DDoS Intrusions Detection in Low Power SD-IoT Devices Leveraging Effective Machine Learning

Jehad Ali, *(Senior Member, IEEE),* Houbing Herbert Song, (*Fellow, IEEE),* Vandana Sharma, ((*Senior Member, IEEE),* Mahmoud Ahmad Al-Khasawneh , ((*Senior Member, IEEE),*

***Abstract*—Security and privacy are significant concerns in software-defined networking (SDN)-applied Internet of Things (IoT) environments, due to the proliferation of connected devices and the potential for cyberattacks. Hence, robust security mechanisms need to be developed, including authentication, encryption, and distributed denial of service (DDoS) attack detection, tailored to the constraints of low-power IoT devices. Selecting a suitable tiny machine learning (TinyML) algorithm for low-power IoT devices for DDoS attack detection involves considering various factors such as computational complexity, robustness in dealing with heterogeneous data, accuracy, and the specific constraints of the target IoT device. In this paper, we present a two-fold approach for the optimal TinyML algorithm selection leveraging the hybrid analytical network process (HANP). First, we make a comparative analysis (qualitative) of the machine learning algorithm in the context of suitability for TinyML in the domain of SD-IoT devices and generate the weights of suitability for TinyML applications in SD-IoT. Then we evaluate the performance of the machine learning algorithms and validate the results of the model to demonstrate the effectiveness of the proposed method. Finally, we see the effect of dimensionality reduction with respect to features and how it affects the precision, recall, accuracy, and F1 score. The results demonstrate the effectiveness of the scheme.**

***Index Terms*—Low power IoT, SDN, DDoS attacks, Machine learning, Decision making**

I. Introduction

The Internet of Things (IoT) makes a network consisting of sensors and devices including software programs utilizing the Internet for transmission of data. The inclusion of IoT into consumer electronics (CE) made a revolution in next-generation CE [1], [2]. The extensive proliferation of Internet IoT applications renders technology increasingly susceptible to attacks. As the service domains of IoT continue to expand continuously, security concerns remain prevalent. This is mainly due to the diverse and heterogeneous nature of networks employed in IoT, which incorporate both large as well as small devices [3], [4]. The small sensor devices in IoT, characterized by constrained computational power, computing resources, and storage capacity, pose challenges in implementing efficient protection methods and cryptographic algorithms for security. Moreover, the lack of privacy-preserving algorithms in small IoT devices leaves them vulnerable to exploitation by malicious actors, who leverage these vulnerabilities to enlist them as bots for launching attacks [5], [6].

Software-defined IoT (SD-IoT) is susceptible to distributed denial of service (DDoS) attacks due to vulnerabilities in underlying IoT devices such as its widespread connectivity [7]. Hence, legitimate users are not able to access the resources due to the high utilization of the server with attackers' illegitimate or attack requests. Hence various studies are proposed to detect and mitigate these attacks as illustrated in [8]–[22]. These approaches can be broadly categorized into two types. i.e. the one using traditional mechanisms to detect DDoS attacks [8]–[14] such as entropy-based methods, and the second type utilizes machine learning-based algorithms to detect DDoS attacks [15]–[22].

The entropy-based schemes mentioned in [8]–[14] for DDoS attack detection in SDN environments exhibit some limitations. First, the entropy-based schemes often rely on static threshold values to detect anomalies in network traffic entropy. However, these thresholds may not adapt well to dynamic changes in network behavior or varying traffic patterns. As a result, they may lead to high false positives or false negatives, diminishing the effectiveness of DDoS attack detection. In addition, the entropy-based strategies analyze traffic entropy at the packet or flow level without considering the broader context of network activities. This lack of context awareness may fail to distinguish between legitimate fluctuations in entropy and actual DDoS attacks. Moreover, without contextual information about the network topology, application behavior, or user activities, the accuracy of detection may be compromised. Furthermore, these entropy approaches may be sensitive to specific network characteristics, such as traffic volume, traffic type, or network topology. Variations in these characteristics across different network environments can impact the efficacy of entropy-based DDoS attack detection methods. Additionally, attackers may exploit these variations to evade detection by manipulating traffic patterns or exploiting vulnerabilities in the entropy calculation process.

The forecast of the microcontroller units (MCUs) at an exponential rate and its forecast (with more than 40 billion MCUs adopted until 2022) is shown in [23]. As the availability and capabilities of microcontrollers (MCUs) continue to

Jehad Ali is with the Department of AI Convergence Network, Ajou University, Suwon, South Korea (Email address:jehadali@ajou.ac.kr)

Prof Song is with the Department of Information Systems University of Maryland, Baltimore County (UMBC), 1000 Hilltop Circle, Baltimore, MD 21250 USA (Email address:songh@umbc.edu)

vandana Sharma Christ University, Delhi NCR Campus, India (Email address:vandana.juyal@gmail.com)

Mahmoud Ahmad Al-Khasawneh School of Computing, Skyline University College, University City Sharjah, 1797, Sharjah, UAE and Applied Science Research Center. Applied Science Private University, Amman, Jordan and Jadara University Research Center, Jadara University, Jordan (Email address:mahmoud@outlook.my)

Corresponding author: jehadali@ajou.ac.kr

increase and evolve, so does the development of tiny machine learning (TinyML) algorithms. With advancements in MCU technology, including higher processing speeds, lower power consumption, and increased memory capacities, developers can implement more complex machine-learning models directly on embedded devices [24]. TinyML offers significant advantages over typical machine learning algorithms for low-power IoT devices due to its lightweight and energy-efficient nature. Unlike traditional algorithms as illustrated in [15]–[21], which often require substantial computational resources and memory, TinyML models are designed to operate efficiently on resource-constrained devices with minimal energy consumption, which enables local processing of sensor data on the IoT device, reducing the need for frequent data transmission to centralized servers and minimizing latency. Additionally, TinyML's compact model sizes make them suitable for deployment on devices with limited storage capacity, allowing for on-device training and inference without compromising performance. Moreover, TinyML's ability to leverage hardware accelerators, such as microcontrollers and specialized processors, further optimizes energy efficiency and computational speed, making it an ideal choice for a wide range of IoT applications where power consumption and resource constraints are critical considerations [25]–[30].

Hence, in this paper, we propose a novel method to qualitatively analyze the suitability of the machine learning algorithms for TinyML implementation and then validate it through evaluations in Software-Defined Internet-of-Things (SD-IoT) DDoS attack detection. To the best of our knowledge, our paper proposes a novel hybrid algorithm with qualitative and quantitative decision-making mechanisms for the implementation of TinyML in SD-IoT. The main contributions of our proposed method are as follows.

