International Journal of Future Generation Communication and Networking
Vol. 15, No. 1, (2022), pp. 56–68

BIE_{IDS}: Bio-Inspired Ensemble Method for Intrusion Detection System

Arshad Hashmi

Department of Information Systems, Faculty of Computing and Information Technology in Rabigh (FCITR), King Abdulaziz University, Jeddah, 21911,
Saudi Arabia
Mail Id: hashmi.arshad80@gmail.com

Abstract

Intrusion detection systems (IDSs) take part in information protection by detecting and preventing malevolent computer network actions. For many years, intrusion detection has been a prominent subject of study, and various intrusion detection systems have been cited in the literature. In this research work, a bio-inspired ensemble method is proposed for the intrusion detection system. IDS usually manages large quantities of data traffic containing redundant and inappropriate features that have an unhelpful effect on the IDS's performance. Using various dimensionality reduction techniques, unnecessary and improper features in the network traffic data are first eliminated using bio-inspired ensemble feature selection. Particle swarm optimization, Binary cuckoo search, and Fish swarm optimization algorithm are used to remove irrelevant features and select optimized features. The bagging and boosting ensemble classification model are built on the features chosen to detect the intrusion. The model and algorithms were simulated and tested using the available real-time datasets. The experimental results indicate that the proposed model can increase the detection rate and reduce the false alarm rate efficiently.

Keywords: *Intrusion detection, feature selection, ensemble model, classification, bio-inspired algorithms*

1. Introduction

Cyber-security measures are commonly used to defend against attack, disruption, and unauthorized access to information and computers. One part of the cyber protection scheme is Intrusion Detection System (IDS). By analyzing data collected via network devices, IDS is used to discover, assess and identify intrusions [1]. It is possible to describe an intrusion as an effort to obtain an illegal connection to system resources. In detecting threats from both outside and within networks, IDS has been exceptional. An IDS is an application or tool that can deal with threats from the internet or the intranet. To detect intrusions inside the network, it can map, log system traffic, and apply detection methods.

Signature and anomaly are the two main categories of IDS. The signature-based approach uses precise definitions, such as traffic monitoring guidelines or signatures [2]. Because such attacks do not appear in the preset pattern lists, they cannot safeguard the system from unknown threats. Maintaining an updated database becomes time-consuming and impossible as the number and diversity of network attacks increase. The system traffic is tracked and connected to the system's regular use activities by an anomaly-based IDS. Any deviation from standard usage patterns is interpreted as an attempt at interference. The anomaly detection method can identify novel assaults, whereas signature-based detection cannot detect novel attacks because they have not been previously described.

Several investigators concentrated on designing IDSs that leverage method of machine learning [3]. The range of machine learning methods[4][5] has attained an acceptable detection level; when adequate training data are presented, comprehensive hand-engineering features are designed to achieve ample overview and identify both attack varieties and new attacks. Employing machine learning methods, effective anomaly-based detection can be constructed. This implies solving a binary classification problem by supervised learning to determine if the network has normal or abnormal use patterns [6].

ISSN: 2233-7857 IJFGCN

An IDS works in various stages, such as information gathering, data preprocessing, selection of features, and Identification. Typically, many dimensions signify that some of them are redundant or insignificant, and their existence increases the learning algorithm error. Thus, feature selection has become a necessary preprocessing step to decrease the data dimensionality. Feature selection is a method of choosing important features; for classification, the selected features comprising a portion of the entire features are considered. Accuracy is a primary concern because, in real-time, IDS can detect a wide range of intrusions. The fundamental problem in designing IDS is ranking and choosing the sub-set of highly discriminating features.

A single feature selection approach may yield an optimal or suboptimal local subset of characteristics, for which a learning method sacrifices efficiency. The ensemble-based feature selection strategy combines many feature subsets to find an acceptable subset of features using a feature rating combination that improves classification accuracy [7]. A mixture of significant feature selectors are selected during the first stage of the ensemble method, and each selector produces a filtered list of features. The second stage aggregates the selected subsets of features using various techniques of aggregation. This paper uses three bio-inspired feature selection algorithms such as particle swarm optimization, binary cuckoo search and fish swarm optimization with two decision tree classification methods for selecting the best-optimized feature set. After selecting optimized features, two ensemble classification algorithms bagging and boosting, are used to classify the intrusion.

The remains of this article are planned as follows. Section 2 explains the related work of previous feature selection and intrusion detection methods. Section 3 describes the proposed bio-inspired ensemble IDS, and Section 4 discusses the performance assessment. At last, section 5 gives the conclusion and future enhancement.

2. Related Work

Different investigators have used machine learning algorithms and various freely accessible data for anomaly-based IDS analysis to achieve better detection performance. Machine learning-based intrusion detection methods are reviewed in this section. An enhanced genetic and deep belief network-based intrusion detection framework is proposed by Zhang et al. [8]. The optimum number of hidden layers and neurons are created dynamically through multiple iterations of the genetics. High recognition accuracy with a simple configuration is achieved by the interruption recognition model based on the network.

A new innovative approach was proposed and discussed in [9] for the anomaly intrusion detection system based on the mixture of composite feature selection and classifiers such as rotation forest and bagging classifier. Chkirbene et al. [10] proposed a Trust-based intrusion detection and classification technique that reduced the number of input characteristics using a unique feature selection method. [11] offers a network IDS model based on convolutional neural network-IDS.

