Intelligent IoT Security Monitoring based on Fuzzy Optimum-Path Forest Classifier

Yongzhao Xu
Dongguan University of Technology
Renato Souza
IFCE: Instituto Federal de Educacao Ciencia e Tecnologia do Ceara
Elias Medeiros
IFCE: Instituto Federal de Educacao Ciencia e Tecnologia do Ceara
Neha Jain
UFC: Universidade Federal do Ceara
Lijuan Zhang
Dongguan University of Technology
Leandro Passos
UNESP: Universidade Estadual Paulista Julio de Mesquita Filho
Victor Hugo C. de Albuquerque (victor.albuquerque@ieee.org)
Federal University of Ceara: Universidade Federal do Ceara

Research Article

Keywords: Intrusion detection, Fuzzy Optimum-Path For- est, IoT, Machine Learning

Posted Date: March 18th, 2022

DOI: https://doi.org/10.21203/rs.3.rs-1199639/v1

Intelligent IoT Security Monitoring based on Fuzzy Optimum-Path Forest Classifier

Yongzhao Xu, Renato W. R. de Souza, Elias P. Medeiros, Neha Jain, Lijuan Zhang, Leandro A. Passos, and Victor Hugo C. de Albuquerque

***Abstract*—The detection of intrusions in IoT networks is essential to maintain the availability and integrity of data transmitted and generated by devices connected to these networks. This is primarily when the data originates from critical activities, such as activities in the military, financial, industrial, and health sectors. Machine learning techniques have been adopted to create ways to detect or improve the accuracy of existing models for automatic intrusion detection. However, it is difficult to find in the literature an accurate intrusion detection technique in anIoT environment, as there are different types of attacks that can happen in different ways. Therefore, to solve this problem, this work proposes applying Fuzzy OPF (Optimum-Path Forest) as a new detection algorithm for any threat that escapes the regular traffic of an IoT network. We evaluate our proposed approach by using five different ML algorithms: Linear Discriminant Analysis, Support Vector Machine, Bayes, K-Nearest Neighbors, and Optimum-Path Forest. Experimental results analysis showed that our proposed model outperforms well-known algorithms in the literature regarding the Accuracy, Recall, and F1 metrics.**

***Index Terms*—Intrusion detection, Fuzzy Optimum-Path Forest, IoT, Machine Learning**

I. Introduction

INternet of Things (IoT) is a concept used to refer to a vast number of connected electronic devices capable of transmitting and collecting data over the internet [1]. Currently, billions of electronic devices are connected to the internet [2]. IoT can be applied to monitor environment-related events, collect information associated with human behavior, monitor industrial activities, and provide information for military operations. [3]. The data generated in the IoT environment poses integrity risks as it can be easily manipulated and destroyed by attacks from the outside world. [4]. Attacks can happen in several ways, and types:

- Physicals attacks (e.g., node tampering, and node attack).
- Software attacks (e.g., code injection, and data privacy issues).
- Network attacks (e.g., sybil attack, and blackhole at- tack) [3, 5].

In this perspective, IoT security is essential, especially when the data generated is information from critical tasks, such as activities in the military, industrial, financial, or health sectors. [6].

In the last decades, machine learning (ML) techniques have been applied to developing intelligent systems for solving problems in various areas (image recognition [7], medical diagnosis [8], remote sensing [9], and others) and achieving good results, including cyber-attack detection in IoT environments. The objective of these works is to find solutions that can prevent, detect, or mitigate attacks on this type of network.

Thus, we can find in recent literature several works that address the use of ML algorithms in intrusion detection [10–15], in summary, all these works show that the components of an IoT network are limited concerning their computational capacity, which makes traditional security methods such as encryption or firewalls impractical to protect these devices. As an alternative to this context of computational limitation of IoT devices, Intrusion Detection Systems (IDSs) arise, in which they classify and detect anomalies in network communication.

Thus, several works have been using machine learning algorithms to build Intrusion Detection Systems (IDS). Cheema et al. [16] presented an intrusion detection system based on distributed machine learning using Blockchain technology, which divides the IoT network into autonomous systems. The classification technique - support vector machine (SVM) was trained using the datasets obtained from each of the nodes of the IoT network.

Other recent approaches point to the need for data quality used in the construction of IDS, as an outlier or anomaly can degrade data quality and, therefore, affect the final decision. Therefore, some works deal with preparing data to be used more effectively by ML algorithms. The results show that the better the quality of the data will be the results obtained in the classification[17–19].

Recently, some works presented hybrid approaches. Chkirbene et al. [20] presented a proposal that combines two machine learning algorithms to detect attacks through efficientresource selection and classification. They use Random Forestto select important dataset features and Classification and Regression Trees (CART) to classify different attack classes. Rachid and Ghazi [21] proposed a cloud system for real-time intrusion detection and monitoring communication and attacks before they spread across the network. Alalade [22] used Extreme Learning Machine and Artificial Immune System

Xu Yongzhao is with the Dongguan University of Technology, China.(email: xuyz@dgut.edu.cn)

Renato William R. de Souza is with University of Fortaleza, Brazil. (email: renatowilliam@edu.unifor.br

Elias P. Medeiros is with the Federal Institute of Education, Science and Technology of Ceará, Brazil.(email: elias.paulino@ifce.edu.br)

Neha Jain is with the Graduate Program in Electrical Engineering, Federal University of Ceará, Fortaleza/CE, Brazil.(email: neha.juet@gmail.com)

Lijuan Zhang is with the Dongguan University of Technology, China.(email: zhanglijuan@dgut.edu.cn)

Victor Hugo C. de Albuquerque is with the Department of Teleinformatics Engineering, Federal University of Ceará, Fortaleza/CE, Brazil (e-mail: victor.albuquerque@ieee.org)

Leandro A. Passos is with the São Paulo State University, Brazil. (email: leandropassosjr@gmail.com).

(AIS-ELM) to build an IDS to detect anomalies in home networks. In all works, the results show that the proposed method achieves a good performance compared to existing algorithms, which makes hybrid algorithms a good alternative for intrusion detection in IoT networks. Maniriho et al.[23] presented the new anomaly-based approach to IoT networks that are implemented with a resource selection mechanism. The proposal uses the Random Forest algorithm to classify traffic as normal or anomalous. Performance was evaluated using a recent anomaly detection dataset, IoTID20. The resultsachieve an accuracy of 99.9% in detecting DoS attacks.

