imbalanced data set. In this work, adaptive synthetic sampling method (ADASYN) is employed for solving class imbalance issue [17]. Table III explores the original and balanced data set. In this work, this data set is used as a binary classification problem to determine normal or abnormal attacks.

V. Results and Analysis

This section reports the validation of the proposed hybrid method (i.e., EBGA with LSTM) to detect intrusion in cloud computing systems. All experiments are employed based on cross-validation method with kfold=10. We implemented the proposed approach using MATLAB 2019b. We used six criteria to evaluate the proposed method which are: accuracy (sSee Eq.(6), Specificity (see Eq.(7)), Precision (see Eq.(9)), Recall (see Eq.(10)), and F-Measure (see Eq.(11)).

$$Accuracy = \frac{TP + TN}{TP + FP + FN + TN} \tag{6}$$

$$Specificity = \frac{TN}{TN + FP} \tag{7}$$

$$Sensitivity = \frac{TP}{TP + FN}. \tag{8}$$

$$Precision = \frac{TP}{TP + FP} \tag{9}$$

$$Recall = \frac{TP}{TP + FN} \tag{10}$$

$$F - Measure = \frac{2 \times (Recall \times Precision)}{Recall + Precision} \tag{11}$$

Fig. 5 explores the performance of the original GA and EBGA as wrapper feature selection algorithms. Here, we used kNN as an internal classifier for all FS methods. The performance of EBGA outperform the original one with accuracy equals 88.7475, while the performance of original GA was the worst with accuracy equals 87.523. It is obvious here, the EBGA select 18 features out of 43, while the original GA select 11 features. The obtained results here give us a good indication that our proposed feature selection algorithm can explore the search space better than the original one.

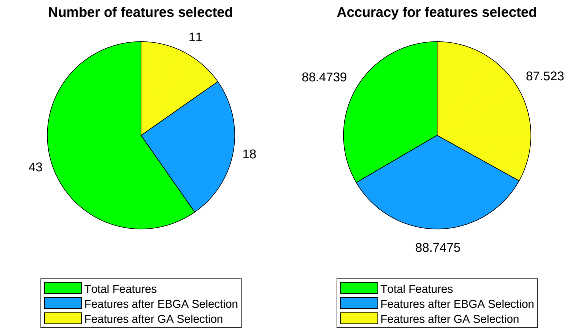

Fig. 5. Selected Features.

To perform a good analysis of the proposed approach, we simulated the proposed hybrid approach (i.e., EBGA with LSTM) with three settings; balanced data set with FS (i.e., EBGA), balanced data set without FS and Original data set without FS. Table IV explores the obtained results for three types of experiments. It is clear that the performance of feature selection improves the overall performance of LSTM compared to other experiments without feature selection. For example, the obtained results for testing data set show a good improvement (i.e., 6%) for the proposed method over balance data set. Fig. 6 explores the performance of LSTM in the training process. The classification error (i.e., RMSE) has a smooth convergence for balanced data with feature selection (i.e., blue line). Fig. 7 explores the loss convergence for the three experiments. It is clear that employing FS method helps LSTM to converge faster.

From the obtained results, we believe that the proposed method can enhance the overall performance of IDS inside cloud computing system.

TABLE I. Features of UNSW_NB15 Data Set.

