A Supervised Feature Selection Method for Malicious Intrusions Detection in IoT Based on Genetic Algorithm

Saman Iftikhar[1], Daniah Al-Madani[1], Saima Abdullah[2], Ammar Saeed[3], Kiran Fatima[4]
[1]Faculty of Computer Studies, Arab Open University, Saudi Arabia,{s.iftikhar, d.almadani}@arabou.edu.sa
[2]Department of Computer Science, The Islamia University of Bahawalpur, Bahawalpur, Pakistan, saima.abdullah@iub.edu.pk
[3]Department of Computer Science, COMSATS University Islamabad, Wah Campus, Wah Cantt Pakistan, ammarsaeed1997@gmail.com
[4]TAFE - New South Wales, Australia, kiran.fatima4@tafensw.edu.au

Abstract
Machine learning methods diversely applied to the Internet of Things (IoT) field have been successful due to the enhancement of computer processing power. They offer an effective way of detecting malicious intrusions in IoT because of their high-level feature extraction capabilities. In this paper, we proposed a novel feature selection method for malicious intrusion detection in IoT by using an evolutionary technique – Genetic Algorithm (GA) and Machine Learning (ML) algorithms. The proposed model is performing the classification of BoT-IoT dataset to evaluate its quality through the training and testing with classifiers. The data is reduced and several preprocessing steps are applied such as: unnecessary information removal, null value checking, label encoding, standard scaling and data balancing. GA has applied over the preprocessed data, to select the most relevant features and maintain model optimization. The selected features from GA are given to ML classifiers such as Logistic Regression (LR) and Support Vector Machine (SVM) and the results are evaluated using performance evaluation measures including recall, precision and f1-score. Two sets of experiments are conducted, and it is concluded that hyperparameter tuning has a significant consequence on the performance of both ML classifiers. Overall, SVM still remained the best model in both cases and overall results increased.

Keywords:
Evolutionary computing; Genetic Algorithm (GA); Internet of Things; BoT-IoT dataset; Malicious Intrusions; Support Vector Machine (SVM); Logistic Regression (LR).

1. Introduction

Information technologies, including IoT systems, must meet the three goals of security: availability, confidentiality, and integrity. Whereas availability signifies that the system should be able to offer the services for which it was implemented at all times, confidentiality entails preventing unauthorized access to information. The system should also exhibit integrity in terms of ensuring that the data, software, and hardware are not altered. Based on Buczak et al., at the very minimum, computer systems have anti-malware, firewalls, and intrusion detection systems [2]. The role of these systems is to detect and deal with malware and other risks affecting a network. Firewalls and intrusion detection systems, in particular, implement a variety of tools and strategies to identify and prevent bad traffic from entering the network. The challenge associated with this approach is that attack signatures must be defined and policies for identifying malicious traffic configured. Security threats and attacks on IoT are described in Fig. 1. However, in a setting where attacks continue to evolve at a fast rate, the use of machine learning and deep learning algorithms can be far more effective and efficient. By reducing human intervention, intrusion detection systems and firewalls can offer automated identification in an efficient and effective way.

According to Alhajri et al., some of the areas of risk associated with IoT systems that could be compromised encompass weak interaction protocols, non-user interfaces, vulnerabilities in the middleware layer, sensitive data modification, non-availability of the storage volume, and the multiplicity of attacks. Similarly, some of the attack vectors associated with these vulnerabilities encompass insecure login credentials, malware distribution, and unauthorized physical access to devices and their operating systems [1]. The occurrence of successful attacks targeting IoT systems can present dire consequences considering that these devices collect and store sensitive information. In addition to personal information, these devices contain sensors that take private images. From a legal and regulatory perspective, the need to ensure data privacy and confidentiality is paramount.

Machine learning, which encompasses machines, performs tasks without explicit programming, offers a promising way of detecting malware. According to Buczak et al., there are three main methods of operation for intrusion detection systems (IDS): signature-based, anomaly-based, and hybrid [2]. As the name suggests, signature-based

Manuscript received March 5, 2023
Manuscript revised March 20, 2023
https://doi.org/10.22937/IJCSNS.2023.23.3.5

techniques detect attacks by utilizing known signatures of those attacks. Therefore, this approach is effective for detecting known attacks but it needs regular updating of the database with signatures and rules. For this reason, signature-based approaches cannot detect zero-day attacks. Anomaly-based methods model the typical network behavior and identify abnormalities as plausible malware. This approach is advantageous as it offers a fairly effective method for dealing with both known and novel attacks [2]. The final hybrid approach combines both signature-based and anomaly-based methods. Machine learning techniques are routinely utilized to model the network behavior and identify anomalies. However, due to the high false positives, they often adopt the hybrid approach. By combining known malware signatures and the network behavior, the ability of the intrusion detection system to detect malware accurately increases. In addition to whether it is supervised, semi-supervised, or unsupervised, the selection of an algorithm should also be based on its accuracy, recall, and precision.