Dimensionality reduction data features are automatically extracted using CNN, and supervised learning extracts more efficient information to detect intrusion. Kasongo et al. [12] present a deep learning-based IDS that combines neural networks with a filtering-based feature selection approach.

In [13], a learning-based IDS was developed that used a non-symmetric deep auto-encoder (NDAE) for attribute learning and a stacked NDAE classification. It is an auto-encoder consisting of several hidden layers that are non-symmetrical. Wang et al. [14] use FA and SVM to obtain enhanced features and produce efficient recognition accuracy results.

Sarvari et al. [15] developed an anomaly-based detection using a fuzzy-based cuckoo search algorithm for feature selection and a neural network for intrusion classification. This method utilizes transformation to test the investigate space more accurately to permit members to avoid local optima.

ISSN: 2233-7857 IJFGCN

Papamartzivanos et al. [16] suggest a new approach incorporating self-taught learning and the MAPE-K mechanism. This technique allows the misuse of IDS to maintain a high attack detection rate if placed on successive and dramatic alters in the setting.

A double Particle Swarm Optimization (PSO)-the based algorithm was suggested by Elmasry et al. [17] to pick both feature subsets and hyperparameters in one process. In the pre-training method, the earlier approach is utilized to determine the optimized features and model hyperparameters automatically. Benmessahel et al. [18] address feed-forward neural network training issues using the locust swarm optimization meta-heuristic optimization algorithm to construct a sophisticated detection system and enhance IDS performance. Ghanem et al. [19] propose a new binary IDS classification framework based on the Artificial Bee Colony Algorithm and the Dragonfly Algorithm for artificial neural network preparation to enhance classification precision for malicious and non-malicious networks traffic.

Shen et al. [20] describe an ensemble technique by an arbitrary subspace in which the base classifier is chosen as an extreme learning machine. The bat algorithm (BA) is suggested for optimizing the ensemble model. Meanwhile, the BA specifies a strength function based on an ensemble's precision and the mixture to obtain an enhanced classifier subset. Khammassi et al. [21] apply a genetic algorithm-based wrapper approach and logistic regression for selecting the optimal portion of features for network IDS. In order to improve IDS efficiency, a modified binary grey wolf optimization is suggested in [22].

3. Methodology BIE_{IDS}

This section explains the proposed bio-inspired ensemble feature selection for intrusion detection. The proposed architecture of BIEIDS is shown in figure 1. The proposed approach contains the Feature Selection and Classification phases. The subset of features is selected for anomaly detection in the feature selection phase. Three bio-inspired feature selection algorithms, such as particle swarm optimization (PSO), binary cuckoo search (BCS), and fish swarm optimization (FSO), are used to select the finest attribute subset. Ensemble classification algorithms such as bagging and boosting are used in the classification phase to build the classification model.

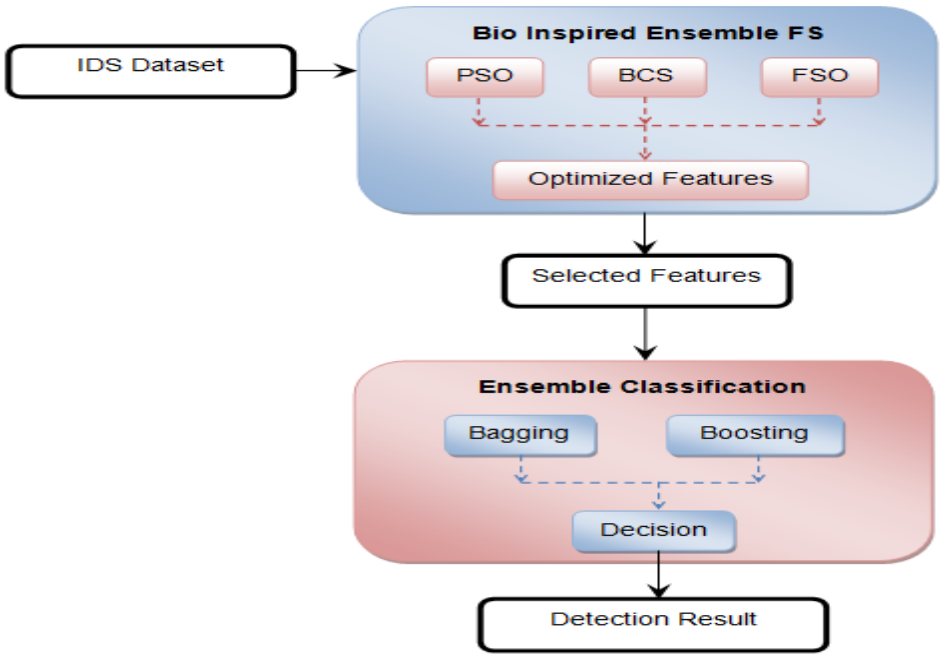


Figure 1 BIE IDS Architecture

ISSN: 2233-7857 IJFGCN

3.1 The Proposed BIE IDS model:

The architecture in the Figure1 represents the steps from data input to final IDS model creation. We collected the data from Kaggle and then performed feature selection as shown in figure 1. we proposed three Bio-inspired feature selection methods on both datasets as discussed in section 3.3.1 as Algorithm-1. The selection of features is choosing a portion of attributes from a collection of attributes to achieve the classifier's accurate, compact, and fast performance. In this phase, irrelevant and unnecessary features are removed from the data. The selected features are fed as an input to the ensemble classifier. Finally, the performance of the classification model was analyzed using various matrices. The following bio-inspired attribute selection algorithms are used to select the optimal feature set.