Other works address the use of IDSs in the industrial context, the so-called Industrial IoT (IIoT). Arshad et al. [24] proposed an intrusion detection framework for the energy- constrained IoT devices that form the basis of an IIoT ecosys- tem. The ad hoc nature of these systems, as well as com-plex emerging threats such as botnets, utilized collaboration between the host (IoT devices) and the Gateways, minimiz- ing energy consumption and communication overhead. This proposal was implemented with the Contiki operating system. The results show that the proposed structure can minimize energy and communication costs while achieving effective collaborative intrusion detection for IIoT systems. Hassan et al. [25] proposed a cooperative data generator based on a trained downsampler encoder using a DL algorithm and ML techniques to ensure better performance of detection models in IIoT environment. The results show that the proposed approach outperforms conventional deep learning and other ML techniques. Magaia et al. [26] used deep reinforcement learning techniques available for IIOT in smart cities, in addition to recurrent neural networks and convolutional neural networks.

In addition to ML algorithms, some works also propose the use of Deep Learning (DL). Sugi and Ratna [27] presented an IDS model based on DL and ML to overcome security attacks in IoT networks, using K-Nearest Neighbor (KNN), and Long Short-Term Memory (LSTM). The ML techniques were compared and evaluated according to detection time, sensitivity, and kappa statistics. The approach was trained and tested using the Bot-IoT dataset.

Alkadi et al. [28] presented a Deep Blockchain Framework (DBF) designed to detect distributed types of intrusions. The method used a Bidirectional Long Short-Term Memory (BiLSTM) deep learning algorithm to detect attacks such as TCP DDoS, UDP DDoS, HTTP DDoS, TCP DoS, UDP DoS, and HTTP DoS. Qiao et al. [4] proposed a deep learningmodel Stacked De-noising Auto-encoder Supporting Vector Machine (SDAE-SVM) for intrusion detection in the TCP/IP layers of an IoT network, including the transport, network, and application layers. Ravi et al. [29] created a semi-supervised model called SDRK to detect intrusion in fog nodes between the IoT layers and the cloud. Tian et al. [30] proposed a distributed system for detecting web attacks, analyzing URLs using the deep learning technique M-ResNet. Guimaraes et al. [31] used a supervised version of Optimum-Path Forest (OPF) to classify anomalies in wireless sensor networks.

Other proposals based on Fuzzy concepts have also been studied for the detection of attacks in IoT networks. Cristiani et al. [32] proposed a model called Fuzzy Intrusion Detection System for IoT Networks (FROST). FROST uses the basis of fuzzy theory to make learning models more flexible, seeking to improve performance in the classification of imprecise data. Naik et al. [33] built a dynamic fuzzy rule interpolation (D-FRI) approach to improve the Fuzzy rule interpolation (FRI) model that works with static rules. D-FRI was employed to support network security analysis in constructing an intelligent intrusion detection system (IDS). Manimurugan et al. [34] presented an algorithm based on the combination of Crow Search Optimization (CSO) and Adaptive Neuro-Fuzzy Inference System (ANFIS) techniques.

Fuzzy Optimum-Path Forest (Fuzzy OPF) [35] is a variant of the OPF classifier designed as a pattern recognition technique. Recently, OPF and its variations have been used to design approaches for classifying problems in different areas, including detecting anomalies in wireless networks and obtaining promising results [8, 36, 37]. However, so far, it has not been possible to identify a work that evaluates Fuzzy OPF in the context of intrusion detection in IoT networks.

The main contributions of this work are:

- A new algorithm for detecting intrusions in IoT environments.
- The proposed algorithm has better performance than other well-known algorithms in the literature.

The remainder of this paper is organized as follows. Sec- tion II describes the data sets and configurations of the experiments performed in this work will be presented. We willalso introduce the Fuzzy OPF algorithm and the experimental results III, respectively. Finally, Section IV states conclusions and future works.

II. Methodology

In this section, the data sets and configurations of the experiments performed in this work will be presented. We will also introduce the Fuzzy OPF algorithm.

A. Fuzzy Optimum-Path Forest

Fuzzy OPF [35] is a new machine learning algorithm developed from Optimum-Path Forest (OPF) that applies fuzzy logic to improve sample selection and classifier output performance, also contributing to alleviate some problems, such as: noise, unbalanced classes and outliers.

Fuzzy OPF applies at each entry point to Fuzzy association where samples have a degree of participation in each class, that is, each sample will have a value that is calculated by considering the density of the instance that is made with the unsupervised form. In [38] displays the padding of the region where the swatch is in the resource space. According to the Graph $G = (N, A)$ where N imposes the grouping of the train classes nodes and characterize the group of edges that connect each pair of samples.

This cited process is done with the unsupervised model fo OPF [38] which has the function of grouping the trainingdata, in addition to a sample density calculation (q) this is done using a function that will calculate the probability density(PDF), as follows:

$$\rho(\boldsymbol{q}) = \frac{1}{\sqrt{2\pi\psi^2 k}} \sum_{\forall u \in A_k(\boldsymbol{q})} \exp \frac{-d^2(\boldsymbol{q}, \boldsymbol{u})}{2\psi^2}, \quad (1)$$

where $A_k(\boldsymbol{q})$ stands for the k-Neighborhood of sample $\boldsymbol{q}$, $\psi = \frac{d_f}{3}$, and d_f is the highest value between the edges of the graph. (N, A).

The function for Fuzzy membership $F_\Theta(\boldsymbol{q}) \in [0, 1]$, where $\Theta = \sigma, \rho_{min}, \rho_{max}$ describes the function of the parameter sets, in which in the training step calculates and assigns a real value to each sample $\boldsymbol{q}$, thus defining membership for the class.For an adequate membership of members, some restrictions are considered: (i) Determination of an ideal parameter as a lower limit $\sigma > 0$, (ii) Have the model's ability to describe the behavior and properties of samples [39]. below is the equation for membership for fuzzy opf.:

$$F_\Theta(\boldsymbol{q}) = (1 - \sigma) \left(\frac{\rho(\boldsymbol{q}) - \rho_{min}}{\rho_{max} - \rho_{min}} \right)^2 + \sigma, \quad (2)$$

where $\rho_{min} \leq \rho(\boldsymbol{q}) \leq \rho_{max}$, and ρ_{min} and ρ_{max} set the lowest and highest densities, respectively.