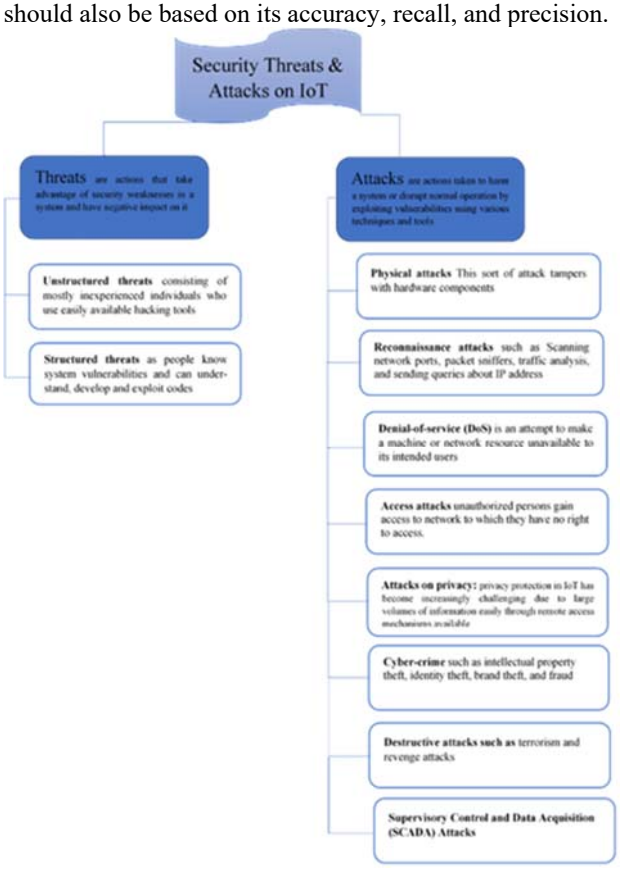

Fig. 1. Security threats and attacks on IoT.

In this paper, we are exploiting various machine learning algorithms to detect malicious intrusions based on feature selection method in IoT by using an evolutionary technique – Genetic Algorithm (GA) and ML algorithms. In the proposed work, a model is developed for the classification of BoT-IoT dataset. The data is shortened and several preprocessing steps are applied over it including unnecessary information removal, null value checking, label encoding, standard scaling and data balancing. To select the most relevant features and maintain model optimization, GA is applied over the preprocessed data. The selected features from GA are given to ML classifiers LR and SVM and the results are evaluated using evaluation measures. Two sets of experiments were conducted; where it has concluded that hyperparameter tuning has a significant effect on the performance of both ML classifiers. SVM still remains the best model in both cases and overall results increased.

The rest of this paper is structured as follows; section 2 provides a review of the literature in the field of current research. Section 3 presents a description of the botnet IoT dataset and the proposed methodology. The results of the proposed approach have been given in section 4 for intrusion detection of malicious activities in IoT networks. The overall idea behind the research has been concluded in section 5.

2. Related Work

In the literature, machine learning techniques based on feature selection have been used to detect intrusion detection in IoT and network-based traffic [3-8].

Abdullah et al. [3] illustrated the application of an ensemble method in the detection of network anomalies. The system was based on dividing the input into different subsets based on the attack in question. Thereafter, the researchers performed a feature selection technique using a filter for each subset of attack. As a result, optimal feature sets were developed by combining feature sets for each attack. The optimal features were implemented in a combination of algorithms to enhance the detection accuracy of the system. Indeed, feature selection must consider the two, often contradicting, goals of reducing complexity and computational needs and increasing the accuracy of detection.

Liu et al. [28] presented a semi-supervised dynamic ensemble for detecting anomalies in IoT environments. The algorithm combined mutual information criteria and semi-supervised extreme learning machine. Experiments conducted on practical datasets showed that the proposed algorithm outperformed selected state-of-the-art approaches in terms of classification accuracy. Evidently, ensemble approaches appear to be a promising direction of research mainly because of their ability to minimize biases and increase accuracy.