3.1 Particle Swarm Optimization (PSO):

Kennedy and Eberhart [23] suggest the PSO technique. The objection function is used to determine the consistency of the most recent solution. To overcome optimization issues, it utilizes the analogy of the grouping actions of birds. A feature set is represented in this technique by elements in a group. Several elements are put in a subspace in which every aspect has an arbitrary position xi and density vi. The rules for updating every element's location and velocity are as follows:

$$x_i(t+1) = x_i(t) + v_i(t+1) \quad (1)$$

$$v_i(t+1) = wv_i(t) + c_1 n_1\left(p_i - x_i(t)\right) + c_2 n_2\left(g - x_i(t)\right) \quad (2)$$

Where w = inactivity power, c_1 and c_2 = behavioral and social learning, n_1 and n_2 = random numbers. p_i and g = i^{th} particle best local and global location

3.2 Binary Cuckoo Search

Yang and Deb [24] suggested a novel meta-heuristic approach for continuous programming, called Cuckoo Search (CS), based on cuckoo birds' interesting technique for reproduction. In conventional cuckoo search, the solutions are modified into continuously valued positions in the search space. Unlike in the binary cuckoo search for feature selection, the global optimum is modelled as a Boolean n-dimensional structure in which the results are modified around a hypercube corner. Therefore, because the difficulty is in picking a particular attribute or not, a binary sequence solution is used, where 1 refers to whether an attribute is chosen to create the new set of data and 0 if not. Using Equation (4) to construct this binary vector which can only offer binary numbers in the binary structure, limiting the new results to binary values only:

$$S\left(x_i^j(t)\right) = \frac{1}{1+e^{-x_i^j(t)}} \quad (3)$$

$$x_i^j(t+1) = \begin{cases} 1 \text{ if } S\left(x_i^j(t)\right) > \sigma \\ 0 \text{ otherwise} \end{cases} \quad (4)$$

In which $\sigma \sim U(0,1)$ and $x_i^j(t)$ denotes the value of the new eggs at time step t.

3.3 Fish Swarm Optimization

An innovative swarm intelligent algorithm motivated by the normal schooling activities of fish named FSA was suggested by Li et al. [25] in 2009. The FSA has an excellent capacity to stay away from local minimums to enhance global optimization. The FSA repeats three distinctive habits: searching, swarming, and following. Searching is an arbitrary investigation for food with a propensity toward food focus. Swarming seeks to meet the requirements for food consumption, connect candidates of the swarm,

ISSN: 2233-7857 IJFGCN

and recruit new swarm candidates. Adjacent people chase the fish that find the food. The visual distance (visual), maximum step length (step), and crowd factor are the FSA parameters. The efficacy of the FSA is mainly affected by these parameters.

3.3.1 Algorithm-1

Procedure for feature selection Input: IDS Data set Output: Optimal Feature Set (OFS)
1. psobest = applyPSO() …….(3.1) 2.bcsbest = applyBCS() ……..(3.2) 3. fsobest= applyFSO() ……..(3.3) 4. R1 = ∩ (psobest, bcsbest, fsobest) 5. R2 = ranking (psobest, bcsbest, fsobest) 6. S1 = (R1 ∩ R2) 7. S2 = R2 – S1 8. OFS = OFS ∪ S1 9. EG = Compute Entropy and Gain (S2) 10. If EG > 0 then 11. OFS = OFS ∪ S2 12. End If

3.3.2 Working Example

Consider the KDD data set, which contains 41 features with one class label. The feature selection method removes the class label and takes only the remaining features.

The best selected features list are shown as follows (1 – selected, 0 – not selected)

psobest = 1 1 0 1 1 1 0 0 1 0 0 1 0 1 0 1 1 1 1 1 1 1 0 1 1 1 1 1 0 1 1 1 0 1 1 0 1 1 1 1 1

bcsbest = 0 1 0 0 1 0 0 1 0 1 1 0 0 0 0 1 1 0 1 1 0 0 0 0 0 1 0 0 0 0 1 1 0 1 1 1 1 1 0 1 1

fsobest = 1 1 0 0 1 1 0 1 1 1 1 1 1 1 1 1 1 0 1 1 1 0 0 1 0 1 0 0 0 1 1 0 1 1 0 1 1 1 1 1 1

R1 = 0 1 0 0 1 0 0 0 0 0 0 0 0 0 0 1 1 0 1 1 0 0 0 0 0 1 0 0 0 0 1 0 0 1 0 0 1 1 0 1 1

R2 = 1 1 0 0 1 1 0 1 1 1 1 1 0 1 0 1 1 0 1 1 1 0 0 1 0 1 0 0 0 1 1 1 0 1 1 1 1 1 1 1 1

S1 = 0 1 0 0 1 0 0 0 0 0 0 0 0 0 0 1 1 0 1 1 0 0 0 0 0 1 0 0 0 0 1 0 0 1 0 0 1 1 0 1 1

S2 = 1 0 0 0 0 1 0 1 1 1 1 1 0 1 0 0 0 0 0 0 1 0 0 1 0 0 0 0 0 1 0 1 0 0 1 1 0 0 1 0 0

OFS = 1, 4, 15, 16, 18, 19, 25, 30, 33, 36, 37, 39, 40

Compute Entropy and Gain (EG) for the following features 0, 5, 7, 8, 9, 10, 11, 13, 20, 23, 29, 31, 34, 35, 38

