

Improving the Security of Internet of Things (IoT) Applications Based on a New Machine Learning Technique

*Alia A. Hussein*¹, *Ali J. Ramadhan*^{1,2*}, *Ali TaeiZadeh*² and *Mohand Hussein Issa*³

¹University of Alkafeel, Najaf, Iraq

²University of Qom, Qom, Iran

³Ministry of Education, Directorate of Education, Najaf, Iraq

Abstract. The Internet of Things, or IoT, is changing practically every aspect of modern life and entering both the business and residential domains. The proliferation of IoT has led to a rise in the frequency of cyber catastrophes. Attackers are using new methods or changing old ones, making the danger more sophisticated. The threat landscape that security professionals face is dynamic, complex, and diversified. This paper proposes a novel approach to enhance Internet of Things applications by fusing the swarm intelligence of Salp Swarm Algorithms (SSA) with the predictive power of Random Forest (RF) and Decision Tree (DT) models. Salp Swarm Algorithms simulate the cooperative behavior of salps in the natural world, wherein individual agents coordinate their actions to achieve common goals. This work uses SSA to optimize the Random Forest and Decision Tree model training process in an IoT context. SSA's collaborative nature makes it easier to explore the solution space effectively, which enhances the models' ability to capture the complex correlations found in IoT data. The effectiveness of the model is evaluated. We were able to attain a maximum accuracy of 95.54% for the Decision Tree of the OT-MQTT dataset and 96.19% for the random forest.

1. Introduction

A number of items that have been upgraded with CPUs and network cards to allow Internet connectivity and data exchange make up the Internet of Things, or IoT. The term "things" here refers to a broad category of objects that can be controlled by users through the internet, applications, or other interfaces, ranging from sophisticated cloud servers to basic sensors [1]. These gadgets can communicate with one another with little assistance from humans. IoT technologies bring smart applications to cities, factories, healthcare, education, and many other fields, which enhance the quality of life [2-3]. Although the Internet of Things was only introduced in 2008 and 2009, it is already one of the fastest-growing industries [4]. According to a Statista poll [5], as illustrated in Figure 1, there will be more than 50 billion IoT-connected devices by 2023 and 75 billion by 2025.

* Corresponding author: ali.j.r@alkafeel.edu.iq

Experts have acknowledged that despite this expansion, a large number of IoT devices remain weak security links and intrinsically susceptible [6]. Our objective in this work was to create a novel machine-learning method to enhance Internet of Things (IoT) application security. The MATLAB 2023b simulator was used to design, implement, test, analyze, and evaluate the proposed system technique.

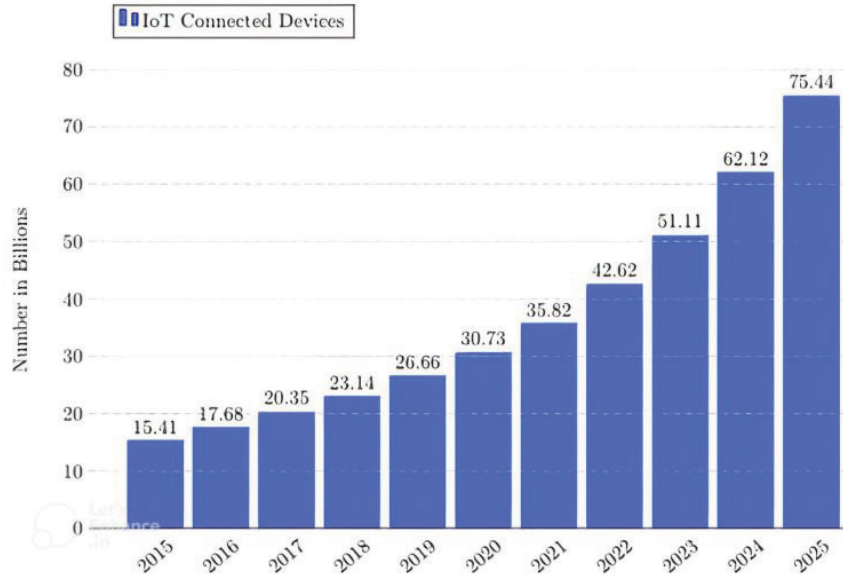


Fig. 1. Number of IoT devices worldwide from 2015 to 2025.

This is how the rest of the paper is structured. An overview of the suggested methodology and system is provided in Section 2. Performance measurements and the design and implementation of the suggested technique are covered in Section 3, and the outcomes are covered in Section 4. Finally, the conclusions are presented in Section 5.