Perhaps the most important deduction when examining machine learning methods for detecting anomalies is that feature extraction and selection is imperative. Ultimately, the capacity of a classifier to detect anomalies accurately depends on the features selected. To this end, various studies were found that examined feature extraction and selection in the detection of anomalies in the IoT environment [4-8].

Egea et al. [4] proposed a modification to the fast-based-correlation feature (FCBF) algorithm, a popular algorithm for feature selection. Their approach encompassed splitting the feature space into fragments of similar sizes hence improving correlation and, as a result, the operation effectiveness of machine learning applications. The authors demonstrated the improvement in accuracy by utilizing the developed feature selection algorithm with different classifiers. Another study conducted by Zhang et al. [5] also entailed designing a hybrid feature selection algorithm that pre-filters most of the features with the Weighted Symmetrical Uncertainty (WSU) metric and further utilizes a wrapper technique to choose features for each classifier with Area Under ROC Curve (AUC) metric. Furthermore, the authors proposed an algorithm referred to as SRSF, which Selects the Robust and Stable Features from the outcomes obtained from WSU_AUC, to overcome the challenges of dynamic traffic flows. Experimental results showed that this algorithm attains a flow accuracy of more than 94%.

Su et al. [6] developed a correlation-change based feature selection method for detecting anomalies in IoT equipment. The method encompassed clustering correlated sensors together to identify the duplicated sensors and monitoring changes in data correlation in real time to select sensors with correlation changes. The sensors with correlation changes were chosen as representative features for anomaly detection. The experimental evaluation showed that this approach reduces false negatives and false positives in anomaly detection by about 30%. This study is particularly insightful as it offers information about some of the features that ought to be considered in an anomaly detection system for IoT equipment. Similarly, Alhakami et al. [7] combined feature selection with a non-parametric Bayesian approach to detect network anomaly intrusion.

In another study, Shafiq et al. [8] demonstrated an effective approach for selecting features for 5G instant messaging applications in classifying traffic. The proposed hybrid feature selection algorithm named weighted mutual information (WMI) filters features with the MWI metric and then utilizes a wrapper technique to choose features for machine learning classifiers with accuracy (ACC) metric. The algorithm was evaluated with different datasets and classifiers and achieved satisfactory outcomes. The capability to classify traffic accuracy offers a mechanism for detecting anomalies as well.

In [9], the authors present a review that details several types of IDS, such as misuse based, host-based, anomaly-based, hybrid-based, and network-based. However, it is interested in behavior based and anomaly-based n real network traffic. In [10], a review interest with using Machine Learning (ML) techniques in the IDS. In this study, the applications into a system with ML discussed. In addition, a detailed comparison of different methods for the IDS based ML is presented. This paper resulted in that it is difficult to train the ML methods with an insufficient or not available amount of traffic data. IDS developed in [11] for network behavior identification using K-Means and RF. Moreover, the proposed model shows that the system based Random Forest algorithm presents a good indication in the term of classification accuracy detection correctness.

The proposed genetic algorithm (GA) presented in [12] used Decision Tree (DT) classifier to locate the combination of features that can correctly identify the behaviour into normal and botnet attacks. UNSW-NB15 and CICIDS2017 dataset are used as evaluation datasets. The experimental results reveal that the proposed system can effectively identify the relevant features from the whole features set. An evolutionary technique-based feature selection method and a Random Forest-based classifier are in [13] to select the essential features and reduces dimensions of the data, which improve the True Positive Rate and reduce the False Positive Rate at the same time. UNSW-NB15 datasets and NSL-KDD datasets used in this framework for the evaluation process. Many statistical results and detailed comparison to other existing approaches are presented in this research.

Genetic neural feed-forward network utilized to propose anomaly detection model [14]. The main concept of this model is Searching for the best setting for the initial weights of backpropagation feed-forward neural networks. The UNSW-NB15 dataset is analyzed practically and statistically in [15]. The authors used different classifiers, such as: Decision Tree (DT), Naïve Bayes (NB), Logistic Regression (LR), Artificial Neural Network (ANN), and the clustering method - Expectation-Maximization (EM) are exploited to assess the complexity in terms of accuracy and false alarm rate.

A single objective GA is utilized in [16] to explore the high space of candidate features and attempt locate the best subset that could enhance the G performance-based botnet detection system. In [17], three different feature selection methods have been developed to define the impact of feature selection on the performance of a botnet detection method. The authors in [18], used Random Forest technique to select notable features and SVM to improve the classification result. To reach a higher attack detection rate, only 14 features (in total 41 features) are utilized also the KDDCUP99 dataset is used.