EG > 0 for the following features 7, 8, 10, 11, 13, 20

Final OFS = 1, 4, 7, 8, 10, 11, 13, 15, 16, 18, 19, 20, 25, 30, 33, 36, 37, 39, 40

The two ensemble classifiers bagging and boosting, are used to classify the intrusion. Bagging and Boosting are meta classifiers. Bagging uses sampling with substitution to construct base classifiers of a

ISSN: 2233-7857 IJFGCN

training set. Each sample of the bootstrap is often used to learn a specific base classifier part. The classification is achieved by majority voting while bagging. Boosting is a several base classifier combination strategies whose cumulative output is substantially higher than that of some base classifiers. Each base classifier is trained on data that is measured based on the prior classifier's results. Each classifier votes to achieve the final result.

4. Experimental Result and Discussions

This section evaluates the proposed work's performance through experiments. This IDS classification was developed in Java (version 1.8), and the experiments were carried out on an Intel(R) Pentium(R) system with a clock speed of 2.13 GHz and 4.0 GB RAM running Windows 10 64-bit.

4.1 Dataset Description

KDDCUP99 [26] and NSL-KDD [26] have been the generally used data sets in the intrusion detection study. This research work uses NSL-KDD and UNSW-NB15 [27] IDS data set. NSL-KDD is an enhanced form of the KDDCup 99 data collection that does not use redundant tests, preventing classifiers from being biased. It has 41 features that have class label properties. UNSW-NB15 is an original version of a more recently published intrusion detection dataset. The data set is composed of 42 attributes with class label attributes.

4.2 Performance Metrix

The subsequent metrics are used to estimate the performance of the proposed work

$$Accuracy\ (ACC) = \frac{TP+TN}{TP+TN+FP+FN} \quad (5)$$

$$Detection\ Rate\ (DR) = \frac{TP}{TP+FN} \quad (6)$$

$$False\ Alarm\ Rate\ (FAR) = \frac{FP}{TN+FP} \quad (7)$$

$$Precision = \frac{TP}{TP+FP} \quad (8)$$

$$Recall = \frac{TP}{TP+FN} \quad (9)$$

4.3 Discussions

Figure 2 shows the feature selection accuracy comparison of the NSL-KDD dataset for different size of data. From that result, the FSO algorithm has higher accuracy compared to the other two algorithms.

ISSN: 2233-7857 IJFGCN

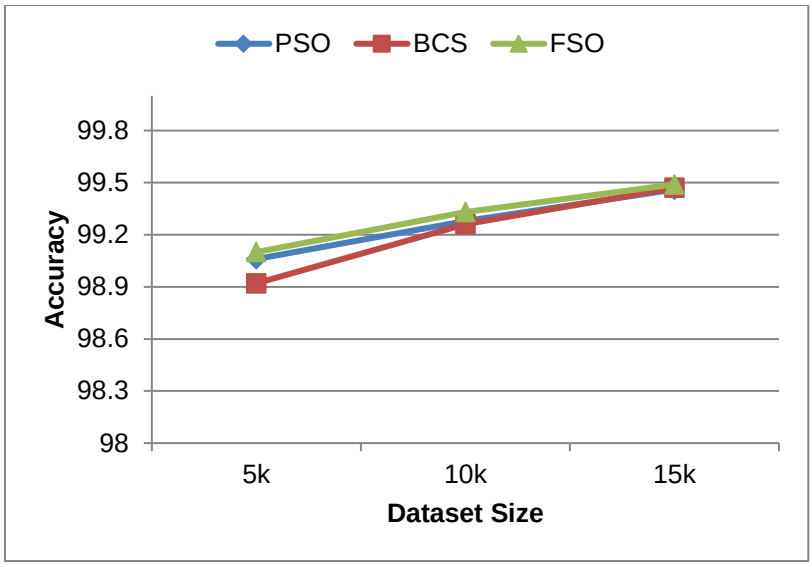


Figure 2 FS Accuracy Comparison for NSL-KDD Dataset

Figure 2 shows the accuracy comparison of PSO, BCS, and FSO for different dataset sizes. It is found that FSO achieved the best accuracy for all three dataset sizes, as shown in the figure.