Briefly, we can say that the data situated at the limits of the groups have a lower density value, being considered weak for a given class, that is, they have a lower "strength" in the conquest process. This causes samples far from the center of the clusters to be penalized, thus helping the problem of over-adjustments. As is done in the standard OPF training step, the Fuzzy OPF chooses of the most significant samples, that is, through the conquest process, the prototypes that compete with each other provide the best path cost for the rest of the samples. This process is done through the path cost function f_{max}, as follows:

$$f_{max}(\langle \boldsymbol{q} \rangle) = \begin{cases} 0 & \text{if } \boldsymbol{q} \in T, \\ +\infty & \text{otherwise} \end{cases}$$
$$f_{max}(\varphi_{\boldsymbol{q}} \cdot \langle \boldsymbol{q}, \boldsymbol{u} \rangle) = \max\{f_{max}(\varphi_{\boldsymbol{q}}), d(\boldsymbol{q}, \boldsymbol{u})\}, \quad (3)$$

Here T defines the prototype group, $\varphi_{\boldsymbol{q}}$ corresponds to a path of related samples in T and concluding in sample $\boldsymbol{q}$, and $d(\boldsymbol{q}, \boldsymbol{u})$ is the cost between the samples $\boldsymbol{q}$ and $\boldsymbol{u}$. However, $\varphi_{\boldsymbol{q}} \cdot \langle \boldsymbol{q}, \boldsymbol{u} \rangle$ estabelece o caminho $\varphi_{\boldsymbol{q}}$ e a borda $\langle \boldsymbol{q}, \boldsymbol{u} \rangle$ sets the path $f_{max}(\varphi_{\boldsymbol{q}})$ characterizes the highest cost among nearby samples $\varphi_{\boldsymbol{q}}$.

Conjecture $T^* \subseteq T$ as a grouping of samples that alleviate the errors of the training step [1] The Fuzzy OPF in its training process assigns for all sample $\boldsymbol{u} \in N$ an optimal value $P(\boldsymbol{u})$, shown below:

$$P(\boldsymbol{u}) = \min_{\forall \boldsymbol{q} \in N} \{F_\Theta(\boldsymbol{u}) * \max\{P(\boldsymbol{q}), d(\boldsymbol{q}, \boldsymbol{u})\}\}, \quad (4)$$

Equation 4 is utilized to calculate the value of the cost of samples in the training and testing stage. It is important to emphasize that low values of fuzzy membership represent samples with little relevance to the training stage, whereas samples with high membership are more representative.

[1]Notice T^* is obtained after calculating the selection of nearby samples that have different labels and after performing the Minimum Spanning Tree.

Note $F_\vartheta(\boldsymbol{x}) \approx 0$ in Equation 2, $P(\boldsymbol{x})$ assigns a value equal to 0 in Equation 4, no more contains OPF features [35]. Thus, this work assumes sigma values within the range [0.2, 1.2] to avoid this problem. The implementation of the proposed model is presented in algorithm 1

Algorithm 1: Fuzzy Optimum-Path Forests algorithm

Input: G, T, λ, σ, d.
Output: O, P, C.
Auxiliary: L, cst, ρ, ρ_{min}, ρ_{max}.

```
for q in N do
    density ← ρ(q) using Equation 1;
    O(q) ← nil, P(q) ← +∞;
    membership ← Fθ(q) using Equation 2;
ρmin, ρmax ← min(density), max(density);
for q in T do
    P(q) ← 0, C(q) = λ(q), L ← q
while L ≠ ∅ do
    Remove from L a sample q such that P(q) is
     minimum;
    for u in N do
        if q /= u and P(u) > P(q) then
            cst ← Fθ(u) * max{P(q), d(q, u)};
            if cst < P(u) and P(u) = +∞ then
                Remove u from L
                C(u) ← C(q), O(u) ← q,
                P(u) ← cst;
                L ← u;
return [O, P, C]
```

The algorithm receives as input, a graph $G = (N, A)$ a set of prototypes $T \subseteq N$, map of training set labels λ, lower bound parameter σ, and distance function d. The outputs are predecessor map O, path-cost map P, and label map C. Five auxiliary variables are used: priority queue L, variable cst, density map ρ, and minimum and maximum densities ρ_{min}, and ρ_{max}, respectively.

Lines 1-4 present the density calculation for each sample, initialize predecessor and cost maps, and all samples have their fuzzy association values calculated, while in line 5 the parameters ρ_{min} and ρ_{max} are defined, used in the input for Equation 2. From line $6-7$ prototypes are initialized with a zero value referring to the cost of their true labels, according to the function λ, right after the initialization process, all prototypes are placed in the queue with priority.

The main loop is defined by lines 8-17, which corresponds to the competition process. In the case of prototypes that have a cost of zero they will be immediately taken from priority queue L line 9. From line 10 we have a loop, which calculates the best path cost of the training samples line 12. If the node is conquered (Line13) it will be eliminated from the priority queue line(14). Lastly, the 15 line updates the predecessor map with the cost value of all samples, assigning each node to the label of the prototype that conquered it.

B. Dataset Description

The approach used in the proposal is evaluated on two datasets, created based on real data Austin[40], Texas 2018 [2]. The Attack Dataset follows a specific pattern of fake data injection attack behavior in IOT networks. Data Set Austin, Texas 2018 is described in table I:

TABLE I: Binary class distribution in datasets

DataSet	Quantity				
	Samples	Normal	Intrusion	Class	Attributes
Austin Texas - Scenario[1]	1300	1020	280	2	5
Austin Texas - Scenario[2]	606	486	120	2	3

The characteristics of the datasets are presented in Table I where the number of samples, normal packages, intrusions, classes and attributes are shown.