2. PROPOSED METHODOLOGY

The dataset, methods, and performance measures are thoroughly explained in this section. The research project is depicted in Figure 5, and we also used data from an open-source platform (OT-MQTT dataset). Using preprocessing techniques, we have preprocessed the dataset. After eliminating null values from datasets, balancing techniques were used to scale and balance the data. We have divided the data into 70% training and 30% testing sets after extracting the best features. The testing set is used to assess the models, whereas the training set is utilized to train the machine learning models.

3. FEATURE SELECTION

We introduce the Salp Swarm Algorithm (SSA) of Mirjalili et al. [7]. The SSA belongs to the class of metaheuristic swarm-based algorithms. The movement patterns of sea salp population chains are imitated by the SSA, a swarm intelligence optimization system [8]. The primary goal of optimization techniques is to maximize the objective function or fitness function to identify the optimal choice (problem-solution). Finding the best value among several feasible options is the aim of the decision-making process. The best option or value among all the possibilities is chosen as a result of the optimization procedure [9]. Figures 2 and 3 [9] illustrate individual salp animals, salp chains, and the ideas of leader and follower. Table 1 presents the pseudocode for the salp swarm algorithm (SSA) [9].

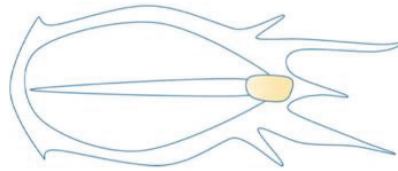


Fig. 2. An example of a single salp animal [9].

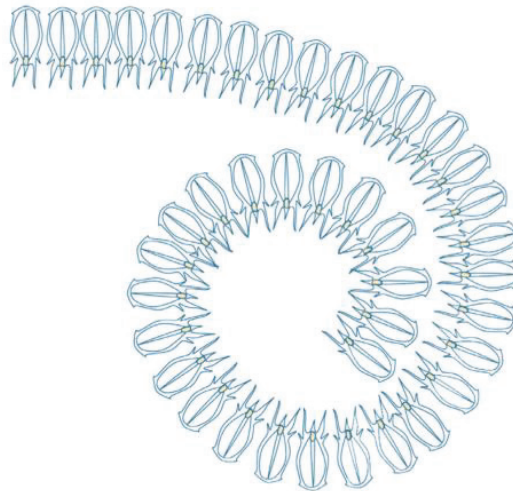


Fig. 3. Illustration of a salp chain with the idea of a leader and a follower [9].

Table 1. Algorithm of the pseudocode for the salp swarm algorithm (SSA) [9]

No.	Input parameter: Population Size, Number of Iterations, Min Values, Max Values
1	Initialize the salp population X_i ($i = 1, 2, 3, \dots, n$) considering ub and lb
2	while (end condition is not satisfied) do
3	calculate the fitness of each search agent (salp)
4	F = the best search agent
5	update c1 by Equation (3)
6	for each salp (X_i)
7	if ($i == 1$)
8	Update the position of the leading salp by Equation (2)
9	Else
10	Update the position of the follower salp by Equation (5)
11	End
12	End
13	Amend the salps based on the upper and lower bounds of variables
14	End
15	Return f
Output: Global best solution	

4. MACHINE LEARNING

The most popular methods for creating IDSs are machine learning (ML) algorithms [10]. The basis of machine learning techniques is building an explicit or implicit model that enables the classification of patterns in unprocessed data. Machine learning techniques for IDSs might employ single, hybrid, or ensemble classifiers. The used classifier has three different operating modes: semi-supervised, supervised, and unsupervised. Overall, supervised mode outperforms the other choices [11]. A few machine-learning techniques used in IDSs are artificial neural networks, logistic regression, k-nearest Neighbors, Naive Bayes, genetic algorithms, support vector machines, and decision trees. Four basic phases are involved in developing a machine learning model: data collection, data preparation, model selection and training, and model evaluation [12], as shown in Figure 4 [13].