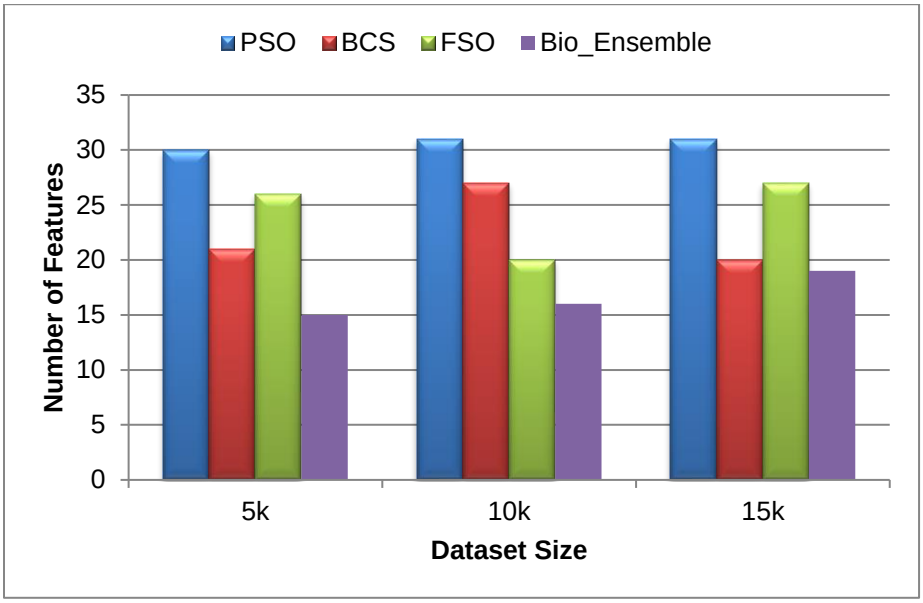


Figure 3 Selected features count on NSL-KDD Dataset

Figure 3 shows the selected feature count of the NSL-KDD dataset for different sizes of data. The bio-inspired ensemble feature reduction method significantly reduces the number of features compared to other algorithms.

ISSN: 2233-7857 IJFGCN

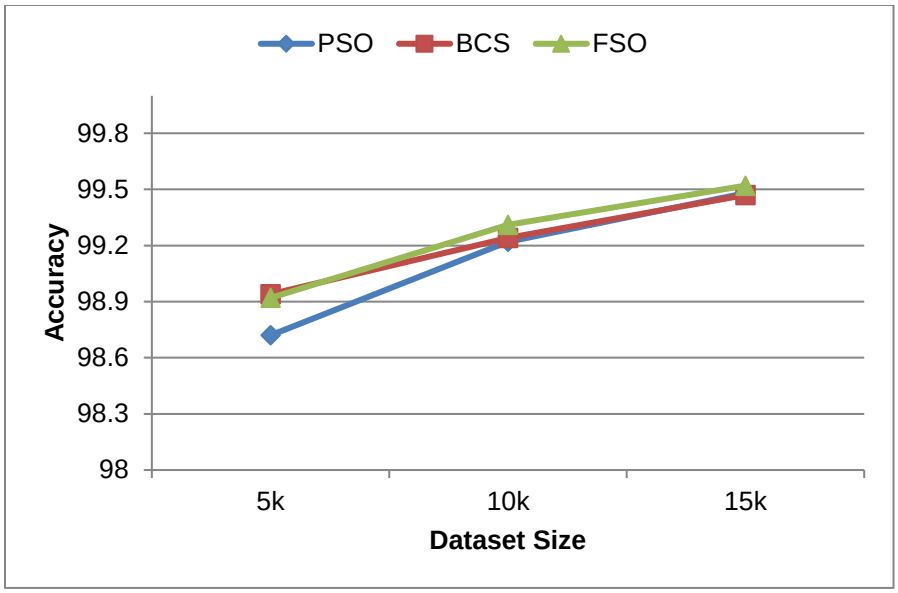


Figure 4 Accuracy Comparison for UNSW_NB15 Data

Figures 4 and figure 5 show the result of UNSW_NB15 data. The bio ensemble feature selection method reduces the number of features and has higher accuracy.

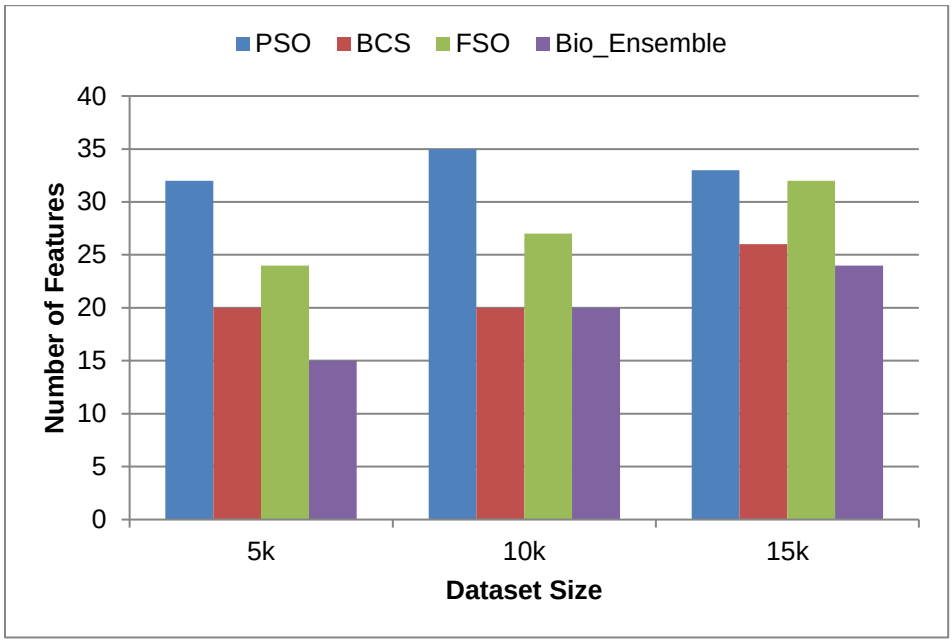


Figure 5 Selected features Count on UNSW_NB15

Table 1 shows the performance comparison of ensemble classification algorithms.