C. Experimental Setup

The experiments cover the comparison of Fuzzy OPF with five traditional classifiers: Optimum-Path Forests (OPF), Support Vector Machine (SVM), Bayesian classifier (Bayes), K-Nearest Neighbors classifier (KNN) and Linear Discriminant Analysis classifier (LDA). As a methodological procedure, for a performance evaluation of the proposed model we used the cross Validation technique, where the dataset was divided and later executed 20 times. In each execution step the dataset was randomly divided into 70% for training, 15% for evaluation and 15% for the test step.

For the adjustments of the Fuzzy OPF hyperparameters we use the values of the evaluation set. k_{max} and σ through a search in the following ranges $k_{max} \in \{1, 10, 20, \ldots, 150\}$ and $\sigma \in \{0.2, 0.4, 0.6, 0.8, 1.0, 1.2\}$ Subsequently, these values are used to maximize the efficiency and accuracy of the Fuzzy OPF classifier.

Finally, it should be noted that the experiments were carried out using the library LibOPF [3], where the Optimum-Path Forest was implemented as well as the Fuzzy OPF. Further- more, we also use the implementations provided by Scikit- learn [41] for the KNN, Bayes, SVM and LDA classifiers.The experiments were done on a machine with 6*Gb* of RAM running an Intel® Core™ *i*3 *M* 380CPU@2.53*GHz* 4 and the Linux Ubuntu 20.04.2 operating system Version 64 bits.

D. Statistical Measures

To assess the performance of intrusion detection data classification, some statistical metrics are used. These metrics are essential to measuring the degree of success of the proposed work, especially when compared to other related solutions, as in the case of traditional classifiers.

This article uses five metrics for a more concise assessment of results.

- **Accuracy**: is a general probability of the correctness of the algorithm, i.e., the global average of the correctnessof both classes.
- **Recall**: is used to indicate the relationship between positive predictions made correctly and all predictions that are positive.
- **Intrusion**: is the probability that packets that make up the dataset samples characterized as intrusions are in fact intrusion samples.
- **Normal**: is the probability of packets that make up the dataset samples characterized as normal packets, whether they are normal samples.
- **F1 Score**: is the harmonic average between precision (Intrusion) and recall (Normal), thus appropriate for imbalanced datasets evaluation.

E. Proposed approach

Figure 1 gives an overview of the intrusion detection approach using the fuzzy optimum-path-forest.

The topology shown in the figure presents a network topology where all equipment is connected to a concentrator device (Switch), that is, all data traffic passes through this device, among them we have the IDS, which is a Monitoring Server which runs the OPF Fuzzy Sorter. This server monitoring collects all network data traffic and in the Data Collector module a data set is generated to train the Fuzzy OPF. Samples from this dataset are labeled intrusions and non-intrusions. Later, these resources are used to generate alerts on intrusion detection, where if there is an infected device on the network or an external cyber attack threat, the system will generate alarms so that a person responsible for the system makes a decision to stop this attack.

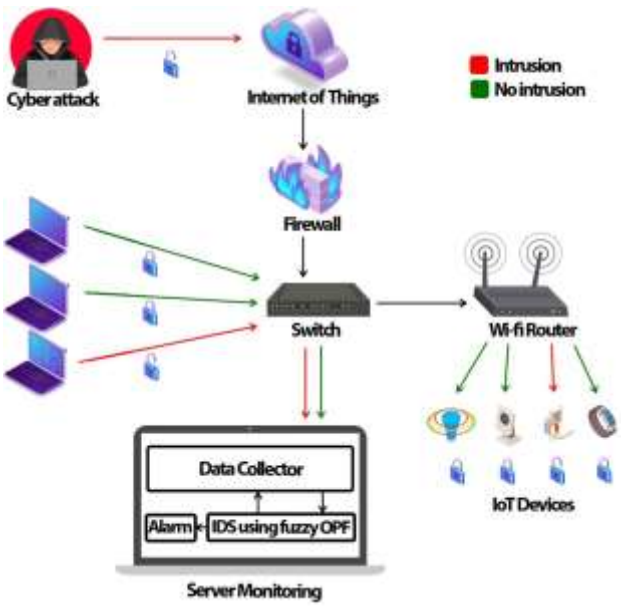


Fig. 1: Overview of the intrusion detection approach using the Fuzzy Optimum-Path-Forest.

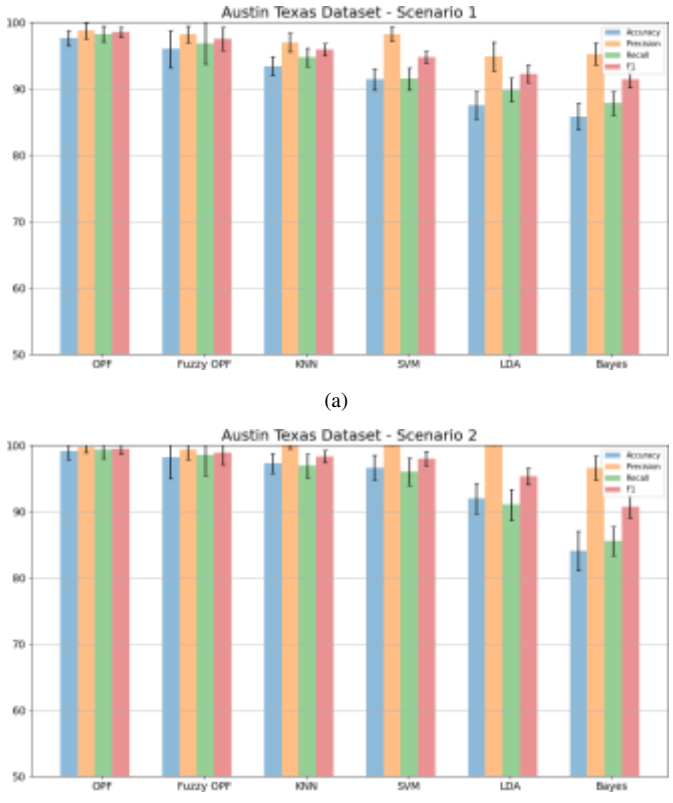
[2] https://https://ieee-dataport.org/documents/attackdatasetaustintexas2018
[3]https://github.com/jppbsi/LibOPF

III. Results

This section presents and discusses the results achieved during the development of the proposed approach and the adjustment process of the Fuzzy OPF hyperparameters.