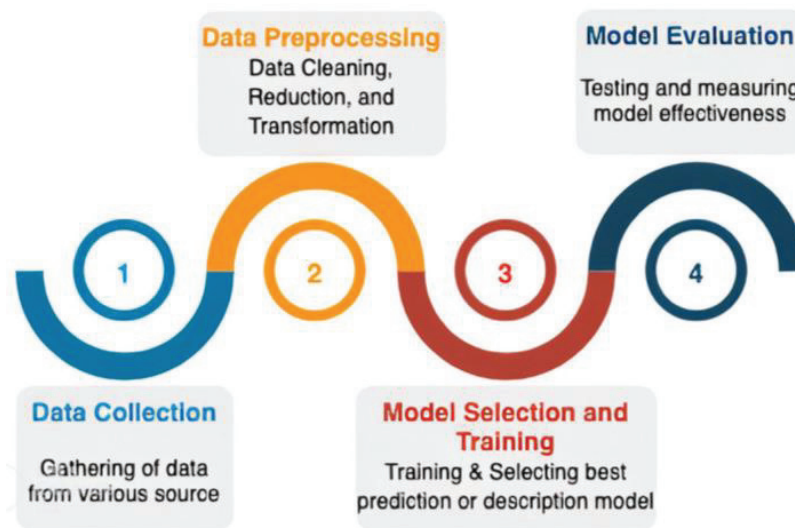


Fig. 4. Steps of Machine Learning.

4.1 DECISION TREE (DT)

Decision Tree belongs to the family of supervised learning. It is mostly used for problem-solving strategies for loan classification and prediction. Tree-based learning algorithms are often used with supervised learning prediction models to achieve high precision. Decision trees can manage digital and categorical data. Decision trees can assist you in coming to a decision or making a decision [12].

4.2 RANDOM FORESTS (RF)

During training, several decor-related decision trees are randomly created and combined into a "forest," which then outputs the class result. This technique is known as random forests. In a classification task, the outcome is the mode of classes; in a regression job, the outcome is the mean prediction of each tree. To boost accuracy and get rid of the decision tree's tendency to overfit its training set, RF combines the flexibility of bagging assembling techniques with the simplicity of decision trees [14].

5. PROPOSED SYSTEM

This section describes the proposed system the suggested system that combines Decision Tree (DT) and Random Forest for classification with the Salp Swarm Algorithm for feature selection of the network offers a creative and promising answer to the optimization and classification problems in this particular network structure see Fig. 5 Explains the proposed system.

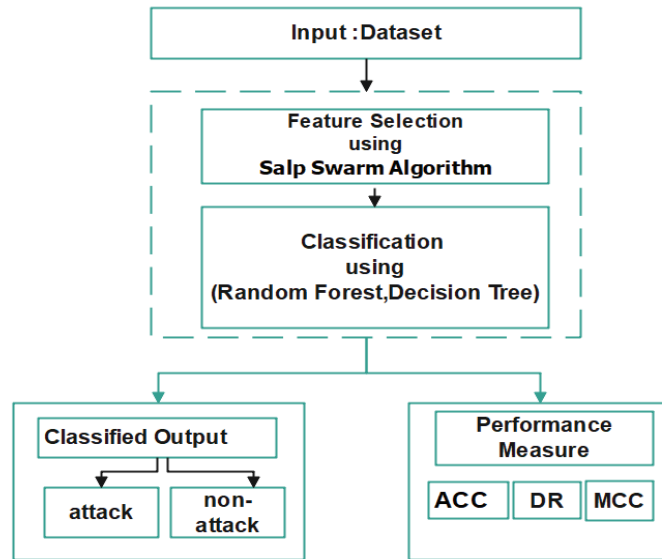


Fig. 5. The proposed system.

6. DESCRIPTION OF THE IOT-MQTT DATASETS

There are two different kinds of assaults in the IoT-MQTT dataset: brute force and scanning. An attacker uses a scanning attack to search across linked devices on a network and collect useful data about the operating systems and services they are using. It is possible to launch attacks using this information. They used the aggressive scan and UDP scan variants of this attack. A hacker employing a brute-force attack attempts every conceivable combination to guess vital data, including encryption keys and login credentials. Brute-force attacks on Sparta SSH and MQTT were conducted. The simulated environment's Ethernet traffic was gathered using the tcp dump utility. They created features that were both packet-based and flow-based, and their tests revealed that flow-based features performed better at differentiating between attack and regular behavior [15].

Table 2. The unidirectional and bidirectional flow-based feature types [17].

Feature Name	Description
num_pkts	Number of Packets in the flow
mean_iat	Average inter-arrival time
std_iat	Standard deviation of inter-arrival time
min_iat	Minimum inter-arrival time
max_iat	Maximum inter-arrival time
mean_pkt_len	Average packet length
num_urg_flags	Number of urgent flags
std_pkt_len	Standard deviation packet length
min_pkt_len	Minimum packet length
max_pkt_len	Maximum packet length
num_bytes	Number of bytes
num_psh_flags	Number of push flag
num_rst_flags	Number of reset flag

In our studies, we employed the unidirectional and bidirectional flow-based feature types (see Table 2 in above).

