ICIC Express Letters
Volume **18**, Number **6**, June **2024**

ICIC International ©2024 ISSN 1881-803X
pp. **597–606**

EMPLOYING MUTUAL INFORMATION FEATURE SELECTION AND LIGHTGBM FOR INTRUSION DETECTION IN IOT

Salam Al-E'mari[1,*], Yousef Sanjalawe[2], Duha Alsmadi[3], Eman Alduweib[3] and Alyaa Alharbi[4]

[1]Information Security Department
[3]Computer Science Department
Faculty of Information Technology
University of Petra
P.O. Box 961343, Amman 11196, Jordan
{ duha.alsmadi; eman.alduweib }@uop.edu.jo
*Corresponding author: salam.ammari@uop.edu.jo

[2]Cybersecurity Department
School of Information Technology
American University of Madaba
P.O. Box 2882, Amman 11821, Jordan
y.sanjalawe@aum.edu.jo

[4]College of Computer and Information Systems
Umm al-Qura University
P.O. Box 715, Makkah 21421, Saudi Arabia
aaralharbi@uqu.edu.sa

Received July 2023; accepted October 2023

Abstract. *The advent of the Internet of Things (IoT) has significantly altered human interaction with the environment, enabling smart ecosystems that simplify day-to-day activities. Despite these benefits, the expansion of IoT technology has also escalated security and privacy vulnerabilities, necessitating robust network protection mechanisms. This paper introduces an innovative approach that combines Mutual Information Feature Selection (MIFS) with the Flower Pollination Algorithm (FPA) and Particle Swarm Optimization (PSO) for effective feature selection. Furthermore, the detection task is performed using the Light Gradient-Boosting Machine (LightGBM). However, the empirical tests on the IoTID20 dataset reveal that the proposed methodology surpasses various state-of-the-art intrusion detection techniques in accuracy, recall, and F1-score. Moreover, the approach exhibits lower false positive rates and higher detection rates, affirming its efficacy in identifying IoT network intrusions.*

Keywords: Bio-inspired algorithms, Internet of Things, Intrusion detection, Feature selection, PSO, FPA, LightGBM

1. **Introduction.** The rapid and extensive growth of Internet of Things (IoT) technology has revolutionized the way humans interact with their environment, giving rise to smart cities, homes, and industries that seamlessly integrate devices and communication protocols. Although IoT offers numerous benefits, its escalating interconnectivity also presents significant security challenges. These systems are increasingly vulnerable to cyber-attacks, making the security and privacy of IoT networks crucial for their successful deployment and widespread acceptance [1]. Intrusion Detection Systems (IDSs) play a pivotal role in identifying and countering potential threats, thereby preserving the integrity and resilience of IoT networks [2, 3, 4]. In light of the unique characteristics of IoT networks, such as the sheer volume of devices, the heterogeneity of communication protocols, and

DOI: 10.24507/icicel.18.06.597

the vast amount of data generated, the development of efficient and effective IDS tailored to IoT networks has become a pressing research area [5]. In addition, a critical aspect of designing a robust IDS is the selection of relevant features from the high-dimensional datasets typical of IoT systems. Where the efficacy of feature selection techniques can profoundly influence the performance of the classification model. By reducing dataset complexity and dimensionality, these techniques enable the classifier to more accurately distinguish benign from malicious activities [6, 7].