- First we formulate the problem of TinyML algorithm suitability for DDoS attack detection within the low-power IoT and identify the criteria for machine learning algorithms applicability in TinyML.
- We formulate the problem for TinyML alternative algorithm selection and ranking with a hybrid analytical network process (HANP). Moreover, we demonstrate it with an algorithm (HANP).
- To validate the HANP results' effectiveness, we evaluate the machine learning algorithms in terms of various performance metrics.
- We also see the effect of feature reduction and propose an algorithm for optimal principal component selection from the given features

The rest of our paper is organized as follows. In Section 2 we present the problem formulation of our proposed strategy. In section 3, we describe our proposed hybrid analytical network process technique to get the weights for an efficient machine learning algorithm. Section 4 evaluates the results and discusses them. In section 5 the paper concludes with future research directions.

II. Problem Formulation

Several machine learning algorithms can be applied for tiny machine learning in DDoS attack detection for low-power IoT devices such as support vector machines (SVM), Random forests (RF), Decision trees, and Naive Bayes. However, each algorithm has its strengths and weaknesses regarding the criteria shown in Table 1, and the selection should be based on the specific requirements and constraints of the IoT deployment regarding TinyML. Hence, in this paper a two fold approach is utilized for implementing the TinyML algorithms for DDoS attacks detection in SD-IoT network. We formulate the problem with HANP. First, the SD-IoT is denoted as graph $G = (V, E)$. The IoT sensor devices are denoted as $SD_1, SD_2, SD_3, SD_4, ..., SD_n$. The sensor devices in SD-IoT are part of the data plane as shown in Figure 1. Moreover, the criteria are denoted with Eq. (1), and the machine learning algorithms are shown with Eq. (2). The ML_1 up to MLK shows the list of machine learning algorithms. Leveraging the HANP we will first identify and illustrate the suitability of criteria features in TinyML for SD-IoT. Then, we rank them through the HANP i.e. hybrid of analytical hierarchy process (AHP) and ANP. First, our goal is to find the weights and ranking of the suitability of the algorithms for TinyML SD-IoT environment and then validate the efficacy through simulations.

$$CT = \{CT_1, CT_2, CT_3,\ ...,\ CT_n\} \tag{1}$$

$$ML = \{ML_1, ML_2, ML_3,\ ...,\ ML_K\} \tag{2}$$

TABLE I: Criteria for TinyML schemes (machine learning algorithm evaluation)

Name of criteria	Abbreviation	Notation
Resource efficiency	RE	CT1
Model complexity	MC	CT2
Training inference speed	TIS	CT3
Accuracy and detection performance	ADP	CT4
Robustness to data distribution shifts	RDDS	CT5
Dimensionality reduction for big data	DRBD	CT6

III. Proposed Approach and Framework

The overall architecture of our proposed method is shown in Figure 1. Figure 1 shows an SD-IoT architecture employing different modules upon the control plane. The HANP modules rank the machine learning algorithms under consideration for low-power IoT networks in TinyML for DDoS attacks detection. Herein, the HANP module considers the low-power IoT network and suitability of the machine learning algorithms with criteria described in the next subsection. Moreover, we get the data analytics utilizing a centralized SDN model in IoT from the underlying SD-IoT network having data plane and control plane. The data plane consists of low-power IoT sensors/devices and the control plane consists of SDN controller for obtaining data analytics and applications running. The data plane communicates with the controller through a southbound application programming interface. While the control plane interacts with the modules on the control plane via the northbound application programming interface. The data analytics are obtained from the SD-IoT network. Furthermore, feature engineering is done on the dataset with principal component

analysis (PCA) to get the smaller dimension dataset with reduced features. However, the variance should be kept high to consider the DDoS attacks detection features, even with a smaller number of features. Our proposed algorithm for feature selection in the next subsections ensures that the variance is kept high. Then the algorithms are evaluated for accuracy, precision, recall and F1 score to check if they are producing the results according to the ranking generated with HANP.

A. Criteria for ranking the algorithms

In this subsection, we describe the criteria regarding the applicability of machine learning algorithms in TinyML low-power IoT devices. When selecting a machine learning algorithm for tiny machine learning in DDoS attack detection for low-power IoT devices, several criteria are significant to be considered. These criteria contribute int the low-power IoT devices while implementing it for TinyML considering their low computational and battery power. Here are some key factors during the selection of an algorithm.

1) Resource Efficiency (RE): Given the constrained computational resources and low power capabilities of IoT devices, the chosen algorithm should be lightweight and computationally efficient. It should be able to perform adequately on devices with limited memory, processing power, and energy resources.
2) Model Complexity (MC): The algorithm should be able to create models with low complexity to fit within the constraints of IoT devices. Complex models may require more memory and processing power, which can be prohibitive for low-power IoT devices.
3) Training and Inference Speed (TIS): The algorithm should have fast training and inference times to operate efficiently on IoT devices. Rapid training enables quick model updates, while fast inference ensures timely detection of DDoS attacks without significant latency.
4) Accuracy and Detection Performance (ADP): The efficiency is critical, however, the algorithm should still provide adequate detection performance and accuracy in identifying DDoS attacks. It is essential to strike a balance between resource efficiency and detection effectiveness.

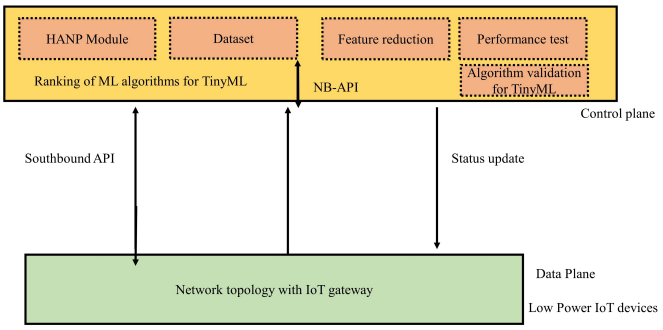

Fig. 1: Proposed SD-IoT architecture for TinyML algorithm selection in DDoS attacks detection

5) Robustness to Data Distribution Shifts (RDDS): IoT environments may exhibit dynamic and heterogeneous data distributions due to changes in network conditions or IoT device deployments. The algorithm should be robust to such shifts and maintain its performance across different scenarios.
6) Dimensionality reduction of big data (DRBD): This feature concerns the algorithm's performance in dealing with data in low dimensions as compared to the original features in the dataset. This is significant in dealing with huge amounts of data hence the algorithm should generate accurate results with a smaller number of features from the dataset. Hence, in our performance evaluation results, we propose a method to select the features (reduced features) from the original number of features. Moreover, we evaluate the performance with fewer features also for the machine learning algorithms.