7. PERFORMANCE METRICS

This study includes all impacts in the accuracy, or F1-Score, PPR, DR, and MCC, because it consistently integrates correctness and recalls into a single value for evaluating the entire system's performance. See Table 3. True Positives (TP) are the quantity of attack episodes that are categorized as attacks. True Negatives are the quantity of normal events that are accurately classified as normal (TN). False Positives (FP) are the quantity of commonplace events that are incorrectly categorized as attacks. The number of real assault cases mistakenly classified as routine occurrences is known as False Negatives (FN).

Table 3. Confusion Matrix

Predicted	Actual positive	Actual negative
Positive	TP	FP
Negative	FN	TN

The accuracy value can be calculated directly from the confusion matrix in Equation (1.1) [16].

$$\text{Accuracy (ACC)} = \frac{TP+TN}{TP+FP+TN+FN} \times 100\% \quad (1)$$

This criterion is assessed using Equation (1.2) [17]. The division of all samples predicts true positive labels on all samples with a true positive and false negative.

$$\text{Precision (PPR)} = \frac{Tp}{FP+Tp} \times 100\% \quad (2)$$

$$\text{Detection Rate (DR)} = \frac{Tp}{TP+FN} \times 100\% \quad (3)$$

The F1-score is crucial for comparing performance between classes because it is calculated independently for each class resolution and accuracy. It can be calculated using Equation (1.4) after obtaining precision and recall [18].

$$\text{Recall} = \frac{TP}{(TP+FN)} \times 100\% \quad (4)$$

$$\text{F1-Score} = 2 * \left(\frac{PPR * SEN}{PPR + SEN} \right) \times 100\% \quad (5)$$

MCC uses the confusion matrix's TP, TN, FP, and FN factors. Equation (2.7) can be used to calculate the MCC score.

$$\text{MCC} = \frac{(TP * TN - FP * FN)}{\sqrt{(TP+FP) * (TP+FN) * (TN+FP) * (TN+FN)}} \times 100\% \quad (6)$$

The values produced by the MCC score range from -1 to 1. MCC = 1 represents a classification that is 100% correct, while MCC = -1 represents a classification that is 100% erroneous.

8. RESULTS AND DISCUSSIONS

Figure 6 shows the confusion matrices that the SSA-RF and SSA-SVM models produce. The SSA-RF model has categorized a total of 16562 instances into class 0 and 5939 instances into class 1, as shown in Fig. 6a. Additionally, Fig. 6b demonstrates that the SSA-DT model has classified a total of 16383 cases into class 0 and 5921 instances into class 1.

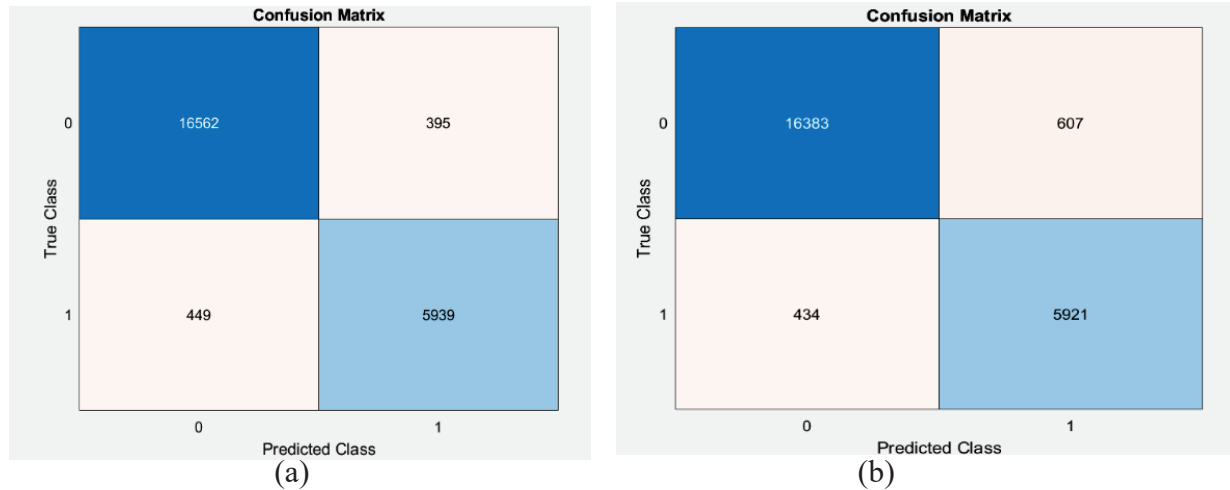


Fig. 6. Confusion Matrix of: a) SSA-RF. b) SSA-DT.