As security threats to IoT networks continue to evolve, researchers and security analysts have developed various IDSs, each employing different strategies for feature optimization and enhanced security. In [8], the authors propose a robust IDS framework for IoT environments, employing deep-learning models to classify intrusions. The IoTID20 dataset is utilized, and PSO is used to select relevant features to improve the system's performance. Furthermore, it uses three deep-learning algorithms for classification tasks: Convolutional Neural Networks (CNN), Long Short-Term Memory (LSTM) networks, and a hybrid CNN-LSTM model. Experimental results reveal high accuracy levels in binary classification, with LSTM leading at 99.82%, followed by the hybrid CNN-LSTM model at 98.80%, and CNN at 96.60%. Moreover, the PSO algorithm was applied to the NSL-KDD dataset for feature selection, reducing the feature set from 41 to 10. This led to improved detection rates and accuracy while lowering false alarms. The performance was evaluated using various classifiers like k-NN, SVM, LR, DT, and Naïve Bayes [9]. In addition, three deep learning algorithms are employed for classification: Convolutional Neural Networks (CNN), Long Short-Term Memory (LSTM) networks, and a hybrid CNN-LSTM model. Experimental results demonstrate high levels of accuracy in binary classification. LSTM outperforms the others with an accuracy rate of 99.82%, followed by the CNN-LSTM hybrid model at 98.80%, and CNN at 96.60%. Alternatively, another study proposed a hybrid model that combines the Cuttlefish Optimization Algorithm (CFA) with a Decision Tree (DT) classifier for detecting network breaches. In this model, CFA is used for feature selection, while DT is employed for classification of anomalous events. The model's effectiveness was evaluated using the KDDCup99 dataset. Results showed a significant increase in detection rate and accuracy when the number of features was reduced to fewer than 20 [10].

On the other hand, [11] offers an in-depth review of various models used for intrusion detection in IoT systems. It highlights deep learning as a promising approach but acknowledges existing challenges. The paper calls for further research, especially in distributed, efficient, and unsupervised deep learning methods, to address issues like scalability, resource constraints, and data limitations in large IoT networks. Moreover, [12] introduces a novel botnet dataset specifically designed for intrusion detection in IoT networks. This dataset includes both flow-based and statistical features, making it valuable for developing and evaluating flow-based IDSs in IoT settings. The dataset also provides high-ranking features suitable for malicious activity prediction models. To create three subsets from this dataset, a recursive feature elimination algorithm was used. The dataset's credibility was then assessed using various metrics, including accuracy, precision, recall, and F-score.

This paper introduces an innovative approach to intrusion detection in IoT systems. It combines an enhanced mutual information feature selection method – enriched with the FPA and PSO – with the robust classification capabilities of the Light Gradient-Boosting Machine (LightGBM). This proposed methodology seeks to effectively pare down the feature space's complexity and amplify the intrusion detection model's efficacy. The integration of nature-inspired optimization algorithms with MIFS facilitates a more refined feature selection, while the LightGBM classifier promises high precision and efficiency, especially with expansive IoT datasets. The effectiveness of this approach is validated through performance evaluations on the IoTID20 dataset. Preliminary results showcase

the superiority of our method in terms of accuracy, recall, precision, and F1-score, underscoring its potential to bolster IoT security by swiftly and accurately detecting intrusions and thwarting potential threats.

The remainder of this paper is structured as follows. Section 2 details the proposed approach, elaborating on the enhanced feature selection procedure and the LightGBM-based classification. Section 3 outlines the experimental framework, dataset, and evaluation metrics used to assess our method's performance. Finally, Section 4 concludes the paper, highlighting potential avenues for future research.

2. **Methodology.**

A robust intrusion detection system designed specifically for IoT networks is detailed in this paper. The proposed approach aims to tackle the challenges of high-dimensional feature spaces and improve the classifier's performance in detecting malicious activities within IoT environments. The methodology comprises several critical steps, including data preprocessing, feature selection using FPA and PSO for identifying mutual features, and classification through LightGBM, as depicted in Figure 1.