Sarker et al., proposed an algorithm based on decision trees to detect intrusions [24]. The algorithm ranked security features based on their importance and then built a tree-based generalized model for detecting intrusions. The algorithm was found to be accurate for unseen test cases and computationally efficient.

Various datasets are available for examining the effectiveness of machine learning algorithms for detecting IoT-related intrusions [19, 25, 26, 27]. Koroniotis et al., in [19] developed one such dataset for the purposes of training and validating system credibility. The authors created a

dataset called Bot-IoT, which integrated both simulated and legitimate internet of things traffic, including the different types of attacks. The study also encompassed presenting a test-bed environment for examining the current drawbacks of datasets, including capturing complete information about the network, accurate labeling, and emerging complex attacks. An evaluation of the BoT-IoT dataset using diverse machine learning and statistical methods compared with other datasets established adequate reliability. Ullah and Mahmoud in [26] exploited a Botnet dataset from an existing one for detecting anomalous activity in IoT networks. The dataset had broader network and flow-based features, which were tested using diverse machine learning approaches, such as feature correlation, and recursive feature elimination. In addition to possessing the required accuracy levels, the dataset offers a good ground for analyzing anomalous activity detection models for IoT systems. Sharafaldin et al., [27] also worked with an intrusion detection dataset. According to the researchers, most of the current datasets lack reliability, especially in the face of emerging threats. After evaluating their dataset, the authors found that it exhibits reliability and accuracy when used together with machine learning algorithms to detect diverse attack categories. However, this study did not focus on IoT-based systems, which is a key weakness as IoT networks tend to face unique attack threats. Nevertheless, the public availability of such datasets is imperative as it supports the creation and evaluation of IoT malicious detection models.

For better detection of malicious intrusions, a more effective method is needed, which is the focus of this research. The idea is to adopt the supervised feature selection approach by combining two or more machine learning methods. As a result, selecting the right features to improve the accuracy for IoT attack detection.

3. Proposed Methodology

In the proposed work, a model is proposed for the classification of BoT-IoT dataset. The data is shortened and several preprocessing steps are applied over it including unnecessary information removal, null value checking, label encoding, standard scaling and data balancing. To select the most relevant features and maintain model optimization, GA is applied over the preprocessed data. The selected features from GA are given to ML classifiers LR and SVM and the results are evaluated using evaluation measures. The overall model diagram is shown in Fig. 2.

3.1 Data Preparation

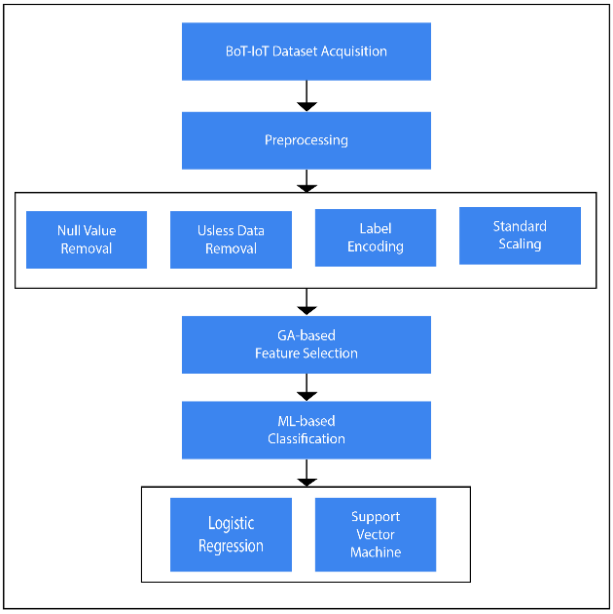

Fig. 2. Proposed Model Flow Diagram

The data used in this work is BoT-IoT dataset that is designed by Cyber Range Lab of UNSW Canberra and is publicly available[1]. The original BoT-IoT dataset contains more than 7 million records and occupies an enormous storage size [19]. It contains attacks performed on networks and computer systems over the years with respect to their categories mainly Distributed Denial of Service attack (DDoS), Denial of service attack (DoS), keylogging etc. The original dataset contains 19 columns with different features of which 7 features are selected for this work as shown in Table 1 including "min, state_number, mean, N_IN_Conn_P_DstIP, max, saddr_enc and daddr_enc. Data is loaded into the console where its head and columns are visualized.