Table 1 Performance Comparison

Dataset	Size	Bagging			Boosting			BIE_IDS		
		ACC	DR	FR	ACC	DR	FR	ACC	DR	FR

ISSN: 2233-7857 IJFGCN

			Detection Rate	False Alarm Rate						
NSL-KDD	5k	97.99	98.2	2.01	92.90	92.94	7.13	98.25	98.24	1.725
	10k	98.96	98.99	1.06	94.03	94.06	5.99	98.97	99	1.049
	15k	99.41	99.42	0.595	94.26	94.46	5.93	99.46	99.47	0.54
UNSW_NB	5k	89.27	98.54	2.99	75.81	97.08	3.78	90.84	98.56	1.893
	10k	88.02	99.33	1.23	73.99	98.59	3.12	93.05	99.36	0.69
	15k	88.44	99.56	1.673	72.17	99.01	1.78	92.72	99.61	0.51

The proposed BIE_IDS has high detection accuracy and a low false alarm rate than the individual ensemble classifier. Accuracy is the overall model correctness, as shown in equation 5. From Table.1, an accuracy comparison for the intrusion dataset is noted. The proposed BIEDS algorithm has the highest accuracy for the NSL-KDD dataset for 5k and 15k size as shown 98.25,99.46 respectively, but for 10k size accuracy is 98.97, slightly low compared to Bagging, but it is more significant than boosting. However, BIEDS showed better classification results than existing methods.

Detection rate (DR) is the percentage of TP, which are an intrusion and real feature class, as shown in equation 6. From Table.1 DR comparison for the intrusion dataset is noted. For both datasets, the result showed that the detection rate for the BIEDS is comparatively more significant than the existing method. The proposed BIEDS algorithm has a higher detection rate than existing methods, accounting for better classification results.

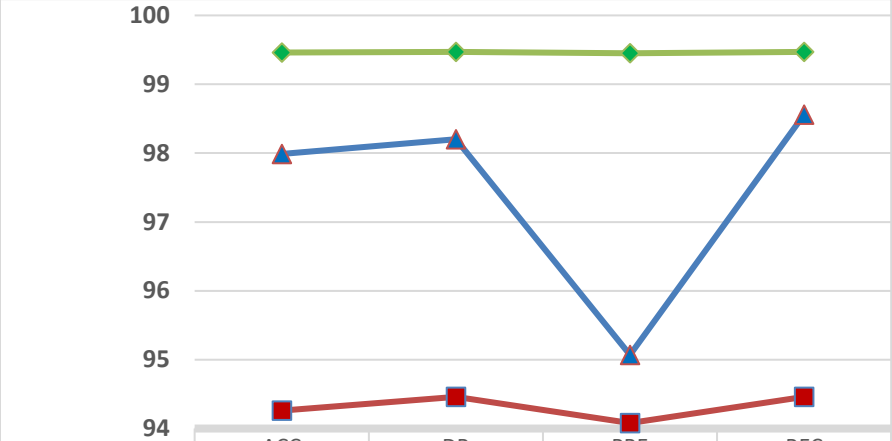
Further, in Table1, performance of the proposed method is listed in terms of the DR, accuracy, and FAR with different sizes. According to the achieved results, increasing the size directly affects the performance metrics. When the size of the NSL-KDD dataset is expanded from 5k to 10k, the detection rate (DR) increases from 98.24 to 99. On the other hand, accuracy showed an enhanced value from 98.25 to 98.97, while the FAR decreased from 1.725 to 1.049.

In a similar way, on increasing the size from 10k to 15k, the DR showed an enhanced value from 99 to 99.47. Also, the accuracy value showed enhancement from 98.97 to 99.46. But the FAR decreased from 1.049 to 0.54. It is evident from the findings that all of the suggested method's matrices outperformed other approaches.

When the size of the UNSW NB dataset expanded from 5k to 10k, the DR and accuracy climbed from 98.56 to 99.36 and 90.84 to 93.05, respectively, while the FAR decreased from 1.893 to 0.69. When the size was

ISSN: 2233-7857 IJFGCN

expanded from 10k to 15k, the DR and accuracy increased from 99.36 to 99.61 and 93.05 to 99.46, respectively, while the FAR decreased from 1.049 to 0.54. It is clear from the results that the entire matrix



	ACC	DR	PRE	REC
Boosting	94.26	94.46	94.08	94.46
Bagging	97.99	98.2	95.07	98.56
BIEIDS	99.46	99.47	99.45	99.47

Figure 6 Performance metrics for NSL-KDD Data

In figure 6 performance matrix of NSL-KDD data is shown in terms of the ACC, DR, PRE, and REC as discussed in the equation 5,6,8 and 9, respectively. According to the achieved results, it is evident from figure 6 and its corresponding table that BIEIDS achieved the best performance among others.