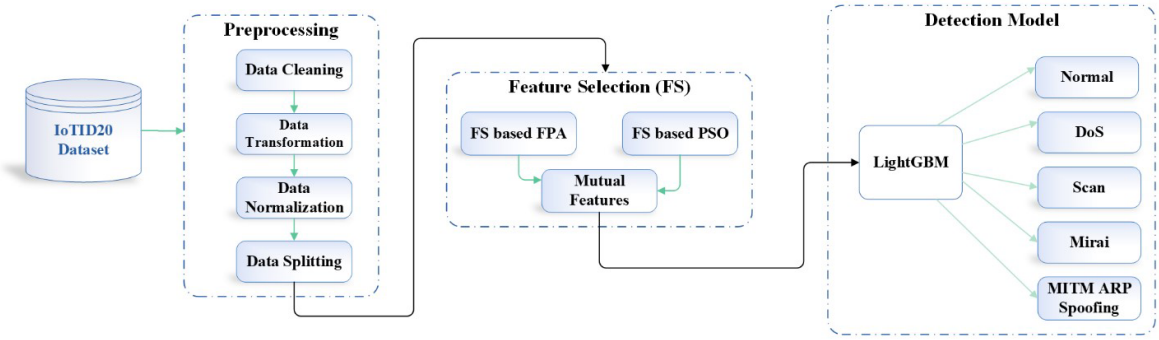


FIGURE 1. The framework of the proposed approach

2.1. **Dataset description and preprocessing.**

The proposed approach was rigorously evaluated using the IoTID20 dataset, which was created in 2020 and obtained from an online source. Derived from PCAP (Packet Capture) files, this dataset features 82 distinct attributes that help distinguish between normal and anomalous network traffic. The IoTID20 dataset contains 625,783 records and includes several categories such as Scan, Man-In-The-Middle (MITM), Denial of Service (DoS), Mirai attacks, and normal traffic, all of which were used for classification purposes [8, 13]. Figure 2 provides a detailed visualization that clearly illustrates the distribution of these categories within the dataset, highlighting the proportion of each type of network activity. However, the dataset requires initial preprocessing, detailed in the subsequent section. The aim of this phase is to optimize the dataset for analysis by removing biases, cleaning the data, and transforming features, which can significantly impact the performance of feature selection and classification algorithms [14]. The preprocessing steps for this dataset are based on the methodology outlined in [15] and include

- Data Cleaning: To ensure consistency and prevent calculation errors during analysis, NaN (Not a Number) and INF (Infinity) values have been removed. Additionally, features such as Timestamp, FlowID, SrcIP, and DstIP have been excluded to enhance the model's learning efficiency.
- Data Transformation: Categorical features, specifically Src Port, Dst Port, and Protocol have been converted into one-hot encoded vectors.
- Data Normalization: A min-max scaling technique has been employed to standardize the range of numerical features, thus enhancing the model's overall performance.