[1]https://research.unsw.edu.au/projects/bot-iot-dataset

TABLE 1: DATASET FEATURES INVOLVED IN THE RESEARCH WORK

Sr. No	Features	Description
1	min	Minimum duration of aggregated records
2	state_number	Numerical representation of feature state
3	mean	Average duration of aggregated records
4	N_IN_Conn_P_DstIP	Number of inbound connections per destination IP
5	max	Maximum duration of aggregated records
6	proto	Textual representation of transaction protocols present in network flow
7	saddr	Source IP address
8	sport	Source port number
9	category	Traffic category
10	daddr	Destination IP address
11	dport	Destination port number

3.2 Preprocessing & Data Balancing

Algorithms learn insights directly from the data and the learning outcome needed to solve a particular problem is largely dependent on the nature of data. The data obtained from online sources is prepared by several sources, is mostly based on real-world based statistics or records and there is always a massive user input involved in it somehow. Due to these factors, dataset can contain redundant, unnecessary and complex information that may slow the model down in the later stages. Therefore, data preprocessing is applied as an essential initial step before any ML model can be used as it cleans the data in various ways and make it ready for the model to operate on it.

In this work, the shape of data is modified, Null values are checked, and records containing useless data columns are removed. The data columns with data types "object" are located [proto, saddr, sport, daddr, dport, category] and the frequency of their values is computed one after the another. The data shape is further wrapped. The data entries of complex hexadecimal values to simple real integers. The acquired dataset contains values and labels with various datatypes e.g., string, hexadecimal, integers and characters. Also, some values are exceptionally large and are separated by commas. This can cause an ambiguity for the employed ML or DL algorithms. Therefore, label encoding is applied to all the considered columns of both training and testing data partition. Label encoding is the process of labels into numeric format so that they may be read by machines. ML algorithms can then better decide how those labels should be used later on. In guided learning, it is a crucial pre-processing step for the structured dataset [20]. Encoding shortens the lengthy data patterns, limits the values into an appropriate range and maintains balance data types throughout the data columns. Furthermore, standard scaling is applied to both training and testing data partitions. Standard scaling converts data values such that their distribution has a mean value "0" and standard deviation value of "1". It maps the data values within the range of 0-1. It helps convert all the data columns into "float64" data type [21].

Both these preprocessing steps are employed to avoid data ambiguity, reduce its burden and complexity over the ML algorithms. The data columns are again visualized, and it is noticed that they are imbalanced and biased. This biasness in the training dataset can affect the outcomes generated by many ML algorithms, some of which completely ignore minority classes. This is a problem because prime numbers are usually more important for prediction. To deal with this, random over sampling is applied to the data. It involves randomly selecting examples from the minority class, with replacement, and adding them to the training dataset. It also involves randomly selecting examples from the majority class and deleting them from the training dataset. Fig. 3 Shows the overview of utilized preprocessing steps for the proposed model.

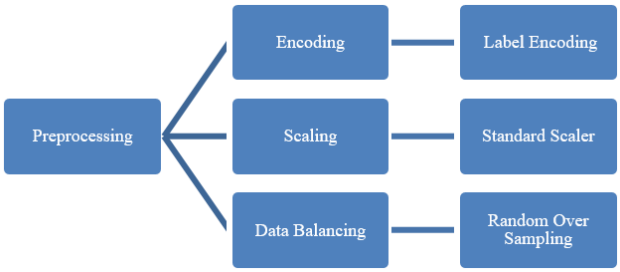

Fig. 3. Preprocessing steps overview

3.3 Feature selection

The process of creating a subset from an initial feature set according to a feature selection criterion, which picks the relevant characteristics of the dataset, is referred to as feature selection. It aids in the compression of data processing scales by removing superfluous and extraneous characteristics. Learning algorithms may be pre-processed using feature selection techniques, and effective feature selection results can enhance accuracy, lessen processing time and simplify learning outcomes [22].

3.4 Genetic Algorithm

In the proposed work feature selection is performed through the implementation of GA which is a heuristic algorithm and works based on five general steps including declaration of initial agent population, formulation of a compact fitness function, selection of fittest agents from the pool of overall population, allocation of a crossover points and finally mutation as shown in Fig. 4. It is a multi-objective feature selection approach based on population of genes and natural selection principles. As described in Eq. (1), it begins by picking an initial set of candidate solution locations known as chromosomes cr at random from an array of problem parameters pb that operate as first generation.

$$cr = [pb_1, pb_2, pb_3, \ldots\ldots] \quad (1)$$

Every input issue has several chromosomes, which are collectively referred to as population. Each iteration maintains the size of the population's solution spots. At each iteration, the fitness function in the population is calculated, and the next chromosomes for reproduction are chosen based on a specified prospect distribution. The probability of a chromosome cr in a certain population p being chosen from a N pool of problem parameters pb according to the fitness function g are represented in Eq (2).