First, we compare Fuzzy OPF performance with SVM, KNN, Bayes and LDA techniques. Fig. 2 shows the classification results considering the intrusion detection task for the Austin Texas - Scenario[1] and Austin Texas - Scenario[2] datasets. Results are presented in terms of accuracy, precision, recall, and F1.

In Scenario 1, we found that Fuzzy OPF and OPF obtained similar results. They reached Accuracy(%): 95.92(2.83) and 97.53(1.13); Precision(%): 98.16(1.24) and 98.69(1.18); Recall(%): 96.79(3.16) and 98.19(1.17); F1(%): 97.44(1.69) and 98.43(0.72), respectively. In scenario 2 the results of these two techniques are also similar. They reached Accuracy(%): 98.13(3.15) and 99.06(1.24); Precision(%): 99.24(1.43) and 99.58(0.78); Recall(%): 98.53(3.10) and 99.26(1.38); F1(%): 98.86(1.87) and 99.42(0.76), respectively.

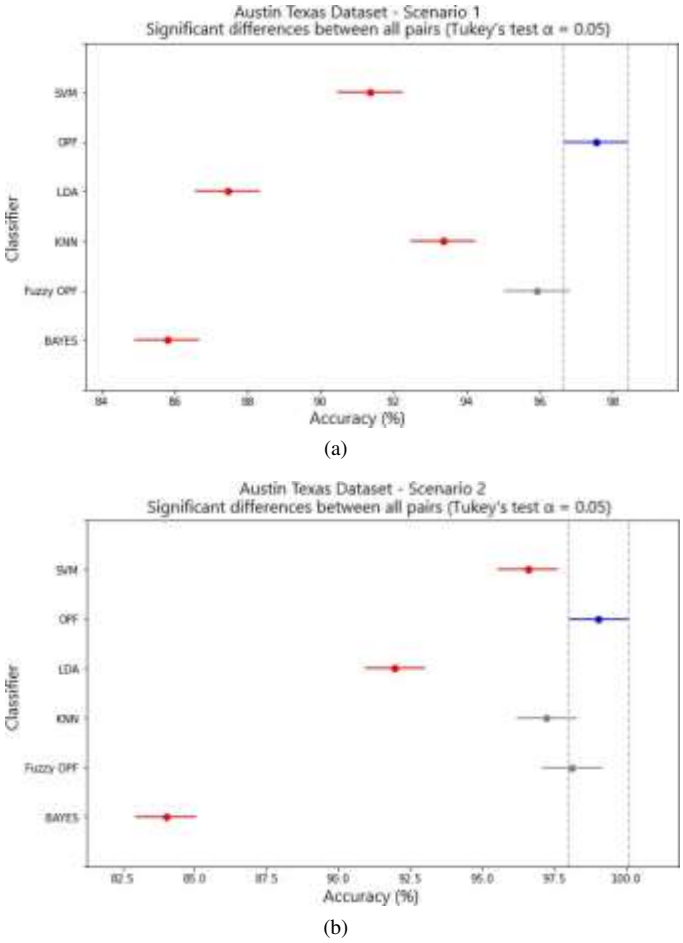


(a)

(b)

Fig. 2: Average Accuracy, Precision, Recall and F1 VALUES of Fuzzy OPF, OPF, Bayes, KNN, SVM and LDA classification. (a) Austin Texas - Scenario[1] dataset; (b) Austin Texas - Scenario[2] dataset.

Due to the proximity of the Fuzzy OPF and OPF results, Tukey's statistical test was used to verify the significant difference in accuracy between the classifiers. In Fig. 3 (a) we can see that there is no significant difference between Fuzzy OPF and OPF in scenario 1, however, they are superior to KNN, Bayes, SVM and LDA. A Fig. 3 (b) shows that, in scenario 2, there is no significant difference between Fuzzy OPF, OPF and KNN, but they are superior to Bayes, LDA and SVM.



(a)

(b)

Fig. 3: Significant differences between all pairs of classifiers accuracy. (a) Austin Texas - Scenario[1] dataset; (b) Austin Texas - Scenario[2] dataset.

We also investigate the model's discrimination rate between the intrusion and no intrusion classes. The Fig. 4 shows the Fuzzy OF and OFF confusion matrices for the two evaluated data scenarios. We see true positive cases above 98.16% for intrusion packages and 87.73% for no intrusion packages (false positive cases) in Scenario 1. In scenario 2, the true positive cases are above 99.24% for intrusion packages and 93.61% for no intrusion packages (false positive cases).

The computational load (in seconds) of each technique is presented in Table II. In both results of the bases, Fuzzy OPF obtained the longest time among the other classifiers in the training stage, which was already expected due to its training stage having a grouping process to calculate adherence before classification. However, when taking into account the test step, Fuzzy OPF has a satisfactory result, where in the dataset Austin Texas - Scenario[1] managed to have a time shorter than the default OPF, then SVM and KNN. In the Austin Texas - Scenario[2] dataset, it obtained a time statistically equal to the