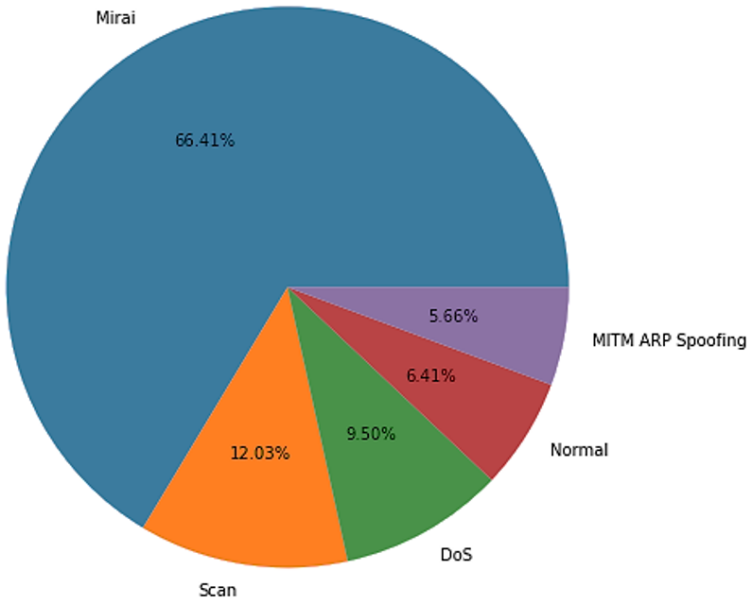


FIGURE 2. Distribution of instances across categories in the IoTID20 dataset

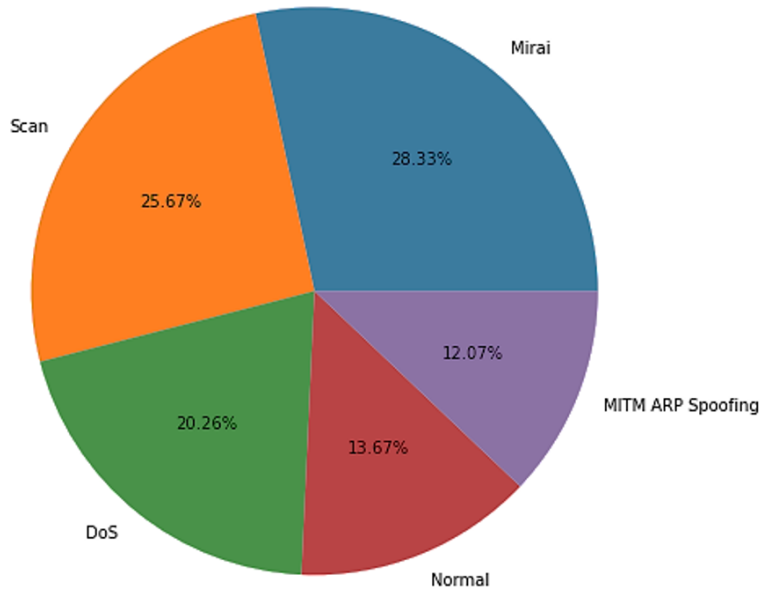


FIGURE 3. IoTID20 dataset distribution after overcoming addressing class imbalance

- Data splitting: The dataset is split into 80% and 20% for the training and test sets, respectively, where the training is set to facilitate model training and the test set for evaluating the model's performance.

The dataset contains 77 numerical features. However, as shown in Figure 2, the dataset is imbalanced. To address this issue, the proposed approach employs an under-sampling technique to balance the dataset, as illustrated in Figure 3.

However, the dataset is now ready for the next phase (Feature Selection) which will be discussed in the following section.

2.2. **Mutual information feature selection.** The feature selection phase is carried out using FPA and PSO, which are nature-inspired optimization algorithms. These algorithms search the feature space to identify the most relevant features that aid in effectively classifying benign and malicious activities. Moreover, reducing the feature set significantly decreases the computational complexity of the subsequent classification phase, resulting

in a more efficient intrusion detection system [16]. In addition, mutual features – those deemed significant by both algorithms – are identified [17]. By focusing on these mutual features, the methodology capitalizes on the strengths of both FPA and PSO, ensuring that the selected features are highly relevant for accurate activity classification within the IoT network.

For the fitness function used in this paper, an additional parameter called 'error' is introduced. This parameter accounts for the weights of the attributes and aids in minimizing the error rate, as depicted in Equation (1). Here, the alpha value is set at 0.99, while the beta value is calculated as $1-\alpha$. The cost function is further defined by Equation (2) [18].

$$error = error_{rate}(X_{train}, Y_{train}) \tag{1}$$

$$Cost = \alpha * error + \beta * \frac{Number\ Features}{Max\ Features} \tag{2}$$

2.2.1. *Particle swarm optimization.* PSO is a population-based optimization algorithm that uses a swarm of particles. Each particle represents a potential solution and is characterized by its position, which indicates its quality, and its velocity, which denotes the speed and direction of its movement in subsequent iterations. A particle's position and velocity are expressed as two-dimensional vectors, the size of which corresponds to the problem's dimensionality. Furthermore, a fitness function evaluates each particle's position, and the best position (*pbest*) that each particle has discovered so far is recorded and shared among the swarm. In the standard PSO, a fully connected topology is employed to identify the global best position (*gbest*) for the entire population. In contrast, some typologies substitute *gbest* with a local best *lbest*. These best positions are then used to adjust each particle's velocity, which in turn determines its position based on Equation (3) and Equation (4) [19, 20, 21], where v_{id}^t and x_{id}^t denote the velocity and position of the ith particle in dimension d at time t, respectively. The inertia weight (w) refers to the particles' moving momentum. At time t, p_{id}^t represents the *pbest* and *gbest* positions in dimension d, respectively. Acceleration constants are represented by c_1 and c_2, while r_{1i} and r_{2i} are random values uniformly distributed between 0 and 1 [19, 20].