$$P(cr_i) = \left| \frac{g(cr_i)}{\sum_{p=1}^{N} g(cr_p)} \right| \quad (2)$$

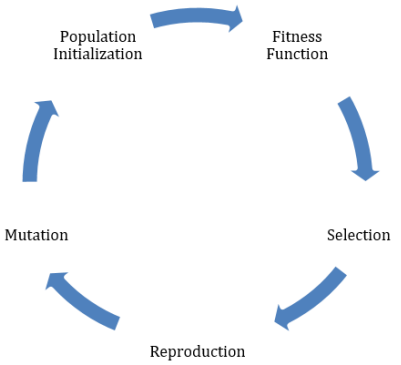

Fig. 4. Standard GA feature selection cycle

In this work, the initial population is maintained as 20 agents, the sole purpose of which is to find the best solution which contain genes embedded in chromosomes. For the next step of finding out which individuals are fit for solution derivation, a fitness function is formulated which calculates the sum of solution samples among each input variable as per its weight. At the end of this phase, each sample is given a fitness score which acts as a root entity for the selection of those samples for the upcoming steps. The next phase is to select fittest individuals for the increment of search agents which are then paired in pairs of two. The individuals having high fitness score have a better chance of being selected for the reproduction phase. The GA-based feature selection in this work is completed in 6 generations. The selected individuals in the pairs of two parents are then directed to reproduction phase starting with crossover. In this phase, 5 parents are allocated for mating where a new individual is formed which contains half of its genes from first parent and other half from second parent. In this way, the newly formed agents contain the properties from multiple parents and thus assist better in finding solution. The ultimate step includes the mutation of newly formed individuals in which some of their genetic information is tweaked to eliminate the problem of local minima and amplify the working of algorithm. This feature selection phase delivers a fresh pool of newly formed population which is then passed on to fitness function together with input data and labels to compute accuracy. This step finds out the best performing solution based on its index corresponding to its fitness and finds its accuracy. This best solution is amalgamated with newly formed population obtained from the reproduction phase to select best features. These selected features are then passed on to the ML classifiers - LR and SVM to evaluate the performance of proposed model.

3.5 Classification

Classification is a supervised learning concept in ML that categorizes a set of data into classes. It might be a multi-class problem or a binary classification task [22]. In ML, there are a variety of classification algorithms. The proposed work makes use of SVM and LR and results are taken. The accuracy of these models can be realized once they are trained on real dataset and evaluated using several performance measures such as recall, precision and f1-score.

4. Experimentation and Results

A model is proposed in this work to classify a partition of BoT-IoT dataset's attributes based on their labels. The BoT-IoT dataset is obtained from its respective online source which is passed through several preprocessing steps that include data creation, null value removal, label encoding, standard scaling and data balancing. Feature selection is achieved through GA. Finally, the selected features are classified using ML classifiers - LR and SVM and evaluated using Performance Evaluation Measures (PEMs). To improve accuracy, hyperparameter tuning is applied on ML algorithms and the results achieved before and after this process are noted and compared. Two experiments are performed in total. The first experiment involves the classification results achieved by the ML models without parameters tuning. The second experiment involves the classification results of ML models after parameters are tuned. K fold and cross validation methods are applied in hyperparameter tuning and the means of scores are computed.

TABLE 2: ESULTS ACHIEVED BY ML CLASSIFIERS BEFORE HYPERPARAMETER TUNING

ML Classifier	Accuracy	1 score	Precision	Recall
LR	0.95	0.95	0.95	0.95
SVM	0.96	0.96	0.96	0.96

As shown in Table 2, the SVM classifier achieves slightly better accuracy of 96% as well as achieves better performance measure rates on standard model as compared to LR with 95% accuracy rate without hyper-parameter tuning. The same results can be seen in Fig. 5.

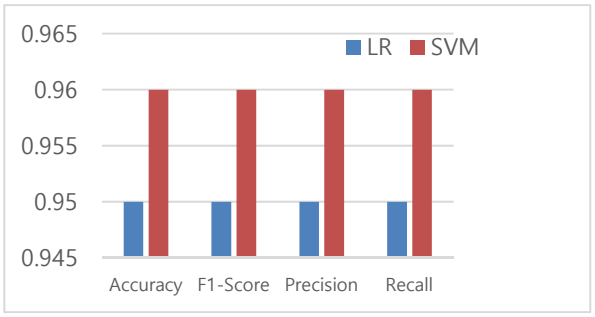

Fig. 5. Comparison of ML classifiers before hyperparameter tuning using PEMs

In the next experiment, the same sequence of experiments is repeated and the results are shown in Table 3.