In the next subsection, we explain the HANP to rank the alternative algorithms regarding the criteria metrics. Moreover, we also evaluate it through simulations.

B. Hybrid Analytical Network Process

To select a machine learning algorithm using Hybrid Analytic Network Process (HANP), incorporating both ANP and AHP, for the given alternatives (Random Forest, Decision Trees, SVM) and criteria (RE, MC, TIS, ADP, RDDS, DRBD), we follow these steps:

1) Define the goal and criteria hierarchy: First, we identify the overarching goal, which is to select the significant machine learning algorithm. Then, establish a hierarchy with the goal at the top, followed by criteria (RE, MC, TIS, ADP, RDDS, and DRBD then the alternatives (Random Forest, Decision Trees, SVM, and KNN) at the bottom. Figure 2 describes the hierarchical clusters of ANP with criteria and alternatives.
2) Pairwise Comparison in AHP: Within the AHP framework, we conduct pairwise comparisons between criteria to determine their relative importance. The pairwise comparisons are made between each criterion to establish their relative priority. For example, compare RE against MC, TIS, ADP, RDDS, and DRBD. Moreover, we use the AHP scale (e.g., 1 to 9) to assign relative importance values to each pair of criteria based on their perceived significance (the precedence of one feature over another regarding applicability in TinML for SD-IoT. During this process, we assign a priority to the machine learning algorithm working well with the reduced feature set. Initially, we suppose that DT works well with reduced features while applying the dimensionality reduction. Then, we create alternatives ranking through ANP.
3) Compute Criteria Weights in AHP: Calculate the normalized weights for each criterion based on the pairwise comparison judgments using AHP's eigenvalue method or other suitable methods.
4) Pairwise Comparison in ANP: In addition, with the ANP framework [31], we conduct pairwise comparisons

between alternatives for each criterion to determine their performance relative to one another. Compare each pair of alternatives (Random Forest vs. Decision Trees, Random Forest vs. SVM, Decision Trees vs. SVM) for each criterion (RE, MC, TIS, ADP, RDDS, DRBD). Herein, we also use the scale (e.g., 1 to 9) to assign relative performance values to each pair of alternatives for each criterion. The process is done through creating a matrix and assigning the values from 1 to 9 based upon the alternative (machine learning algorithm) significance regarding the criteria for selection.

5) Compute Performance Scores in ANP: Similarly, we calculate the normalized performance scores for each alternative for each criterion based on the pairwise comparison judgments using ANP's supermatrix approach. These scores represent the relative performance of each alternative for each criterion.
6) Aggregation of criteria and alternatives: Then, we aggregate the criteria weights from AHP and performance scores from ANP to obtain an overall score for each alternative. Moreover, we multiply the criteria weights by the corresponding performance scores for each alternative to obtain weighted performance scores. The weighted performance scores across all criteria to compute the overall score for each alternative.
7) Ranking of Alternatives: The final step is ranking the alternatives based on their overall scores, with higher scores indicating better suitability in TinyML applications. The alternative with the highest overall score is considered the most suitable machine learning algorithm for the given decision scenario (SD-IoT network DDoS attacks detection with TinyML for low-power IoT sensors). The step-by-step process is shown in algorithm 1. Algorithm 1 considers the AHP to rank the criteria and ANP for creating weights for alternatives.

Herein, we describe the comparison matrices and the incorporation of values in these matrices. Moreover, the normalization, eigenvectors and computation of the consistency index and ratio index using mathematical equations.

- Eq. (3) shows a sample comparison matrix in ANP. The Eq. (4) identify the comparison matrix with some example values, which reveals the relative significance of one machine learning algorithm (ML) over another concerning the criteria CT defined in Eq. (1). The a shows the values which will be incorporated from Table II. The quantitative value of the relative significance of one ML over another ML is derived from a 9-level scale, which is shown in Table II, where 1 shows an equal importance level and 9 shows the extreme significance of one ML algorithm compared to other algorithms. Similarly, 3 shows that an ML is moderately more significant than the other ML regarding criteria. These values are given in Table II, and are incorporated in Eq. (4) for all ML concerning each criteria CT. In addition, as described these comparison matrices will be used by AHP for criteria and by ANP for alternatives.

TABLE II: Scale of importance

Scale	Explanation
1	The ML have an equal value of importance
2	One ML is equally leading to moderately good than other ML
3	An ML is moderately slight dominant from the other ML
4	The ML is significantly more crucial than other ML
5	It indicates that an ML is significantly dominant (more)
6	It reveals that a ML is remarkably important with another
7	Remarkably dominant (slightly greater) of a ML from other ML
8	One of a ML is remarkable to more significant with respect to other
9	An algorithm is excessively more dominant concerning other ML

$$M = \begin{bmatrix} & ML_1 & ML_2 & ML_3 & \rightarrow & ML_n \\ ML_1 & 1 & a_{12} & a_{13} & \rightarrow & a_{1n} \\ ML_2 & \frac{1}{a_{12}} & 1 & a_{23} & \rightarrow & a_{2n} \\ ML_3 & \frac{1}{a_{13}} & \frac{1}{a_{23}} & 1 & \rightarrow & a_{3n} \\ \downarrow & \downarrow & \downarrow & \downarrow & 1 & \downarrow \\ ML_n & \frac{1}{a_{1n}} & \frac{1}{a_{2n}} & \frac{1}{a_{3n}} & \rightarrow & 1 \end{bmatrix} \quad (3)$$

$$M = \begin{bmatrix} & ML_1 & ML_2 & ML_3 & ML_4 \\ ML_1 & 1 & \frac{1}{3} & \frac{1}{3} & 3 \\ ML_2 & 3 & 1 & 1 & 6 \\ ML_3 & 9 & 1 & 1 & 3 \\ ML_4 & \frac{1}{9} & \frac{1}{6} & \frac{1}{3} & 1 \end{bmatrix} \quad (4)$$