$$v_{id}^{t+1} = w * v_{id}^t + c_1 * r_{1i} * \left(p_{id}^t - x_{id}^t\right) + c_2 * r_{2i} * \left(p_{gd}^t - x_{id}^t\right) \tag{3}$$

$$x_{id}^{t+1} = x_{id}^t + v_{id}^{t+1} \tag{4}$$

2.2.2. *Flower pollination algorithm.* FPA is a nature-inspired algorithm introduced by Yang in 2012 [27]. It is fundamentally based on the pollination process observed in flowering plants and operationalizes the mechanisms of both global (biotic) and local (abiotic) pollination, ensuring effective exploration of the solution space. In its global pollination phase, FPA generates a population of solutions, represented as flowers, each containing a unique candidate solution known as pollen. These pollens are shared among the flowers through a mechanism called Levy flight, which is inspired by the long-distance flights undertaken by certain pollinators. This process allows the algorithm to make significant leaps across the solution space, thereby promoting exploitation. Conversely, in its local pollination phase, FPA fine-tunes solutions based on differences between them and their neighboring solutions. This mechanism, inspired by the natural dispersal of pollen by wind, enables the algorithm to explore the solution space more thoroughly, refining solutions that are close to known optimal ones [22]. Moreover, there are four rules that apply in FPA as follows [22, 23]:

1) Biotic and cross-pollination are components of global pollination, carried out by pollinators following a Levy flight distribution,
2) Abiotic and self-pollination mechanisms trigger local pollination,
3) The likelihood of reproduction between two flowers is influenced by their level of similarity, also referred to as flower constancy,

4) A switch probability, ranging between 0 and 1, governs the selection between local and global pollination.

2.3. **Light gradient-boosting machine.** In 2017, Microsoft introduced the LightGBM, a model that has gained prominence for its reduced computational time compared to Extreme Gradient Boosting (XGBoost). Due to its speed, efficiency, and accuracy, LightGBM has quickly become popular in the machine-learning community, offering several advantages over traditional gradient-boosting methods. As a gradient-boosting framework, LightGBM employs tree-based learning algorithms and works on the fundamental principle of gradient boosting. This involves the incremental and sequential construction of a model while optimizing a differentiable loss function. LightGBM sets itself apart by incorporating innovative techniques such as Gradient-based One-Side Sampling (GOSS) and Exclusive Feature Bundling (EFB). These strategies significantly accelerate the model training process and optimize memory usage, respectively [24, 25].

3. **Results and Discussion.** In this section, the findings are presented and the experimental results of the proposed approach are analyzed. To ensure the consistency and reliability of these results, all experiments were conducted on a dedicated system. The specifications for this computational environment are detailed in Table 1.

Table 1. System specifications for experimental setup

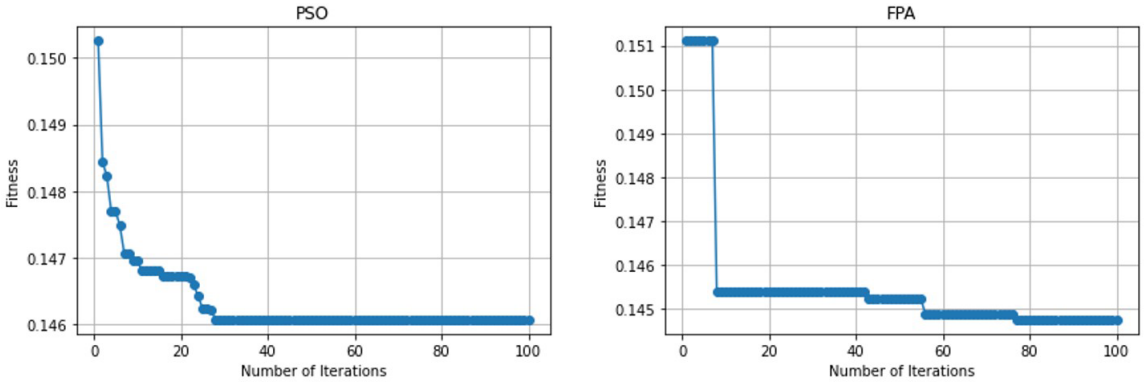
Component	Specification
Operating system	Windows Server 2016 Datacenter
System architecture	64-bit
Memory (RAM)	64 GB
Processor	Intel(R) Xeon(R) Silver 4314 CPU @ 2.4 GHz
Number of processors	12
Programming language	Python 3.11

On the other hand, fine-tuning the hyperparameters of both PSO and FPA is essential for optimizing performance in the specific task at hand, aiming to strike a balance between exploitation and exploration. The meticulous tuning of these hyperparameters can have a significant impact on the overall efficacy of the algorithms. Table 2 details the specific hyperparameters for PSO and FPA used in the proposed approach.