The results in Table 4 demonstrate how well the SSA algorithm fared in terms of DR, F1, and Accuracy in the IoT-MQTT dataset. In terms of outcomes, SSA+RF performed better than SSA+DT.

Table 4. Analysis of SSA+RF and SSA+DT method

Dataset	Method	ACC	F1	DR	MCC
Biflow	SSA+DT	95.54	91.92	90.70	88.86
	SSA+RF	96.19	93.03	93.29	90.41

The Figure 7 shows the ROC analysis of the SSA+RF and SSA+DT models. These figures pointed out that the SSA+RF model has obtained a ROC of 97.37 whereas the SSA+DT model has resulted in an decreased ROC of 95.55.

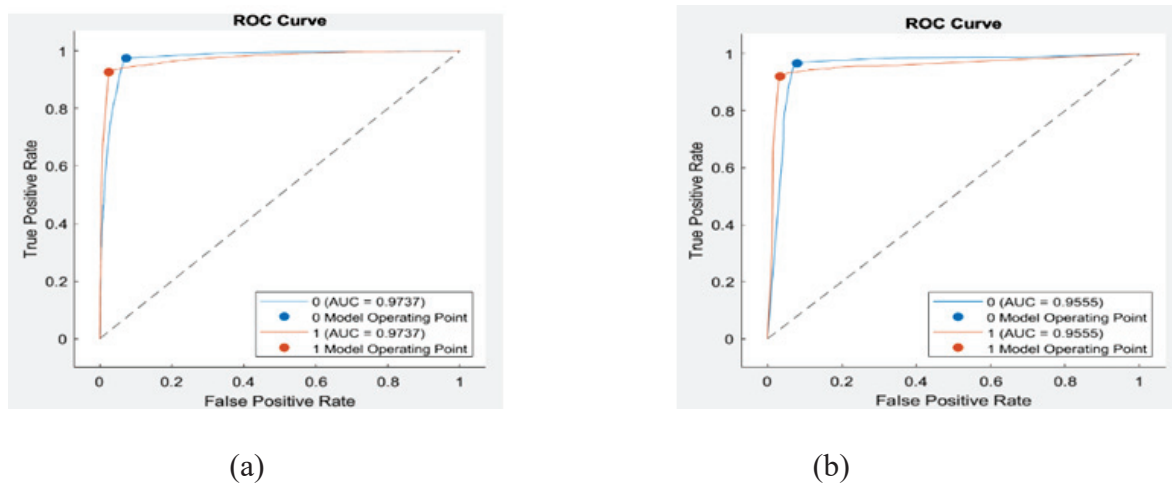


Fig. 7. ROC of: a) SSA-RF. b) SSA-DT.

9. CONCLUSION

An effective machine learning technique for intelligent intrusion detection in the Internet of Things (IoT) is swarm-based feature selection. Ant colony optimization and particle swarm optimization are two instances of swarm intelligence approaches that are used to address the challenges brought on by the massive dimensionality and dynamic nature of IoT data. Swarm-based feature selection has several benefits, including finding relevant features, lowering dimensionality, and quickly scanning a vast feature space. This improves the intrusion detection system's interpretability as well as the model's performance. The optimal qualities must be selected to build dependable and effective models that can accurately identify aberrant activity in IoT networks. Swarm-based methodology also meshes well with Internet of Things settings' flexible and self-organizing features. Because these algorithms can adapt to changing threat scenarios and network conditions, they are perfect for dynamic, real-time intrusion detection applications. The efficiency of intrusion detection systems in the Internet of Things could be greatly increased by combining machine learning with swarm-based feature selection. As technology and research in this field advance, further development and use of these tactics will likely result in the production of more intelligent and strong security solutions for the evolving Internet of Things environment. The system's obtained results demonstrate that improvements in Internet of Things (IoT) security can be achieved by combining the swarm intelligence of Salp Swarm Algorithms (SSA) with the predictive strength of Random Forest (RF) and Decision Tree (DT) models. The accuracy of these combinations is 95.54% for DT and 96.19% for RF.