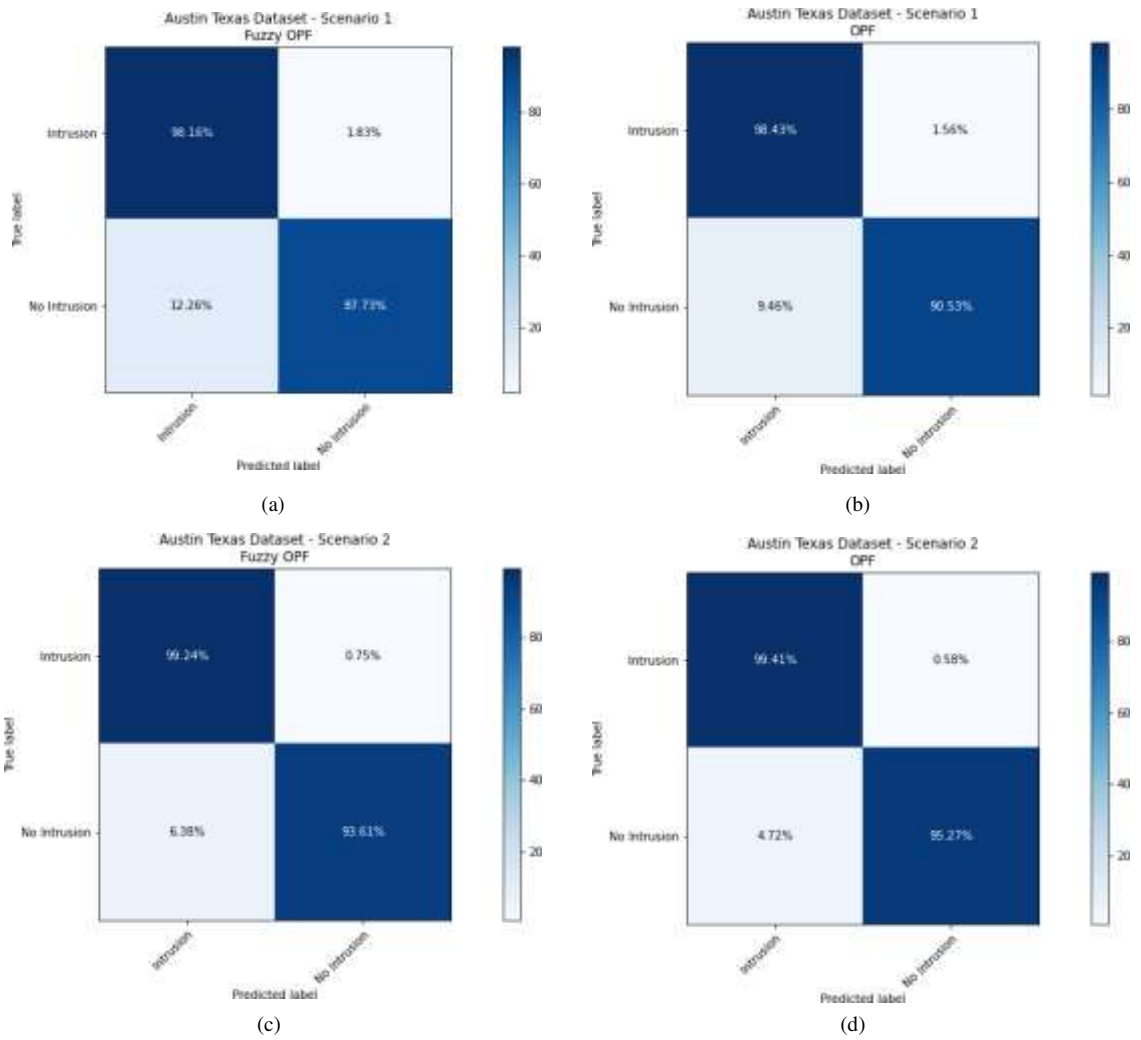


Fig. 4: Confusion matrix for the best results using Fuzzy OPF and OPF for Austin Texas - Scenario[1] (fig(a) and fig(b)) and Austin Texas - Scenario[2](fig(c) and fig(d)) datasets.

TABLE II: Mean training and testing times (in seconds) dataset.

DataSet	Time [seconds]	FuzzyOPF	OPF	SVM	Bayes	KNN	LDA
Austin Texas - Scenario[1]	Train	55.7321	0.0446	0.0289	**0.0145**	0.0195	0.0188
	Test	0.0042	0.0047	0.0058	**0.0019**	0.0196	0.0021
Austin Texas - Scenario[2]	Train	25.8156	0.0082	0.0109	0.0079	**0.0074**	0.0082
	Test	0.0007	**0.0006**	0.0024	0.0019	0.0105	0.0017

default OPF and less time than the other classifiers.

Although Fuzzy OPF has a lower performance than theother techniques in the training phase, it received an acceptable efficiency in the test phase, because according to the results in table II only OPF had a better result than it in the Austin Texas - Scenario[2]. since it had a shorter test time than the others, in the Austin Texas - Scenario[1] it had a shorter time than OPF, SVM and KNN. Thus, it is possible to state that Fuzzy OPF can be indicated as an option to run on low-power embedded computational devices.

This feature makes Fuzzy OPF suitable to be an intrusion detector for IOT networks, as it can achieve good results with qualified computational resources. .

A. Ablation

This section presents Fuzzy OPF hyperparameter optimization step considering the two data sets. In this context, figures 5(a) and 5(b) represent the grid-search procedure, where the possible arrangements of sigma and kmax are considered to provide the best results on the validation sets.

Regarding the Austin Texas - Scenario[1], it can be seen in figure 5 that the most accurate results were obtained in the sigma and kamx intervals, ie values that were between [$X : X; X : X$] for sigma and [$X : X; X : X$] for Kmax. In the Austin Texas - Scenario[2] datasets, Fuzzy OPF obtained the best results, where the values of the combinations of sigma and kmax were between [X: X; X:X] for sigma and [X:X; X:X] for Kmax.

This behavior leads to the conclusion that: 1 – When the parameters are close to $1 : 0$, the Fuzzy OPF has a performance similar to the Standard OPF, that is, in the worst case it gets results as good as those of the aforementioned classifier. 2 –

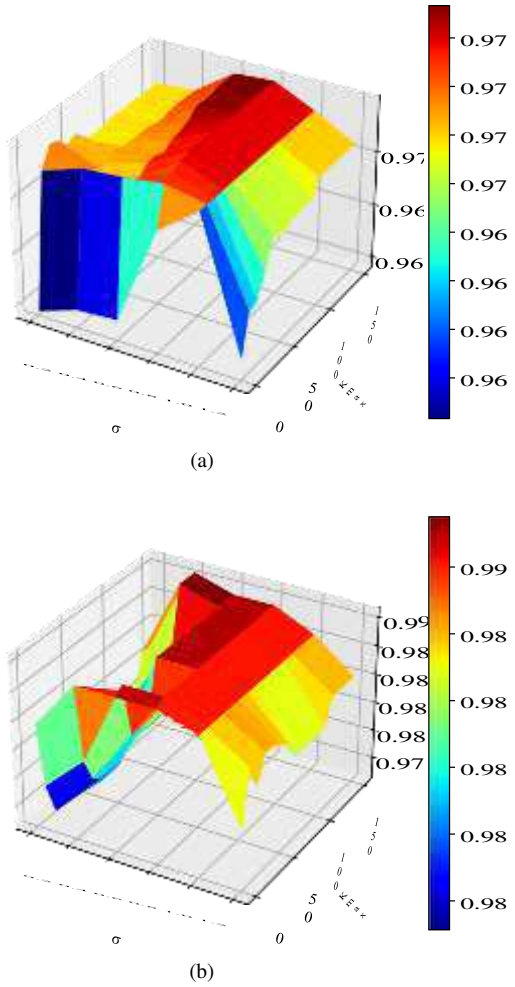


Fig. 5: Grid-search employed for Fuzzy OPF hyperparameter optimization considering Austin Texas-Scenario[1] images over: (*a*), Austin Texas-Scenario[2] (*b*),

Next, Fuzzy OPF can find better alternatives for scenarios that have a greater degree of complexity, such as in cases that have a reduced number of resources, which leads to an adequate selection of its hyperparameters.