Table 2. PSO and FPA hyperparameters

Algorithm	Parameter	Variable	Value
PSO + FPA	Fitness function	FF	Equation (2)
	Max iteration	i	100
	Population size	N	10
PSO	Cognitive component	c_1	1.5
	Social component	c_2	1.5
	Random values	r_1, r_2	$[0, 1]$
	Inertia weight	w	0.9
	Dimension	dim	77
FPA	Levy component	B	1.5
	Switch probability	P	0.8

After optimizing the hyperparameters for both PSO and FPA, significant results emerged. For PSO, the algorithm selected a set of 21 features from the initial dataset, enhancing the model's performance and emphasizing the effectiveness of PSO in identifying crucial features while eliminating redundant ones. With a fitness value of 0.14606670, PSO



(a) (b)

FIGURE 4. Fitness function results for (a) PSO and (b) FPA

proved competent for this specific task. In contrast, FPA selected a larger set than PSO of 34 features after its parameter optimization. Although the computed fitness value for FPA was 0.14474929, suggesting a good solution, it was slightly less optimized compared to PSO, given that a lower fitness value indicates a more optimal solution. Moreover, Figure 4 plots the evolution of fitness function values over a series of iterations for both PSO and FPA. Each point on the curve reflects the fitness value of the best solution for that respective iteration. The curves for both algorithms show a trend of decreasing fitness values over iterations, signaling a continuous improvement in solution quality.

The contrast in feature selection between the two algorithms and their respective fitness values highlights the influence of their distinct search strategies. Further, it indicates that while both algorithms effectively reduce the problem's dimensionality, they may not always converge to the same solution. Table 3 presents the feature selection results for each algorithm, sorted by index, along with the results from the MIFS method.

TABLE 3. Overview of feature selection experiments

Algorithm	Best FF	Feature size	Feature selection
PSO	0.14606670	21	[0 5 6 16 24 26 31 38 39 47 50 52 53 56 58 62 63 65 66 68 71]
FPA	0.14474929	34	[1 2 5 8 9 10 16 17 19 21 24 28 32 33 34 37 38 44 47 48 49 50 53 54 55 57 58 59 62 65 66 69 71 74]
MIFS	–	11	[5 16 24 38 47 50 53 58 62 66 71]

During the feature selection experiments, the MIFS method identified 11 features that significantly influenced the performance of the LightGBM model. The descriptions of these features are provided in Table 4.

Due to the nature of IoT networks, which consist of heterogeneous devices with limited resources, a lightweight classification algorithm is highly desirable for an efficient security system. Accordingly, the LightGBM model is considered an optimal choice for ensuring excellent performance in IoT network security systems [26]. This claim is further substantiated by the results shown in Table 5, which indicate high detection accuracy, especially after the feature selection process.

With respect to the model trained on a feature set of 77 features, the recorded accuracy, recall, precision, and F1-score range between 86% and 88%. Although this set contains a large number of features, these performance metrics fall short when compared to models trained with fewer features. This discrepancy may arise from noise or redundancy in

Table 4. Feature selection description

ID	Feature	Description
5	Protocol	The type of protocol used for data transmission.
16	Bwd_Pkt_Len_Max	The maximum size of the packet transferred backward during a transmission.
24	Flow_IAT_Max	The maximum time interval between two consecutive packets transmitted within the flow.
38	Fwd_URG_Flags	Flag that marks the urgent data needing to be sent forward.
47	Pkt_Len_Std	The standard deviation of packet length.
50	SYN_Flag_Cnt	The number of synchronizations in a packet.
53	ACK_Flag_Cnt	The number of acknowledgment flags in a packet.
58	Pkt_Size_Avg	The average size of packets transferred during a session.
62	Fwd_Pkts/b_Avg	The average size of forwarding packets per bulk.
66	Bwd_Blk_Rate_Avg	The average rate of data blocks transferred backward.
71	Init_Fwd_Win_Byts	The initial size of the forward window byte.