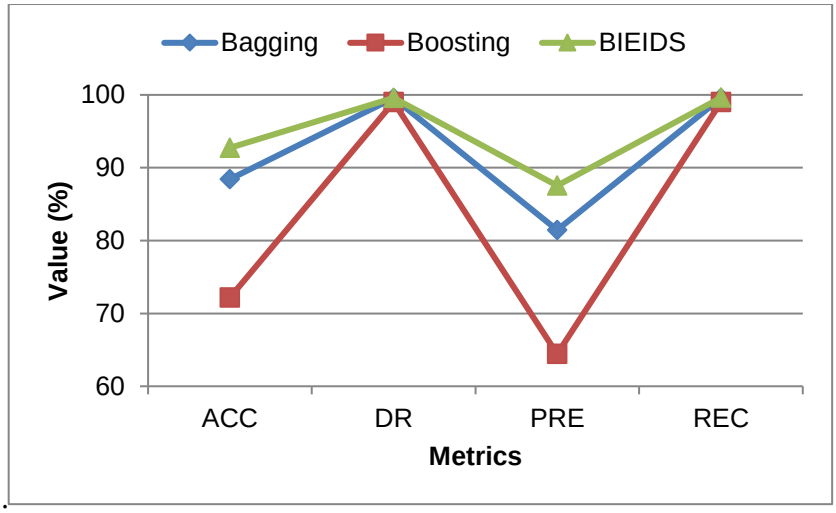


Figure 7 Performance metrics for UNSW_NB15

The performance matrix of UNSW_NB15 data is shown in figure 7 in terms of the ACC, DR, PRE, and REC. It is evident from the figure that BIEIDS achieved the highest accuracy, 92.72, while bagging an accuracy value of 88.44 and boosting accuracy value is 72.17, which is the least. Further, it is observed that BIEIDS achieved the highest DR value of 99.61. The boosting method obtained the most negligible DR value of 99.01. The proposed BIEIDS obtained the best precision value of 87.54, while the boosting method reached the value of 64.43. Finally, in terms of recall, the best value, 99.613, is achieved by

ISSN: 2233-7857 IJFGCN

BIEIDS, while boosting obtained the most negligible value of 99.01. Therefore, BIEIDS showed the overall best performance, and the proposed approach considerably enhanced the detection rate for both datasets.

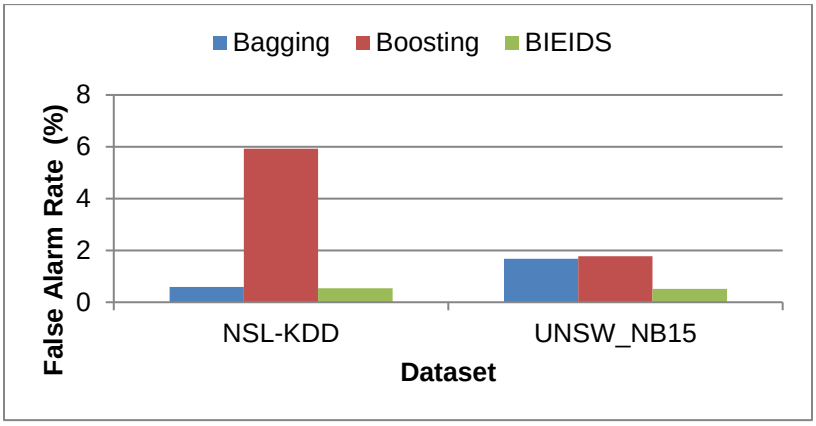


Figure 8 False Alarm Rate Comparison

Figure 8 shows the achieved experimental result of the False alarm rate in comparison to the NSL-KDD and UNSW_NB15. It is found from the experiment BIEIDS false alarm rate value of 0.54 is the least among others bagging (0.595) and boosting(5.93) for the NSL-KDD. Further, for the UNSW_NB15 dataset, the false alarm rate value obtained by BIEIDS is 0.51. On the otherhand false alarm rate value obtained are 1.673 and 1.78 respectively for bagging and boosting methods.

5. Conclusion

The bio-inspired method of feature selections and ensemble classifiers for IDS is proposed to classify the intrusion efficiently. Three methods for feature selection PSO, BCS, and FSO are involved in achieving the best features. Besides, two ensemble classifier algorithms bagging and boosting, are used for classification analysis. To assess the efficiency of the proposed solution, publicly available intrusion datasets were used. It could be argued, based on the statistical significance analyses, that the proposed methodology betters the other previous approaches. The future direction is to use big data with the deep learning concept for intrusion detection.

6. Acknowledgement:

The authors would like to thank the Dean of FCIT in Rabigh, King Abdulaziz University for providing the outstanding platform for doing research work**.**

References

[1] A. L. Buczak and E. Guven, "A survey of data mining and machine learning methods for cyber security intrusion detection", IEEE Communications Surveys Tutorials, vol. 18, no. 2, pp. 1153–1176, 2016.

[2] K. Kim and M. E. Aminanto, "Deep learning in intrusion detection perspective: Overview and further challenges", in Proc. 2017 Int. Workshop on Big Data and Information Security (IWBIS), pp. 5–10, 2017

[3] H. Liu and B. Lang, "Machine learning and deep learning methods for intrusion detection systems: A survey", Appl. Sci., vol. 9, pp. 4396, 2019.

[4] A. Ahmim, M. Derdour and M. A. Ferrag, "An intrusion detection system based on combining probability predictions of a tree of classifiers", Int. J. Commun. Syst., vol. 31, no. 9, 2018.

ISSN: 2233-7857 IJFGCN

[5] Z. Xiaofeng and H. Xiaohong, "Research on intrusion detection based on improved combination of K-means and multi-level SVM", Proc. IEEE 17th Int. Conf. Commun. Technol., pp. 2042-2045, 2017.

[6] B. A. Tama and K.-H. Rhee, "An in-depth experimental study of anomaly detection using gradient boosted machine," Neural Comput. Appl., vol. 31,no. 4, pp. 955–965, 2019.