- The normalization process of the comparison matrix given in (4) is performed according to Eq. (5) to obtain local weights of (ML) and the criteria parameters (CT) in the shape of eigenvectors as denoted in Eq. (6). Equation. (6) for the eigenvectors shows the priorities (weights) of one ML over other ML. Moreover, to prove if the judgments are precise and accurate according to the values provided in the comparison matrix, another parameter i.e. consistency index (CI) is calculated. The final value for $CI \leq 0.1$ indicates the preciseness of the pairwise values used in the judgments for one ML compared to another ML. The prerequisite Y_j is computed for CI i.e. the consistency measure denoted with Y_j as well as λ_{max} as denoted in Eq. (7) and Eq. (8). The Eq. (8) identify the calculation process for λ_{max} as well as consistency measure procedure. In addition, the (RI) value is inserted according to the criteria number in Eq. (9). Further it is inserted in Eq. (10). The final value for CR is shown in Eq. (10).

$$M_\alpha = \begin{bmatrix} \frac{a_{11}}{\sum_{i=1}^{n} a_{i1}} & \cdots & \frac{a_{1n}}{\sum_{i=1}^{n} a_{in}} \\ \vdots & \ddots & \vdots \\ \frac{a_{n1}}{\sum_{i=1}^{n} a_{i1}} & \cdots & \frac{a_{nn}}{\sum_{i=1}^{n} a_{in}} \end{bmatrix} \quad (5)$$

$$X_i = \frac{1}{n} \sum_{j=1}^{n} a_{ij} \quad (6)$$

$$Y_j = \frac{M_j * X}{x_i} \quad (7)$$

$$\lambda_{max} = \frac{1}{n} \sum_{j=1}^{n} Y_j \quad (8)$$

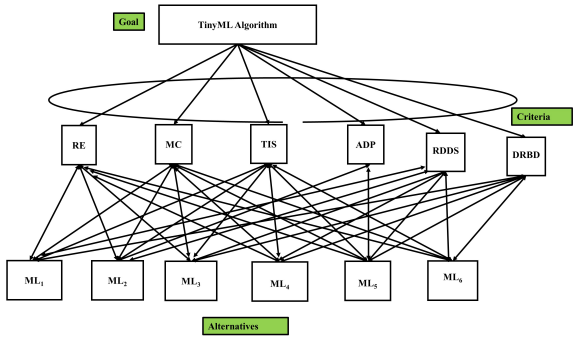

Fig. 2: Analytical network process for ranking the alternative machine learning algorithms

$$CI = \frac{(\lambda_{max} - n)}{(n-1)} \tag{9}$$

$$CR = \frac{RI}{CI} \tag{10}$$

- Next, the eigenvectors as denoted with (6) for each ML are arranged in unweighted super-matrix form that gives a local priority of one ML over another ML such as SVM and KNN or SVM and random forest, etc.
- Finally, a limit super-matrix of converged values having stable weights is calculated via taking the power of the weighted super-matrix until the convergence of its values. The convergent matrix shows the priority order of the ML algorithms for application in TinyML.

The ranking produces weights applying HANP algorithm 1, i.e., high weight for DT, then RF, SVM and KNN.

Algorithm 1 Hybrid Analytic Network Process (HANP) for Machine Learning Algorithm Selection

Require: Define criteria, alternatives (in Eq.(1), (2)
Require: Criteria: RE, MC, TIS, ADP, RDDS, DRBD
Require: Alternatives: SVM, RF, DT, KNN
Ensure: Ranked list of machine learning algorithms
Ensure: Ranked list is generated for alternatives

Define the Goal and Criteria Hierarchy:
Create a hierarchical structure with the goal at the top, followed by criteria, and then alternatives as shown in Fig. 2.
Pairwise comparison in AHP:
Conduct pairwise comparisons between criteria to determine their relative importance using the Analytic Hierarchy Process (AHP).
Compute criteria weights in AHP:
- Calculate the normalized weights for each criterion based on the pairwise comparison judgments obtained from AHP.
Pairwise comparison in ANP:
- For each criterion, conduct pairwise comparisons between alternatives to determine their relative performance using the Analytic Network Process (ANP) as shown in (4).
Compute performance scores in ANP:
Calculate the normalized performance scores for each alternative for each criterion based on the pairwise comparison judgments obtained from ANP.
Aggregation of criteria and alternatives:
Aggregate the criteria weights from AHP and performance scores from ANP to obtain an overall score for each alternative.
Multiply the criteria weights by the corresponding performance scores for each alternative to obtain weighted performance scores.
Sum the weighted performance scores across all criteria to compute the overall score for each alternative.
Ranking of alternatives:
Generate ranking, the alternative with the highest overall score is considered the most suitable.
return Ranked list (converged weights) of machine learning algorithms based on their suitability for the given criteria.

IV. Results and Discussion

In this section, we discuss the comparison of the machine-learning models evaluated through algorithm 1. We discuss various evaluation metrics such as precision, accuracy, F1 score, and recall. Experiments were conducted with SDN Ryu controller [32]. Moreover, we discuss the dataset used for the evaluation of the performance metrics and the feature reduction procedure through the statistical PCA method. In addition, we also propose an algorithm for the reduction of features while maintaining a high variance in the original features from the dataset. Overall, we validate the efficacy of the suggested method.

A. Dataset

We perform the experiments on the IOTID20 dataset [33] because of the features from the IoT environment. Moreover, it contains novel attack features for IoT networks. The IoTID20 dataset encompasses various types of IoT attacks, such as DDoS, DoS, Mirai, and ARP Spoofing, alongside normal (benign) traffic. It is gathered from smart home IoT ecosystems, which commonly integrate diverse interconnected components, including artificial intelligence speakers (e.g., SKTNGU), Wi-Fi cameras (e.g., EZVIZ), laptops, smartphones, tablets, and wireless access points (Wi-Fi). Moreover, in this dataset, the cameras and artificial intelligence speakers serve as the IoT victim equipment, while the other devices function as the attacking equipment. We consider this dataset because it contains attacks with respect to diverse IoT networks.

B. Features/Dimensional reduction

IoT environments generate large volumes of data from various sources, leading to high-dimensional feature spaces. With low-power IoT devices employing TinyML working on high dimensional data is not feasible. Hence, we employ PCA to reduce the dimensionality of the data by transforming it into a lower-dimensional space while preserving the most

important information. Moreover, in IoT environments, some features may be highly correlated with each other, leading to multicollinearity. PCA addresses multicollinearity by transforming the original features into a set of orthogonal principal components. The orthogonal transformation helps in removing redundant information and improving the robustness of DDoS attacks detection models against correlated features.