Table 5. LightGBM results

Feature size	Accuracy	Recall	Precision	F1-score
77	87%	88%	86%	87%
21	89%	90%	88%	89%
34	91%	92%	90%	91%
11	98%	98%	98%	98%

the extensive feature set, potentially undermining the model's performance. Conversely, when the feature set is reduced to 21 or 34 elements, there is a notable enhancement in all performance metrics. This improvement indicates that a condensed feature set allows the model to focus on the most influential data, thereby boosting its predictive accuracy. Intriguingly, the model trained with the smallest feature set, containing just 11 elements, achieves the highest performance in all metrics, consistently scoring 98%. This result indicates that these 11 features are likely the most effective in terms of predictive capability.

4. **Conclusions.** This paper enhances the field of IoT security by both introducing robust intrusion detection and leveraging machine learning and nature-inspired optimization algorithms. The proposed approach employed MIFS augmented by the FPA and PSO for feature selection and used LightGBM for classification. However, the empirical validation affected based on the IoTID20 dataset effectively manifested the proficiency of the proposed methodology, attaining preeminent performance in a variety of evaluation metrics, namely accuracy, recall, and F1-score. Interestingly, the model incorporating the most reduced feature set, comprised of merely 11 features, reported the most commendable performance across all evaluation metrics. This outcome significantly emphasizes the criticality of efficient feature selection strategies in augmenting the predictive accuracy of machine learning models. While these results are encouraging, future work could focus on further refining the feature selection and classification models, testing the methodology on different IoT datasets, and comparing the approach with other machine learning and optimization techniques.

Acknowledgment. The authors would like to thank the University of Petra (UoP) in Jordan for providing the invaluable platform that fostered our learning, exploration, and research.

REFERENCES

[1] Y. Li, Privacy protection algorithm for source node location based on phantom routing in the Internet of Things environment, *International Journal of Innovative Computing, Information and Control*, vol.17, no.3, pp.973-989, 2021.

[2] G. Kalogridis, M. Sooriyabandara, Z. Fan and M. A. Mustafa, Toward unified security and privacy protection for smart meter networks, *IEEE Systems Journal*, vol.8, no.2, pp.641-654, 2013.

[3] S. Al-E'mari, M. Anbar, Y. Sanjalawe, S. Manickam and I. Hasbullah, Intrusion detection systems using blockchain technology: A review, issues and challenges, *Computer Systems Science & Engineering*, vol.40, no.1, 2022.

[4] Y. Yuan, H. Dai, Z. Wu and D. Meng, Novel network intrusion detection method based on IPSO-MTWSVM model, *Engineering Letters*, vol.30, no.2, 2022.

[5] N. Koroniotis, N. Moustafa and E. Sitnikova, Forensics and deep learning mechanisms for botnets in Internet of Things: A survey of challenges and solutions, *IEEE Access*, vol.7, pp.61764-61785, 2019.

[6] M. Rostami, K. Berahmand, E. Nasiri and S. Forouzandeh, Review of swarm intelligence-based feature selection methods, *Engineering Applications of Artificial Intelligence*, vol.100, 104210, 2021.

[7] N. Shone, T. N. Ngoc, V. D. Phai and Q. Shi, A deep learning approach to network intrusion detection, *IEEE Transactions on Emerging Topics in Computational Intelligence*, vol.2, no.1, pp.41-50, 2018.

[8] H. Alkahtani and T. H. Aldhyani, Intrusion detection system to advance Internet of Things infrastructure-based deep learning algorithms, *Complexity*, vol.2021, pp.1-18, 2021.

[9] N. Kunhare, R. Tiwari and J. Dhar, Particle swarm optimization and feature selection for intrusion detection system, *Sādhanā*, vol.45, pp.1-14, 2020.