[7] N. Hoque, M. Singh, D.K. Bhattacharyya, "EFS-MI: an ensemble feature selection method for classification", Complex Intell. Syst., vol. 4, pp. 105–118, 2018

[8] Y. Zhang, P. Li and X. Wang, "Intrusion Detection for IoT Based on Improved Genetic Algorithm and Deep Belief Network," in IEEE Access, vol. 7, pp. 31711-31722, 2019

[9] B. A. Tama, M. Comuzzi and K. Rhee, "TSE-IDS: A Two-Stage Classifier Ensemble for Intelligent Anomaly-Based Intrusion Detection System," in IEEE Access, vol. 7, pp. 94497-94507, 2019

[10] Z. Chkirbene, A. Erbad, R. Hamila, A. Mohamed, M. Guizani and M. Hamdi, "TIDCS: A Dynamic Intrusion Detection and Classification System Based Feature Selection," in IEEE Access, vol. 8, pp. 95864-95877, 2020

[11] Y. Xiao, C. Xing, T. Zhang and Z. Zhao, "An Intrusion Detection Model Based on Feature Reduction and Convolutional Neural Networks," in IEEE Access, vol. 7, pp. 42210-42219, 2019

[12] S. M. Kasongo and Y. Sun, "A Deep Learning Method With Filter Based Feature Engineering for Wireless Intrusion Detection System," in IEEE Access, vol. 7, pp. 38597-38607, 2019

[13] N. Shone, T. N. Ngoc, V. D. Phai, and Q. Shi, "A deep learning approach tonetwork intrusion detection," IEEE Trans. Emerg. Topics Comput. Intell., vol. 2, no. 1, pp. 41–50, Feb. 2018.

[14] H. Wang, J. Gu, and S. Wang, "An effective intrusion detection framework based on SVM with feature augmentation", Knowl.-Based Syst., vol. 136,pp. 130–139, Nov. 2017.

[15] S. Sarvari, N. F. Mohd Sani, Z. Mohd Hanapi and M. T. Abdullah, "An Efficient Anomaly Intrusion Detection Method With Feature Selection and Evolutionary Neural Network," in IEEE Access, vol. 8, pp. 70651-70663, 2020

[16] D. Papamartzivanos, F. Gomez Marmol, and G. Kambourakis, "Intro-ducing deep learning self-adaptive misuse network intrusion detectionsystems," IEEE Access, vol. 7, pp. 13546–13560, 2019

[17] W. Elmasry, A. Akbulut, and A. H. Zaim, "Evolving deep learning archi-tectures for network intrusion detection using a double PSO Metaheuristic," Comput. Netw., vol. 168, Feb. 2020

[18] I. Benmessahel, K. Xie, M. Chellal, and T. Semong, "A new evolutionary neural networks based on intrusion detection systems using locust swarmoptimization," Evol. Intell., vol. 12, no. 2, pp. 131–146, 2019.

[19] W. A. H. M. Ghanem, A. Jantan, S. A. A. Ghaleb and A. B. Nasser, "An Efficient Intrusion Detection Model Based on Hybridization of Artificial Bee Colony and Dragonfly Algorithms for Training Multilayer Perceptrons," in IEEE Access, vol. 8, pp. 130452-130475, 2020

[20] Y. Shen, K. Zheng, C. Wu, M. Zhang, X. Niu and Y. Yang, "An Ensemble Method based on Selection Using Bat Algorithm for Intrusion Detection," in The Computer Journal, vol. 61, no. 4, pp. 526-538, April 2018

ISSN: 2233-7857 IJFGCN

[21] C. Khammassi and S. Krichen, "A GA-LR wrapper approach for feature selection in network intrusion detection," Comput. Secur., vol. 70,pp. 255–277, Sep. 2017.

[22] Q.M. Alzubi, M. Anbar, Z.N.M Alqattan, et al. "Intrusion detection system based on a modified binary grey wolf optimization", Neural Comput & Applic, vol. 32, pp. 6125–6137, 2020

[23] J. Kennedy and R.C. Eberhart, “A discrete binary version of the particle swarm algorithm,” IEEE International Conference on Systems,Man, and Cybernetics – Computational Cybernetics and Simulation,pp.4104–4108, IEEE, 1997.

[24] X.-S. Yang and S. Deb, “Cuckoo search via lvy flights”, in Proceedings of the NaBIC 2009 - World Congress on Nature & Biologically Inspired Computing, pp. 210–214, 2009

[25] X. L. Li, Z. J. Shao, J. X. Qian,“An optimizing methodbased on autonomous animats: fish-swarm algorithm”, [J].System engineering theory and practice, vol. 22, no. 11, pp. 32-38, 2002

[26] M. Tavallaee, E. Bagheri, W. Lu, and A. Ghorbani, "A detailed analysis ofthe KDD CUP 99 data set," Proc. 2nd IEEE Symp. Comput. Intell. Secur.Defense Appl. (CISDA), Ottawa, ON, Canada, Jul. 2009, pp. 1–6.

[27] N. Moustafa and J. Slay, "UNSW-NB15: A comprehensive data set fornetwork intrusion detection systems (UNSW-NB15 network data set)," inProc. Mil. Commun. Inf. Syst. Conf. (MilCIS), Nov. 2015, pp. 1–6

ISSN: 2233-7857 IJFGCN