The authors illustrated in [34] the dimensionality reduction from the dataset of the intrusion detection. To reduce the features from the original dataset the formula is denoted in Eq. (11). Herein, O shows the features present in an original set from data, whereas F_m denote the features computed after the application of dimensionality reduction leveraging PCA. Herein, it is significant to mention that the PCA computes the correlation among input dataset and denotes it using minimal features possessing a high variance. The R shows the ratio of F_m with O. The smaller value for R results in getting a maximum feature reduction.

$$R = \frac{F_m}{O} \tag{11}$$

However, via lowering (reducing) the value for O the variance of the original dataset also reduces. Consequently, we must choose such a value with respect to F_m which indicates and identifies the maximum variance in the DDoS attacks dataset.

C. Finding an optimal value for F_m

Leveraging PCA for the reduction of features in conjunction with the machine learning models for performance analysis, we reduce the original large features from O to F_m. i.e. F_m<O. Here $O_{(i)}$ shows the original number from features and the $O_{(i)}$ approx indicates the projected number for features. Hence, the squared projection error is shown in Equation (12).

$$\frac{1}{n}\sum_{i=1}^{n}||O^{(i)} - O^{(i)}_{approx}||^2 \tag{12}$$

In addition, the total variance of the data is indicated in (13)

$$\frac{1}{n}\sum_{i=1}^{n}||O^{(i)}||^2 \tag{13}$$

choose F_m by keeping the higher percentage for variance in the given dataset. Algorithm 2 shows the process of reducing the features while maintaining the higher ratio for variance in the dataset.

D. Discussion of Results

The precision metric is used to evaluate the quality of the machine learning models i.e. how many DDoS intrusions the model is predicting have a small false positive rate. Table III shows the precision for the machine learning algorithms i.e. DT, RF, SVM, and KNN. Table III indicates the lower false positives in DT model having 99.98%. Figure 3 shows the precision results for the machine learning algorithms. The dimensionality reduction results show a precision value of 99.50%. For the reduced features we have used algorithm 2 to keep the variance as high as possible in the original features representation with reduced principal components. The percentage results for precision with reduced features are slightly lower than the original features. The SVM model generated the lowest results for precision showing the SVM incompatibility with large datasets. The PCA computed the 10 most dominant features using the PCA algorithm we proposed for maintaining a high variance from the dataset. Hence, instead of the original 80 features from the dataset, the reduced features results are based upon the 10 most dominant features. Although there is a negligible reduction in the results with feature reduction as shown in Figure 3, however, for the low-power IoT devices with TinyML algorithms this contributes to and enhances the performance of the sensor devices with low computing and power capabilities.

Algorithm 2 PCA to calculate the value for reduced features F_m

1: F shows features (having reduce dimension)
2: m denotes total number for features having a reduced dimension
3: F_m is the ideal value for the principal (reduced feature number) to compute.
4: for $F_m = 1:1:n$;
5: If
6: $\frac{\frac{1}{n}\sum_{i=1}^{n}||O^{(i)} - O^{(i)}_{approx}||^2}{\frac{1}{n}\sum_{i=1}^{n}||O^{(i)}||^2} \leq 0.01$
7: Compute Ureduce, $F_{(1)}$, $F_{(2)}$, $F_{(3)}$,...,$F_{(m)}$ and $O_{(1)approx}$, $O_{(2)approx}$, $O_{(3)approx}$,..... $O_{(n)approx}$.
8: Print F_m
9: End
10: End

The results for recall comparing DT, SVM, RF, and KNN are shown in Table IV. Table IV shows the DT performs well generating 99.99% recall. Figure 4 indicates that the algorithms i.e. DT exhibit the highest recall rate of 99.99%. It also shows that DT has the highest ability to correctly identify true positives (instances of DDoS attacks in SD-IoT) among all positive instances in the given dataset. Similarly, DT offers the advantage of interpretability, allowing for easy understanding of the decision-making process. However, it is essential to consider the trade-offs between recall and false positives, especially in DDoS attacks where minimizing false positives is crucial. RF, which has a slightly lower recall rate but offers better generalization and robustness. Hence, RF ranked second by HANP can also be an alternate suitable choice for DDoS detection, considering its ability to handle diverse datasets and mitigate overfitting. Moreover, the use of RF allows for the aggregation of multiple decision trees, reducing the risk of overfitting and improving overall performance, making it a promising option for DDoS detection with low false positives and diverse data.

TABLE III: Precision

Dimensional Reduction	DT	RF	SVM	KNN
Orignal features	99.98	99.94	99.6	99.8
Reduced features	99.5	99.4	99.2	99

TABLE IV: Recall

Dimensional Reduction	DT	RF	SVM	KNN
Orignal features	99.99	99.96	99.7	99.8
Reduced features	99.6	99.5	99.3	99.2

TABLE V: Accuracy

Dimensional Reduction	RF	DT	SVM	KNN
Orignal features	99.88	99.82	99.44	99.68
Reduced features	99.48	99.77	99.2	99.6

TABLE VI: F1 score

Dimensional Reduction	DT	RF	SVM	KNN
Orignal features	99.97	99.5	99.5	99.6
Reduced features	99.5	99.3	99.2	98.49

The accuracy results are shown in Table V. Figure 5 show that RF has the highest accuracy of 99.88%, followed by DT algorithm with an accuracy of 99.82%. Moreover, SVM showed an accuracy of 99.44%, and KNN with an accuracy of 99.68%. The RF and DT performed well in DDoS attack detection for SD-IoT, as indicated by their high accuracy rates. The performance of RF and DT can be linked with their ability to handle complex datasets and capture nonlinear relationships between features effectively. The RF leverages ensemble learning to build multiple decision trees, which helps in mitigate the overfitting problem and improves generalization. In addition, the SVM shows a lower accuracy compared to RF and DT. SVM can find optimal hyperplanes for separating different classes in high-dimensional spaces. Similarly, the KNN achieved a relatively high accuracy as well. However, it is important to note that KNN's performance heavily depends on the choice of distance metric and the value of clusters (number). Therefore, our results (from Figure 5) suggest that ensemble methods like RF and DT are well-suited for DDoS attack detection in SD-IoT environments due to their robustness, scalability, and ability to handle diverse datasets of the SD-IoT network.