[10] A. S. Eesa, Z. Orman and A. M. A. Brifcani, A novel feature-selection approach based on the cuttlefish optimization algorithm for intrusion detection systems, *Expert Systems with Applications*, vol.42, no.5, pp.2670-2679, 2015.

[11] S. Tsimenidis, T. Lagkas and K. Rantos, Deep learning in IoT intrusion detection, *Journal of Network and Systems Management*, vol.30, pp.1-40, 2022.

[12] 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.

[13] A. Sarwar, S. Hasan, W. U. Khan, S. Ahmed and S. N. K. Marwat, Design of an advance intrusion detection system for IoT networks, *2022 2nd International Conference on Artificial Intelligence (ICAI)*, pp.46-51, 2022.

[14] S. Al-E'mari, M. Anbar, Y. Sanjalawe and S. Manickam, A labeled transactions-based dataset on the Ethereum network, *International Conference on Advances in Cyber Security*, pp.61-79, 2020.

[15] Y. Song, S. Hyun and Y.-G. Cheong, Analysis of autoencoders for network intrusion detection, *Sensors*, vol.21, no.13, 4294, 2021.

[16] H. Rao, X. Shi, A. K. Rodrigue, J. Feng, Y. Xia, M. Elhoseny, X. Yuan and L. Gu, Feature selection based on artificial bee colony and gradient boosting decision tree, *Applied Soft Computing*, vol.74, pp.634-642, 2019.

[17] R. Ahmad, R. Wazirali, Q. Bsoul, T. Abu-Ain and W. Abu-Ain, Feature-selection and mutual-clustering approaches to improve DoS detection and maintain WSNs' lifetime, *Sensors*, vol.21, no.14, 4821, 2021.

[18] M. Hosni and A. Idri, Software development effort estimation using feature selection techniques, *SoMeT*, pp.439-452, 2018.

[19] B. Tran, B. Xue and M. Zhang, Variable-length particle swarm optimization for feature selection on high-dimensional classification, *IEEE Transactions on Evolutionary Computation*, vol.23, no.3, pp.473-487, 2018.

[20] M. Rostami, S. Forouzandeh, K. Berahmand and M. Soltani, Integration of multi-objective PSO based feature selection and node centrality for medical datasets, *Genomics*, vol.112, no.6, pp.4370-4384, 2020.

[21] Y. Sanjalawe and T. Althobaiti, DDoS attack detection in cloud computing based on ensemble feature selection and deep learning, *Computers, Materials & Continua*, vol.75, no.2, 2023.

[22] H. Mohammadzadeh and F. S. Gharehchopogh, A novel hybrid whale optimization algorithm with flower pollination algorithm for feature selection: Case study Email spam detection, *Computational Intelligence*, vol.37, no.1, pp.176-209, 2021.

[23] M. F. Nadeem, A. Khalil, I. Sajjad, A. Raza, M. Q. Iqbal, R. Bo, W. ur Rehman et al., Review of flower pollination algorithm: Applications and variants, *2020 International Conference on Engineering and Emerging Technologies (ICEET)*, pp.1-6, 2020.

[24] H. Zhang, L. Ge, G. Zhang, J. Fan, D. Li and C. Xu, A two-stage intrusion detection method based on light gradient boosting machine and autoencoder, *Mathematical Biosciences and Engineering*, vol.20, no.4, pp.6966-6992, 2023.

[25] G. Ke, Q. Meng, T. Finley, T. Wang, W. Chen, W. Ma, Q. Ye and T.-Y. Liu, LightGBM: A highly efficient gradient boosting decision tree, *Advances in Neural Information Processing Systems*, vol.30, 2017.

[26] M. Al-Kasassbeh, M. A. Abbadi and A. M. Al-Bustanji, LightGBM algorithm for malware detection, *Intelligent Computing: Proceedings of the 2020 Computing Conference*, vol.3, pp.391-403, 2020.

[27] X.-S. Yang, Flower pollination algorithm for global optimization, *International Conference on Unconventional Computing and Natural Computation*, pp.240-249, 2012.