TABLE 3: RESULTS ACHIEVED BY ML CLASSIFIERS AFTER HYPERPARAMETER TUNING

ML Classifier	Accuracy	1 score	Precision	Recall
LR	0.96	0.96	0.96	0.96
SVM	0.97	0.97	0.97	0.97

As shown in Table 3, the performance of SVM classifier inclines after hyperparameters are tuned in the proposed model. SVM now provides an accuracy of 97% which is greater than that of LR with 96% accuracy. The same results are evident from Fig. 6.

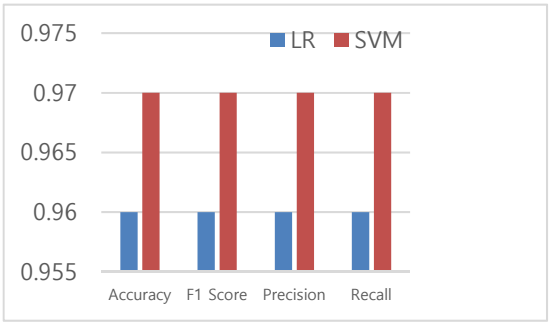

Fig. 6. Comparison of ML classifiers before hyperparameter tuning using PEMs

5. Conclusion

A model is proposed for the classification of BoT-IoT data features. Preprocessing of data is performed with steps such as label encoding, unnecessary information removal, scaling and data balancing. GA is used to derive the most suitable feature attributes. Finally, two ML classifiers LR and SVM are used to classify selected features. In a set of two of the conducted experiments, it can be concluded that hyperparameter tuning has a significant effect on the performance of both ML classifiers. SVM still remains the best model in both cases and overall results increased as also shown in Fig. 7 with visual details for all PEMs.

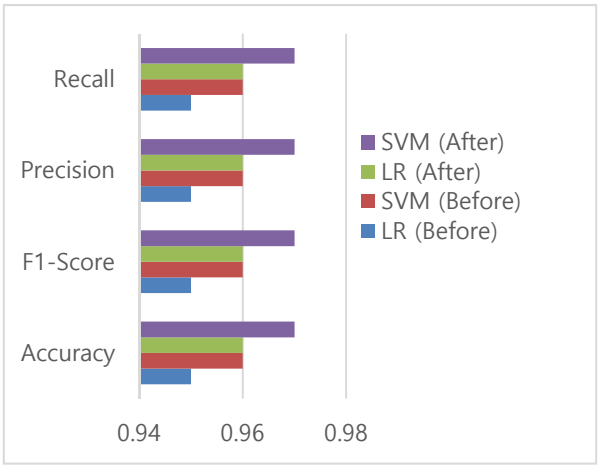

Fig. 7. Comparison of ML classifiers before and after hyperparameter tuning using PEMs

Acknowledgment

The authors would like to thank Arab Open University, Saudi Arabia for supporting this study. Dr. Saman Iftikhar is the corresponding author.

References

[1] R. M. Alhajri, A. B. Faisal and R. Zagrouba. "Survey for anomaly detection of IoT botnets using machine learning auto-encoders," Int J Appl Eng Res, vol. 14, no. 10, pp. 2417, 2019.

[2] 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, 2015.

[3] M. Abdullah, A. Balamash, A. Al-Shannaq, and S. Almabdy. (2018). Enhanced Intrusion Detection System using Feature Selection Method and Ensemble Learning Algorithms. International Journal of Computer Science and Information Security. 16. 48-55.

[4] S. Egea, A. R. Mañez, B. Carro, A. Sánchez-Esguevillas, and J. Lloret, "Intelligent IoT traffic classification using novel search strategy for fast based-correlation feature selection in industrial environments," IEEE Internet of Things Journal, vol. 5, no. 3, pp. 1616–1624, 2018.

[5] H. Zhang, G. Lu, M. T. Qassrawi, Y. Zhang, and X. Yu, "Feature selection for optimizing traffic classification," Computer Communications, vol. 35, no. 12, pp. 1457–1471, 2012.

[6] S. Su, et al. "A correlation-change based feature selection method for IoT equipment anomaly detection," Applied Sciences, vol. 9, no. 3, pp. 437, 2019.

[7] W. Alhakami, et al. "Network anomaly intrusion detection using a nonparametric bayesian approach and feature selection," IEEE Access, vol. 7, pp. 52181-5219, 2019.