Feature number	Feature Name	Type	Feature number	Feature Name	Type
1	id	Nominal	23	dtcpb	Integer
2	dur	Float	24	dwin	Integer
3	proto	Nominal	25	tcprtt	Float
4	service	Nominal	26	synack	Float
5	state	Nominal	27	ackdat	Float
6	spkts	Integer	28	smean	Integer
7	dpkts	Integer	29	dmean	Integer
8	sbytes	Integer	30	trans_depth	Integer
9	dbytes	Integer	31	response_body_len	Integer
10	rate	Integer	32	ct_srv_src	Integer
11	sttl	Integer	33	ct_state_ttl	Integer
12	dttl	Integer	34	ct_dst_ltm	Integer
13	sload	Float	35	ct_src_dport_ltm	Integer
14	dload	Float	36	ct_src_sport_ltn	Integer
15	sloss	Integer	37	ct_dst_src_ltm	Integer
16	dloss	Integer	38	is_ftp_login	Binary
17	sinpkt	Integer	39	ct_dtp_ltm	Integer
18	dinpkt	Integer	40	ct_src_ltn	Integer
19	sjit	Float	41	ct_srv_dst	Integer
20	djit	Float	42	ct_sm_ips_ports	Integer
21	swin	Integer	43	is_sm_ips_ports	Binary
22	stcpb	Integer	44	attack_cat	Nominal

TABLE II. Percentage of Attacks in UNSW-NB1 Dataset.

Attack type	Percentage%
Normal	87.94
Exploits	1.5
DoS	0.53
Backdoor	0.09
Analysis	0.09
Fuzzers	0.88
Generic	8.42
Reconnaissance	0.49
Shellcode	0.05
Worms	0.01

TABLE III. Original AND Balanced UNSW_NB15 Dataset.

Dataset	Number of Normal	Number of Attacks	Total
Original Training	56000	119341	175341
Original Testing	37000	45332	82332
Balanced Training	119341	119341	238682
Balanced Testing	45332	45332	90664

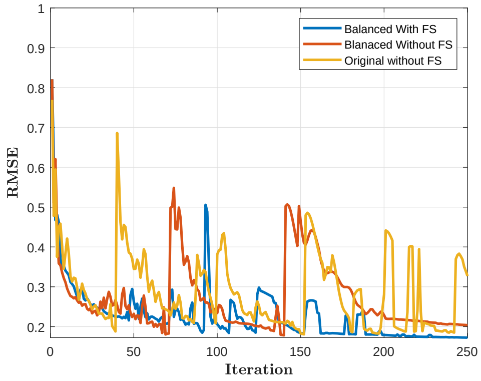

Fig. 6. LSTM Convergence for Original Training Data Set based on RMSE.

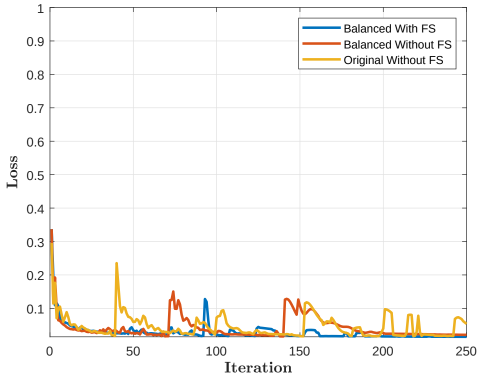

Fig. 7. LSTM Convergence for Original Training Data Set based on Loss.

VI. Conclusion and Future Works

This paper proposed a hybrid method between EBGA and LSTM to detect normal and abnormal network traffic. EBGA works as a wrapper feature selection, while LSTM works as binary classifier. The proposed method employed as IDS for could computing system. We examined the proposed approach over a real public data set called UNSW-NB15. The original data set is imbalanced one. We handled the imbalanced data set using ADASYN method. The obtained results show the importance of feature selection method and its ability of enhancing the classification accuracy. In future work, different feature selection methods such as Harris Hawks Optimization (HHO), Gray Wolf Optimization (GWO), and Whale Optimization Algorithm (WOA) will be applied to reduce the search space and determine the most important features for IDS systems.

References

[1] H. Alazzam, A. Sharieh, and K. E. Sabri, “A feature selection algorithm for intrusion detection system based on pigeon inspired optimizer,” *Expert Systems with Applications*, vol. 148, p. 113249, 2020. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0957417420300749

[2] O. Almomani, “A feature selection model for network intrusion detection system based on pso, gwo, ffa and ga algorithms,” *Symmetry*, vol. 12, no. 6, p. 1046, 2020.