Our future research will focus on developing additional machine learning-based security solutions that are more suited for Internet of Things applications.

References

1. M. A. Al-Garadi, A. Mohamed, A. K. Al-Ali, X. Du, I. Ali, and M. Guizani, "A survey of machine and deep learning methods for internet of things (IoT) security," *IEEE Commun. Surv. Tutorials*, vol. **22**, no. 3, pp. 1646–1685, 2020.
2. H. He et al., "The security challenges in the IoT enabled cyber-physical systems and opportunities for evolutionary computing & other computational intelligence," in *2016 IEEE congress on evolutionary computation (CEC)*, 2016, pp. 1015–1021.
3. M. S. Harsha, B. M. Bhavani, and K. R. Kundhavai, "Analysis of vulnerabilities in MQTT security using Shodan API and implementation of its countermeasures via authentication and ACLs," in *2018 International Conference on Advances in Computing, Communications and Informatics (ICACCI)*, 2018, pp. 2244–2250.
4. D. Evans, "The internet of things," *How Next Evol. Internet is Chang. Everything*, Whitepaper, Cisco Internet Bus. Solut. Gr., vol. **1**, pp. 1–12, 2011.
5. "IoT devices installed base worldwide 2015-2025 | Statista." <https://www.statista.com/statistics/471264/iot-number-of-connected-devices-worldwide/> (accessed on 19 March 2024).
6. "Cyber Security Report 2023 | Check Point Software." <https://pages.checkpoint.com/cyber-security-report-2023.html> (accessed on 19 March 2024).
7. S. Mirjalili, A. H. Gandomi, S. Z. Mirjalili, S. Saremi, H. Faris, and S. M. Mirjalili, "Salp Swarm Algorithm: A bio-inspired optimizer for engineering design problems," *Adv. Eng. Softw.*, vol. **114**, pp. 163–191, 2017.
8. J. Zhang and J.-S. Wang, "Improved salp swarm algorithm based on levy flight and sine cosine operator," *Ieee Access*, vol. **8**, pp. 99740–99771, 2020.
9. L. Abualigah, M. Shehab, M. Alshinwan, and H. Alabool, "Salp swarm algorithm: a comprehensive survey," *Neural Comput. Appl.*, vol. **32**, pp. 11195–11215, 2020.

10. A. Özgür and H. Erdem, “A review of KDD99 dataset usage in intrusion detection and machine learning between 2010 and 2015,” 2016.
11. H. Moradi Koupaie, S. Ibrahim, and J. Hosseinkhani, “Outlier detection in stream data by machine learning and feature selection methods,” *Int. J. Adv. Comput. Sci. Inf. Technol. Vol*, vol. **2**, pp. 17–24, 2014.
12. A. L. Buczak and E. Guven, “A survey of data mining and machine learning methods for cyber security intrusion detection,” *IEEE Commun. Surv. tutorials*, vol. **18**, no. 2, pp. 1153–1176, 2015.
13. M. A. Umar and C. Zhanfang, “Effects of Feature Selection and Normalization on Network Intrusion Detection,” *Authorea Prepr.*, 2023.
14. H. Tchakoute-Tchuigoua and I. Soumaré, “The effect of loan approval decentralization on microfinance institutions’ outreach and loan portfolio quality,” *J. Bus. Res.*, vol. **94**, pp. 1–17, 2019.
15. T. Hastie, R. Tibshirani, J. H. Friedman, and J. H. Friedman, *The elements of statistical learning: data mining, inference, and prediction*, vol. **2**. Springer, 2009.
16. H. Hindy, E. Bayne, M. Bures, R. Atkinson, C. Tachtatzis, and X. Bellekens, “Machine learning based IoT intrusion detection system: An MQTT case study (MQTT-IoT-IDS2020 dataset),” in *International networking conference*, 2020, pp. 73–84.
17. H. Hindy, C. Tachtatzis, R. Atkinson, E. Bayne, and X. Bellekens, “Mqtt internet of things intrusion detection dataset.” Jun, 2020.
18. M. Buckland and F. Gey, “The relationship between recall and precision,” *J. Am. Soc. Inf. Sci.*, vol. **45**, no. 1, pp. 12–19, 1994.
19. D. Chicco and G. Jurman, “The advantages of the Matthews correlation coefficient (MCC) over F1 score and accuracy in binary classification evaluation,” *BMC Genomics*, vol. **21**, pp. 1–13, 2020.