Table VI shows the results for F1 score. Figure 6 shows that DT achieved the highest F1 score of 99.97%, RF, and KNN also demonstrated strong performance with F1 scores of 99.50% and 99.60% respectively. SVM achieved a slightly lower F1 score of 99.50%. The high F1 score of DT indicates its ability to achieve both high precision and high recall simultaneously. This suggests that DT is effective in accurately identifying DDoS attacks while minimizing false positives and false negatives. The RF and KNN also demonstrated strong F1 scores, indicating their ability to balance precision and recall effectively. Moreover, SVM achieved a slightly lower F1 score compared to DT, RF, and KNN. Hence, the SVM is known for its effectiveness in separating classes in high-dimensional spaces, its performance in this scenario may have been affected by the complexity of the dataset or the choice of kernel function. Figure 6 shows that the DT is more suitable for DDoS attack detection in SD-IoT environments, offering high accuracy, precision, recall, and F1 score.

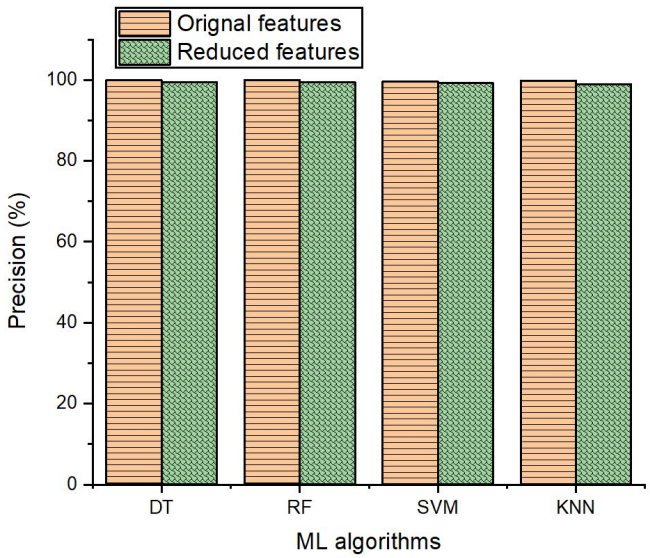

Fig. 3: Precision evaluation of the machine learning (ML) algorithms

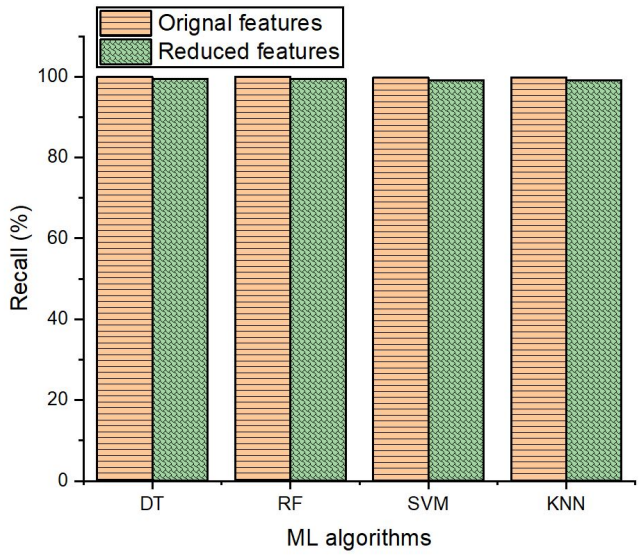

Fig. 4: Recall comparative analysis for the machine learning schemes

Finally, Figure 7 shows the effect of reducing the features i.e. the principal components (PCs), the average accuracy even with the reduced features is shown as 99.52%. However, there is a significant reduction in the number of features. Herein, we show it just for a small number of features to show the effect that accuracy is not much affected although the features were reduced. This is due to the PCA algorithm we proposed which maintains a high variance and chooses the PCs that are significant in DDoS attack detection.

V. Conclusion

Detection of DDoS attacks is challenging with low power IoT network. Hence TinyML helps to improve the computational capability of the resource-constrained IoT network.

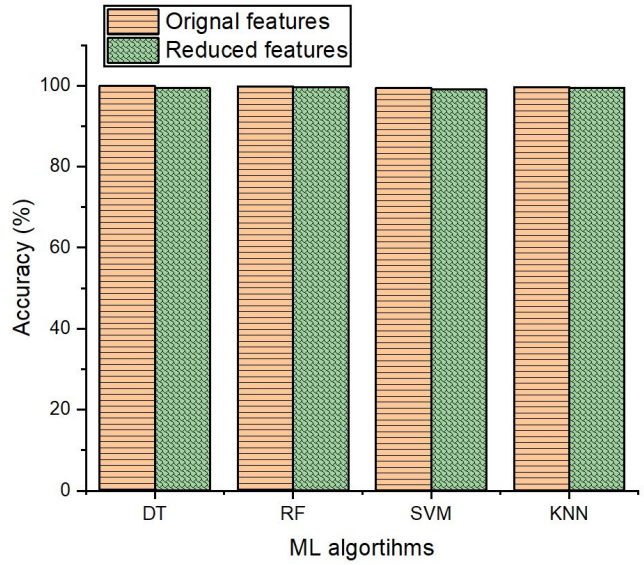

Fig. 5: Accuracy analysis and comparison

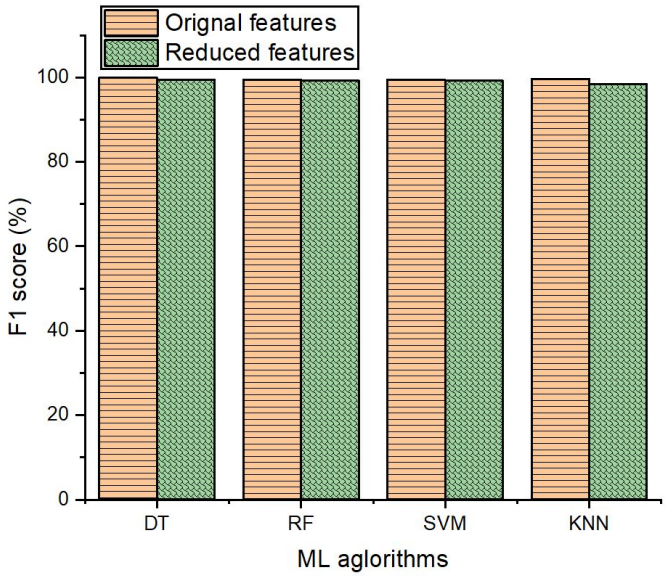

Fig. 6: F1 score evaluation of machine learning algorithms

However, with several machine learning schemes the implementation of the algorithms for TinyML in IoT network is challenging. In this paper, we proposed a hybrid method for ranking the machine learning algorithms regarding their applicability in SD-IoT for TinyML. To achieve this first, we have identified the criteria for TinyML algorithms in SD-IoT network. Then, we performed a qualitative analysis using HANP model to rank and weight the machine learning algorithms regarding their implementation suitability in TinyML for SD-IoT. Next, we have evaluated the machine learning algorithms regarding the precision, recall, accuracy, and F1 score in SD-IoT. The evaluation of the machine learning algorithms is performed with two kinds of features i.e. the first one selected for DDoS attack detection in SD-IoT and the second one with reduced features leveraging PCA. The results