IV. Conclusion

In this paper, we propose a new methodology for detecting intrusion in the IoT network. For that, we use Fuzzy OPF to propose a supervised model to classify IoT network traffic into intrusion and no intrusion packets.

The approach is evaluated on two datasets based on real data from intrusions in IOT networks, and performance compared with five different ML algorithms: SVM, KNN, LDA, Bayes, and OPF. From the results we observed that, among the techniques used, Fuzzy OPF and OPF are the best generals techniques considering an intrusion data detection task, asit obtained the best values in both datasets evaluated. The proposed model based on Fuzzy OPF reached 99.24% of true positive cases. Therefore, we conclude that Fuzzy OPF isa relevant classifier to generalize patterns of any threat that escapes the abnormality of traffic in an IoT network.

Acknowledgment

This work has been supported by the CAPES.

References

[1] H. M. Almohri, L. T. Watson, and D. Evans, "An attack-resilient architecture for the internet of things," *IEEE Transactions on Information Forensics and Security*, vol. 15, pp. 3940–3954, 2020.

[2] S. Sarkar, S. Chatterjee, and S. Misra, "Assessment of the suitability of fog computing in the context of internet of things," *IEEE Transactions on Cloud Computing*, vol. 6, no. 1, pp. 46–59, 2015.

[3] I. Butun, P. Österberg, and H. Song, "Security of the internet of things: Vulnerabilities, attacks, and countermeasures," *IEEE Communications Surveys & Tutorials*, vol. 22, no. 1, pp. 616–644, 2019.

[4] Z. Lv, L. Qiao, J. Li, and H. Song, "Deep learning enabled security issues in the internet of things," *IEEE Internet of Things Journal*, 2020.

[5] A. Khraisat and A. Alazab, "A critical review of intrusion detection systems in the internet of things: techniques, deployment strategy, validation strategy, attacks, public datasets and challenges," *Cybersecurity*, vol. 4, no. 1, pp. 1–27, 2021.

[6] T. Alladi, V. Chamola, B. Sikdar, and K.-K. R. Choo, "Consumer iot: Security vulnerability case studies and solutions," *IEEE Consumer Electronics Magazine*, vol. 9, no. 2, pp. 17–25, 2020.

[7] J. Yao, H. Chen, Z. Xu, J. Huang, J. Li, J. Jia, and H. Wu, "Development of a wearable electrical impedance tomographic sensor for gesture recognition with machine learning," *IEEE Journal of Biomedical and Health Informatics*, vol. 24, no. 6, pp. 1550–1556, 2020.

[8] R. W. de Souza, D. S. Silva, L. A. Passos, M. Roder, M. C. Santana, P. R. Pinheiro, and V. H. C. de Albuquerque, "Computer-assisted parkinson's disease diagnosis using fuzzy optimum- path forest and restricted boltzmann machines," *Computers in Biology and Medicine*, vol. 131, p. 104260, 2021.

[9] X. Huang, C. Xie, X. Fang, and L. Zhang, "Combining pixel- and object-based machine learning for identification of water-body types from urban high-resolution remote-sensing imagery," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 8, no. 5, pp. 2097–2110, 2015.

[10] E. P. Nugroho, T. Djatna, I. S. Sitanggang, A. Buono, and I. Hermadi, "A review of intrusion detection system in iot with machine learning approach: Current and future research," in *2020 6th International Conference on Science in Information Technology (ICSITech)*, 2020,pp. 138–143.

[11] K. A. da Costa, J. P. Papa, C. O. Lisboa, R. Munoz, and V. H. C. de Albuquerque, "Internet of things: A survey on machine learning-based intrusion detection approaches," *Computer Networks*, vol. 151, pp. 147–157, 2019.

[12] 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 Communications Surveys Tutorials*, vol. 22, no. 3, pp. 1646–1685, 2020.

[13] T. Saranya, S. Sridevi, C. Deisy, T. D. Chung, and M. Khan, "Performance analysis of machine learning algorithms in intrusion detection system: A review," *Procedia Computer Science*, vol. 171, pp. 1251–1260, 2020, third International Conference on Computing and Network Communications (CoCoNet'19).

[14] Z. Liu, N. Thapa, A. Shaver, K. Roy, X. Yuan, and S. Khorsandroo, "Anomaly detection on iot network intrusion using machine learning," in *2020 International Conference on Artificial Intelligence, Big Data, Computing and Data Communication Systems (icABCD)*, 2020, pp. 1–5.

[15] A. Shaver, Z. Liu, N. Thapa, K. Roy, B. Gokaraju, and X. Yuan, "Anomaly based intrusion detection for iotwith machine learning," in *2020 IEEE Applied Imagery Pattern Recognition Workshop (AIPR)*, 2020, pp. 1–6.

[16] M. A. Cheema, H. Khaliq Qureshi, C. Chrysostomou, and M. Lestas, "Utilizing blockchain for distributed machine learning based intrusion detection in internet of things," in *2020 16th International Conference on Distributed Computing in Sensor Systems (DCOSS)*, 2020, pp. 429–435.

[17] N. Ghosh, K. Maity, R. Paul, and S. Maity, "Outlier detection in sensor data using machine learning techniques for iot framework and wireless sensor networks: A brief study," in *2019 International Conference on Applied Machine Learning (ICAML)*, 2019, pp. 187–190.