[8] M. Shafiq et al. "Effective feature selection for 5G IM applications

traffic classification," Mobile Information Systems, 2017.

[9] A. Saxena, S. Sinha, P. Shukla, "General study of intrusion detection system and survey of agent based intrusion detection system," Proc. 2017 Int. Conf. Comput. Commun. Autom., pp. 421–471, 2017.

[10] L. H. and M. A. Jabbar, "Role of machine learning in intrusion detection system: Review," Proc. 2018 Second Int. Conf. Electron. Commun. Aerosp. Technol., pp. 925–929, 2018.

[11] Y. Y. A. and M. M. Min, "An analysis of random forest algorithm based network intrusion detection system," Proc. 2017 18th IEEE/ACIS Int. Conf. Softw. Eng. Artif. Intell. Netw. Parallel/Distributed Comput., pp. 127–132, 2017.

[12] T. B. Alhijaj, S. M. Hameed, and B. A. Attea, "A Decision Tree-Aware Genetic Algorithm for Botnet Detection," vol. 62, no. 7, pp. 2454–2462, 2021.

[13] Z. Liu and Y. Shi, "A Hybrid IDS Using GA - Based Feature Selection Method and Random Forest," vol. 12, no. 2, 2022.

[14] J. Yin, C., Awlla, A. H., Yin, Z., & Wang, "Botnet detection based on genetic neural network," Int. J. Secur. Its Appl., vol. 9(11), pp. 97–104, 2015.

[15] J. Moustafa, N., & Slay, "The evaluation of Network Anomaly Detection Systems: Statistical analysis of the UNSW-NB15 data set and the comparison with the KDD99 data set," Inf. Secur. J. A Glob. Perspect., vol. 25(1–3), pp. 18–31, 2016.

[16] E. A. Alejandre, F. V., Cortés, N. C., & Anaya, "Feature selection to detect botnets using machine learning algorithms," Int. Conf. Electron. Commun. Comput., 2017.

[17] M. A. Alauthaman, M., Aslam, N., Zhang, L., Alasem, R., & Hossain, "A P2P Botnet detection scheme based on decision tree and adaptive multilayer neural networks.," Neural Comput. Appl., vol. 29(11), pp. 991–1004, 2017.

[18] Y. Chang, W. LLi,and Z. Yang, "Network intrusion detection based on random forest and support vector machine," Proc. 2017 IEEE Int. Conf. Comput. Sci. Eng. IEEE Int. Conf. Embed. Ubiquitous Comput., pp. 635–638, 2017.

[19] N. Koroniotis, N. Moustafa, E. Sitnikova, B. Turnbull, "Towards the Development of Realistic Botnet Dataset in the Internet of Things for Network Forensic Analytics: Bot-IoT Dataset", https://arxiv.org/abs/1811.00701, 2018.

[20] B. B. Jia and M. L. Zhang, 2021. Multi-Dimensional Classification via Decomposed Label Encoding. IEEE Transactions on Knowledge and Data Engineering.

[21] M. M. Ahsan, M. A. Mahmud, P. K. Saha, K. D. Gupta and Z. Siddique, 2021. Effect of data scaling methods on machine learning algorithms and model performance. Technologies, 9(3), p.52.

[22] J. Cai, J. Luo, S. Wang and S. Yang, 2018. Feature selection in machine learning: A new perspective. Neurocomputing, 300, pp.70-79.

[23] S. B. Kotsiantis, I. D. Zaharakis and P. E. Pintelas, 2006. Machine learning: a review of classification and combining techniques. Artificial Intelligence Review, 26(3), pp.159-190.

[24] I. H. Sarker et al. "Intrudtree: A machine learning based cyber security intrusion detection model," Symmetry vol. 12, no. 5, pp. 754, 2020.

[25] R. Ahmad and I. Alsmadi, I. "Machine learning approaches to IoT security: A systematic literature review," Internet of Things, 100365, 2021.

[26] I. Ullah, and Q. H. Mahmoud, "A Technique for Generating a Botnet Dataset for Anomalous Activity Detection in IoT Networks," 2020 IEEE International Conference on Systems, Man, and Cybernetics (SMC) (pp. 134-140, 2020.

[27] I. Sharafaldin et al., "Toward generating a new intrusion detection dataset and intrusion traffic characterization," ICISSp, pp. 108-116, 2018.

[28] S. Liu, X. Hao, and X. Chen. A semi-supervised dynamic ensemble algorithm for IoT anomaly detection. 2020 International Conferences on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData) and IEEE Congress on Cybermatics (Cybermatics), 264-269.