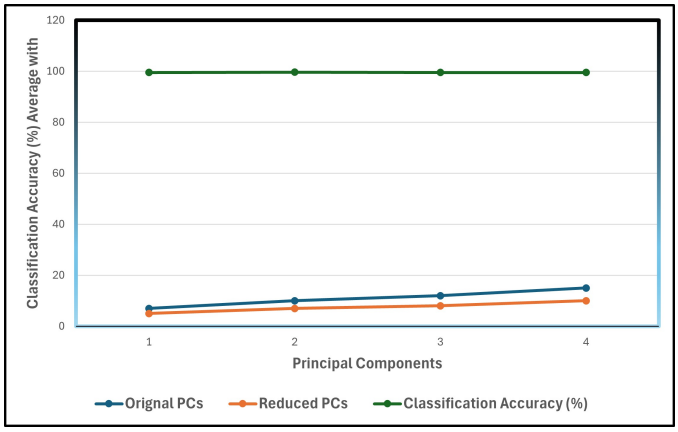

Fig. 7: Accuracy percentage with reduced features (PCs)

show that the algorithm ranked and weighted high with HANP generates improved results as compared to other algorithms, which validates the efficacy and effectiveness of the suggested methodology for DDoS attack detection leveraging tinyML in SD-IoT. Finally, we show that the accuracy with reduced PCs is also high due to the suggested PCA algorithm, which maintains significant variance in the dataset while selecting the PCs for DDoS attack detection.

The limitations of the work include testing it with more parameters for SD-IoT. Our future works focus on other statistical feature reduction methods enhancement for DDoS attack detection in low-power SD-IoT in the production environment.

ACKNOWLEDGMENT

This work was supported partially by the BK21 FOUR program of the National Research Foundation of Korea funded by the Ministry of Education (NRF5199991514504)

REFERENCES

[1] D. Javeed, M. S. Saeed, I. Ahmad, P. Kumar, A. Jolfaei and M. Tahir, "An Intelligent Intrusion Detection System for Smart Consumer Electronics Network," IEEE Transactions on Consumer Electronics, vol. 69, no. 4, pp. 906-913, Nov. 2023, doi: 10.1109/TCE.2023.3277856.

[2] Wu, C. K., Cheng, C. T., Uwate, Y., Chen, G., Mumtaz, S., Tsang, K. F, "State-of-the-art and research opportunities for next-generation consumer electronics," IEEE Transactions on Consumer Electronics, 2022, vol. 69, no. 4, pp. 937-948.

[3] M. Yamauchi, Y. Ohsita, M. Murata, K. Ueda and Y. Kato, "Anomaly detection in smart home operation from user behaviors and home conditions", IEEE Transactions on Consumer Electronics, vol. 66, no. 2, pp. 183-192, May 2020.

[4] Jisi, Chandroth, Byeong-hee Roh, and Jehad Ali. "Reliable paths prediction with intelligent data plane monitoring enabled reinforcement learning in SD-IoT." Journal of King Saud University-Computer and Information Sciences 36, no. 3 (2024): 102006.

[5] Kumari, P., & Jain, A. K. (2023). A comprehensive study of DDoS attacks over IoT network and their countermeasures. Computers & Security, 127, 103096.

[6] Ouhssini, M., Afdel, K., Agherrabi, E., Akouhar, M., & Abarda, A. (2024). DeepDefend: A comprehensive framework for DDoS attack detection and prevention in cloud computing. Journal of King Saud University-Computer and Information Sciences, 101938.

[7] M. Yamauchi, Y. Ohsita, M. Murata, K. Ueda and Y. Kato, "Anomaly detection in smart home operation from user behaviors and home conditions", IEEE Trans. Consum. Electron., vol. 66, no. 2, pp. 183-192, May 2020.

[8] Agrawal, A., Baniya, P., Gupta, B. B., Chaturvedi, S., Singh, G. K., & Yadav, D. (2023, December). A Review of Detecting DDoS Attacks Based on Entropy Computation. In 2023 12th International Conference on System Modeling & Advancement in Research Trends (SMART) (pp. 146-153). IEEE.

[9] Yu, S., Zhang, J., Liu, J., Zhang, X., Li, Y., & Xu, T. (2021). A cooperative DDoS attack detection scheme based on entropy and ensemble learning in SDN. EURASIP Journal on Wireless Communications and Networking, 2021(1), 90.

[10] Bhayo, J., Hameed, S., & Shah, S. A. (2020). An efficient counter-based DDoS attack detection framework leveraging software defined IoT (SD-IoT). IEEE Access, 8, 221612-221631.

[11] D. Tang, S. Wang, S. Zhang, Z. Qin, W. Liang and S. Xiao, "Real-Time Monitoring and Mitigation of SDoS Attacks Using the SDN and New Metrics," in IEEE Transactions on Cognitive Communications and Networking, vol. 9, no. 6, pp. 1721-1733, Dec. 2023

[12] Ahalawat, A., Babu, K. S., Turuk, A. K., & Patel, S. (2022). A low-rate DDoS detection and mitigation for SDN using Renyi Entropy with Packet Drop. Journal of Information Security and Applications, 68, 103212.

[13] Kishansmaran Puranik, Kirankumar Patil, Guruprasad Ghaligi, Rajath Jannu, Sangamesh Patil, D. G. Narayan, Amit Kachavimath, "A Two-level DDoS attack Detection using Entropy and Machine Learning in SDN", 2023 3rd International Conference on Intelligent Technologies (CONIT), pp.1-7, 2023.

[14] Z. Hemmati, G. Mirjalily and Z. Mohtajollah, "Entropy-based DDoS Attack Detection in SDN using Dynamic Threshold," 2021 7th International Conference on Signal Processing and Intelligent Systems (ICSPIS), Tehran, Iran, Islamic Republic of, 2021, pp. 1-5, doi: 10.1109/ICSPIS54653.2021.9729355.