TABLE IV. Comparison between Different Setting Over the Balanced Data Set.

	Balanced Data set with FS		**Balanced Data set without FS**		**Original Data set without FS**	
	Training	**Testing**	**Training**	**Testing**	**Training**	**Testing**
Accuracy	0.97	**0.91**	0.95	0.84	0.86	0.86
Sensitivity	0.95	**0.99**	0.94	0.84	0.96	0.88
Specificity	1.00	0.84	0.99	**0.85**	0.71	0.82
Precision	1.00	**0.88**	1.00	0.84	0.82	0.85
Recall	0.95	**0.99**	0.94	0.84	0.96	0.88
F-measure	0.98	**0.91**	0.97	0.86	0.89	0.87

[3] A. Thakkar and R. Lohiya, “Attack classification using feature selection techniques: a comparative study,” *Journal of Ambient Intelligence and Humanized Computing*, pp. 1–18, 2020.

[4] Y. Zhu, J. Liang, J. Chen, and Z. Ming, “An improved nsga-iii algorithm for feature selection used in intrusion detection,” *Knowledge-Based Systems*, vol. 116, pp. 74 – 85, 2017. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0950705116304245

[5] A. Javaid, Q. Niyaz, W. Sun, and M. Alam, “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)*, 2016, pp. 21–26.

[6] G. Zhao, C. Zhang, and L. Zheng, “Intrusion detection using deep belief network and probabilistic neural network,” in *2017 IEEE International Conference on Computational Science and Engineering (CSE) and IEEE International Conference on Embedded and Ubiquitous Computing (EUC)*, vol. 1. IEEE, 2017, pp. 639–642.

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

[8] T. A. Tang, L. Mhamdi, D. McLernon, S. A. R. Zaidi, and M. Ghogho, “Deep learning approach for network intrusion detection in software defined networking,” in *2016 International Conference on Wireless Networks and Mobile Communications (WINCOM)*. IEEE, 2016, pp. 258–263.

[9] Z. Wang, “The applications of deep learning on traffic identification,” *BlackHat USA*, vol. 24, no. 11, pp. 1–10, 2015.

[10] C. Yin, Y. Zhu, J. Fei, and X. He, “A deep learning approach for intrusion detection using recurrent neural networks,” *Ieee Access*, vol. 5, pp. 21 954–21 961, 2017.

[11] O. Faker and E. Dogdu, “Intrusion detection using big data and deep learning techniques,” in *Proceedings of the 2019 ACM Southeast Conference*, 2019, pp. 86–93.

[12] R. Vinayakumar, M. Alazab, K. Soman, P. Poornachandran, A. Al-Nemrat, and S. Venkatraman, “Deep learning approach for intelligent intrusion detection system,” *IEEE Access*, vol. 7, pp. 41 525–41 550, 2019.

[13] M. M. Hassan, A. Gumaei, A. Alsanad, M. Alrubaian, and G. Fortino, “A hybrid deep learning model for efficient intrusion detection in big data environment,” *Information Sciences*, vol. 513, pp. 386–396, 2020.

[14] J. H. Holland, *Adaptation in Natural and Artificial Systems: An Introductory Analysis with Applications to Biology, Control and Artificial Intelligence*. Cambridge, MA, USA: MIT Press, 1992.

[15] C.-L. Huang and C.-J. Wang, “A ga-based feature selection and parameters optimization for support vector machines,” *Expert Systems with Applications*, vol. 31, no. 2, pp. 231 – 240, 2006. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0957417405002083

[16] 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, pp. 1–6.

[17] Haibo He, Yang Bai, E. A. Garcia, and Shutao Li, “Adasyn: Adaptive synthetic sampling approach for imbalanced learning,” in *2008 IEEE International Joint Conference on Neural Networks (IEEE World Congress on Computational Intelligence)*, 2008, pp. 1322–1328.