[18] A. Vikram and Mohana, "Anomaly detection in network traffic using unsupervised machine learning approach," in *2020 5th International Conference on Communication and Electronics Systems (ICCES)*, 2020, pp. 476–479.

[19] R. R. Guimaraes, L. A. Passos, R. H. Filho, V. H. C. d. Albuquerque, J. J. P. C. Rodrigues, M. M. Komarov, and J. P. Papa, "Intelligent network security monitoring based on optimum-path forest clustering," *IEEE Network*, vol. 33, no. 2, pp. 126–131, 2019.

[20] Z. Chkirbene, S. Eltanbouly, M. Bashendy, N. AlNaimi, and A. Erbad, "Hybrid machine learning for network anomaly intrusion detection," in *2020 IEEE International Conference on Informatics, IoT, and Enabling Technologies (ICIoT)*, 2020, pp. 163–170.

[21] A. E. Ghazi and A. Moulay Rachid, "Machine learning and datamining methods for hybrid iot intrusion detection," in *2020 5th International Conference on Cloud Computing and Artificial Intelligence: Technologies and Applications (CloudTech)*, 2020, pp. 1–6.

[22] E. D. Alalade, "Intrusion detection system in smart home network using artificial immune system and extreme learning machine hybrid approach," in *2020 IEEE 6th World Forum on Internet of Things (WF-IoT)*, 2020, pp. 1–2.

[23] P. Maniriho, E. Niyigaba, Z. Bizimana, V. Twiringiyimana, L. J. Mahoro, and T. Ahmad, "Anomaly-based intrusion detection approach for iot networks using machine learning," in *2020 International Conference on Computer Engineering, Network, and Intelligent Multimedia (CENIM)*, 2020, pp. 303–308.

[24] J. Arshad, M. A. Azad, M. M. Abdeltaif, and K. Salah, "An intrusion detection framework for energy constrained iot devices," *Mechanical Systems and Signal Processing*, vol. 136, p. 106436, 2020.

[25] M. M. Hassan, M. R. Hassan, S. Huda, and V. H. C. de Albuquerque, "A robust deep-learning-enabled trust-boundary protection for adversarial industrial iot environment," *IEEE Internet of Things Journal*, vol. 8, no. 12, pp. 9611–9621, 2021.

[26] N. Magaia, R. Fonseca, K. Muhammad, A. H. F. N. Segundo, A. V. Lira Neto, and V. H. C. de Albuquerque, "Industrial internet-of-things security enhanced with deep learning approaches for smart cities," *IEEE Internet of Things Journal*, vol. 8, no. 8, pp. 6393–6405, 2021.

[27] S. S. Swarna Sugi and S. R. Ratna, "Investigation of machine learning techniques in intrusion detection system for iot network," in *2020 3rd International Conference on Intelligent Sustainable Systems (ICISS)*, 2020, pp. 1164–1167.

[28] O. Alkadi, N. Moustafa, B. Turnbull, and K.-K. R. Choo, "A deep blockchain framework-enabled collaborative intrusion detection for protecting iot and cloud networks," *IEEE Internet of Things Journal*, 2020.

[29] N. Ravi and S. M. Shalinie, "Semisupervised-learning-based security to detect and mitigate intrusions in iot network," *IEEE Internet of Things Journal*, vol. 7, no. 11, pp. 11 041–11 052, 2020.

[30] Z. Tian, C. Luo, J. Qiu, X. Du, and M. Guizani, "A distributed deep learning system for web attack detec- tion on edge devices," *IEEE Transactions on Industrial Informatics*, vol. 16, no. 3, pp. 1963–1971, 2019.

[31] R. R. Guimaraes, L. A. Passos, R. Holanda Filho, V. H. C. de Albuquerque, J. J. Rodrigues, M. M. Komarov, and J. P. Papa, "Intelligent network security monitoring based on optimum-path forest clustering,"*Ieee Network*, vol. 33, no. 2, pp. 126–131, 2018.

[32] A. L. Cristiani, D. D. Lieira, R. I. Meneguette, and H. A. Camargo, "A fuzzy intrusion detection systemfor identifying cyber-attacks on iot networks," in *2020 IEEE Latin-American Conference on Communications (LATINCOM)*. IEEE, 2020, pp. 1–6.

[33] N. Naik, R. Diao, and Q. Shen, "Dynamic fuzzy rule interpolation and its application to intrusion detection," *IEEE Transactions on Fuzzy Systems*, vol. 26, no. 4, pp. 1878–1892, 2017.

[34] S. Manimurugan, A.-q. Majdi, M. Mohmmed, C. Narmatha, and R. Varatharajan, "Intrusion detection in networks using crow search optimization algorithm with adaptive neuro-fuzzy inference system," *Microprocessors and Microsystems*, vol. 79, p. 103261, 2020.

[35] R. W. R. Souza, J. V. C. De Oliveira, L. A. Passos, W. Ding, J. P. Papa, and V. Albuquerque, "A novel approach for optimum-path forest classification using fuzzy logic," *IEEE Transactions on Fuzzy Systems*, pp. 1–1, 2019.

[36] L. A. P. Júnior, C. C. O. Ramos, D. Rodrigues, D. R.

Pereira, A. N. de Souza, K. A. P. da Costa, and J. P. Papa, "Unsupervised non-technical losses identification through optimum-path forest," *Electric Power Systems Research*, vol. 140, pp. 413–423, 2016.

[37] L. Passos, D. Jodas, L. Ribeiro, T. Moreira, and J. Papa, "Oˆ 2pf: Oversampling via optimum-path forest for breast cancer detection," in *2020 IEEE 33rd Interna- tional Symposium on Computer-Based Medical Systems (CBMS)*. IEEE, 2020, pp. 498–503.

[38] L. M. Rocha, F. A. M. Cappabianco, and A. X. Falcão, "Data clustering as an optimum-path forest problem with applications in image analysis," *International Journal of Imaging Systems and Technology*, vol. 19, no. 2, pp. 50–68, 2009. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/ima.20191

[39] C.-F. Lin and S.-D. Wang, "Fuzzy support vector machines," *IEEE Transactions on Neural Networks*, vol. 13, no. 2, pp. 464–471, 2002.

[40] [Online]. Available: https://dx.doi.org/10.21227/6nmw-w693

[41] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.