[15] Su, Yinghao, Dapeng Xiong, Kechang Qian, and Yu Wang. "A Comprehensive Survey of Distributed Denial of Service Detection and Mitigation Technologies in Software-Defined Network." Electronics 13, no. 4 (2024): 807.

[16] Abid, Yawar Abbas, Jinsong Wu, Guangquan Xu, Shihui Fu, and Muhammad Waqas. "Multi-Level Deep Neural Network for Distributed Denial-of-Service Attack Detection and Classification in Software-Defined Networking Supported Internet of Things Networks." IEEE Internet of Things Journal (2024).

[17] Bhayo, Jalal, Syed Attique Shah, Sufian Hameed, Awais Ahmed, Jamal Nasir, and Dirk Draheim. "Towards a machine learning-based framework for DDOS attack detection in software-defined IoT (SD-IoT) networks." Engineering Applications of Artificial Intelligence 123 (2023): 106432.

[18] Ali, J., Roh, B. H., Lee, B., Oh, J., & Adil, M. (2020, October). A Machine Learning Framework for Prevention of Software-Defined Networking controller from DDoS Attacks and dimensionality reduction of big data. In 2020 International Conference on Information and Communication Technology Convergence (ICTC) (pp. 515-519). IEEE.

[19] S. Dong and M. Sarem, "DDoS Attack Detection Method Based on Improved KNN With the Degree of DDoS Attack in Software-Defined Networks, IEEE Access, vol. 8, pp. 50395048, 2020

[20] Aslam, Naziya, Shashank Srivastava, and M. M. Gore. "A comprehensive analysis of machine learning-and deep learning-based solutions for DDoS attack detection in SDN." Arabian Journal for Science and Engineering 49, no. 3 (2024): 3533-3573.

[21] Sahoo, K.S., Tripathy, B.K., Naik, K., Ramasubbareddy, S., Balusamy, B., Khari, M. and Burgos, D., 2020. An evolutionary SVM model for DDOS attack detection in software defined networks. IEEE access, 8, pp.132502-132513.

[22] Chauhan, Pinkey, and Mithilesh Atulkar. "An efficient centralized DDoS attack detection approach for Software Defined Internet of Things." The Journal of Supercomputing 79, no. 9 (2023): 10386-10422.

[23] MCUs-Sales-To-Reach-RecordHigh-Annual-Revenues-Through-2022. Available online: https://www.icinsights.com/news/bulletins/MCUs-Sales-To-Reach-RecordHigh-Annual-Revenues-Through-2022/ (accessed on 3 March 2024)

[24] Schizas, Nikolaos, Aristeidis Karras, Christos Karras, and Spyros Sioutas. "TinyML for ultra-low power AI and large scale IoT deployments: A systematic review." Future Internet 14, no. 12 (2022): 363.

[25] Hayajneh, A. M., Hafeez, M., Zaidi, S. A., & McLernon, D. (2024). TinyML Empowered Transfer Learning on the Edge. IEEE Open Journal of the Communications Society.

[26] Yang, L., El Rajab, M., Shami, A., & Muhaidat "Enabling AutoML for Zero-Touch Network Security: Use-Case Driven Analysis," IEEE Transactions on Network and Service Management, 2024.

[27] Capogrosso, L., Cunico, F., Cheng, D. S., Fummi, F., & Cristani, M "A Machine Learning-oriented Survey on Tiny Machine Learning," 204, IEEE Access.

[28] Dutta, Lachit, and Swapna Bharali. "Tinyml meets iot: A comprehensive survey." Internet of Things 16 (2021): 100461.

[29] Abadade, Youssef, Anas Temouden, Hatim Bamoumen, Nabil Benamar, Yousra Chtouki, and Abdelhakim Senhaji Hafid. "A comprehensive survey on tinyml." IEEE Access (2023).

[30] Zahid, Hafiz Muhammad, Yasir Saleem, Faisal Hayat, Farrukh Zeeshan Khan, Roobaea Alroobaea, Fahad Almansour, Muneer Ahmad, and Ihsan Ali. "A framework for identification and classification of iot devices for security analysis in heterogeneous network." Wireless Communications and Mobile Computing 2022, no. 1 (2022): 8806184.

[31] Saaty TL. Theory and applications of the analytic network process: decision making with benefits, opportunities, costs, and risks. RWS publications; 2005.

[32] Ryu controller: https://ryu-sdn.org/, Accessed: 20th January 2024

[33] Home. Available online: https://sites.google.com/view/iot-network-intrusion-dataset/home (accessed on 20 March 2024).

[34] K. Keerthi Vasan and B. Surendiran, "Dimensionality reduction using Principal Component Analysis for network intrusion detection," Perspectives in Science, vol. 8, pp. 510-512, 2016.

Jehad Ali Dr. Ali is working as an Assistant Professor in Ajou University South Korea. Mr. Ali did his PhD in Computer Engineering from Ajou University, South Korea. Mr. Ali has published his research works in IEEE IoT journal, IEEE Transactions on Industrial Informatics, IEEE Transactions on Intelligent Transportation Systems, Elsevier Journals, and other reputable publishers. Dr Ali research interests include computer networking, SDN, SD-IoT, and applications of artificial intelligence, as well as machine learning in computer networks.

Professor Houbing Song (Fellow, IEEE) received the M.S. degree in civil engineering from The University of Texas at El Paso, El Paso, TX, USA, in December 2006, and the Ph.D. degree in electrical engineering from the University of Virginia, Charlottesville, VA, USA, in August 2012. He has been serving as an Associate Technical Editor for IEEE Communications Magazine since 2017, an Associate Editor for IEEE Internet of Things Journal since 2020, IEEE Transactions on Intelligent Transportation Systems since 2021, and IEEE Journal on Miniaturization for Air and Space Systems since 2020, and the Guest Editor for IEEE Journal on Selected Areas in Communications.

Professor Dr. Vandana Sharma is a professor in CHRIST (Deemed to be University), Delhi NCR Campus, India. She has been awarded as high impact researcher in 2023 in the university by publishing a high number of Scopus index articles in the Delh NCR Campus, India.

Dr. Mahmoud AlKhasawneh is a Professor in Skyline University, United Arab Emirates. Mr Mahmoud di his Ph.D. (computer science) form Universiti Teknologi Malaysia (UTM), Malaysia. Moreover, in 2018, he did his Master (computer science) from Universiti Teknologi Malaysia (UTM), Malaysia, and Bachelor in computer science from Yarmouk University, Jordan in 2003. He has published his research works in reputed IEEE and Elsevier Journals.