ISSN: 2977-0041

Research Article

Journal of Material Sciences and Engineering Technology

Intrusion Detection System in Wireless IoT Network using feature selection and Bi-LSTM

Mustafa T. Mohammed Alhashimi

Communication Techniques Engineering Department, Al Najaf Engineering Technical College, Al-Furat Al-Awsat Technical University (ATU), Al Najaf, 54001, Iraq

***Corresponding author**
Mustafa T. Mohammed Alhashimi, Communication Techniques Engineering Department, Al Najaf Engineering Technical College, Al-Furat Al-awsat Technical University (ATU), Al Najaf, 54001, Iraq.

Received: November 07, 2024; **Accepted:** November 14, 2024; **Published:** November 20, 2024

ABSTRACT
Currently, the Internet of Things has grown widely and has provided new possibilities for applications in various fields. However, because of its dynamic architecture and constrained device resources, the Internet of Things presents numerous security challenges. Messages can be transported between wireless sensor network nodes at the network layer thanks to the Routing Protocol for Low Power Network (RPL). Various methods have been presented to identify RPL network assaults in recent years due to the sensitivity of the RPL routing protocol. In this research, an approach to detect IoT attacks based on the RPL protocol with the help of machine learning methods, principal component analysis (PCA) and neighborhood component analysis (NCA) for feature selection and classification based on Bi-LSTM neural network is presented. The results of the simulation and performance evaluation of the proposed method in the form of Matlab programming language show that the proposed approach has been able to provide a much better performance than the compared method.

Keywords: IoT Attacks, RPL Protocol, PCA, NCA, Bi-Directional LSTM Network (Bi-LSTM)

List of abbreviations

Routing Protocol for Low Power Network	RPL
Bi-directional LSTM network	Bi-LSTM
Principal Component Analysis	PCA
Neighborhood Component Analysis	NCA
Low-power and Lossy Networks	LLN
Intrusion Detection System	IDS

Introduction

The Internet of Things (IoT) has evolved rapidly over the past few decades and has found widespread applications in various sectors such as business, transportation, education, and entertainment. Sensor nodes, RFID tags, near field communication (NFC) devices, and other wired or wireless devices that exchange data with each other make up part of the Internet. Sensor nodes in the Internet of Things network generate data at high speed [1]. They transmit data to the base station for processing purposes and can also communicate with other sensor nodes to finally send their information to the base station [2]. As a result, the Internet of Things exchanges a lot of sensitive and private data, which, if its security is breached, may have serious negative effects on the economy and even endanger human lives [3].

As a result of network characteristics, attacks on IoT devices may be difficult to detect [4]. Thanks to the Routing Protocol for Low Power and Lossy Network (RPL), messages can be transferred between wireless sensor network nodes at the network layer. The routing strategy chosen for these networks should also consider both the dispersion of communication links and the energy requirements of sensor nodes, such as the RPL routing protocol, which is customized for LLNs used in the Internet of Things [5]. The directed rotation diagram (DODAG) that RPL creates enables point-to-point and multi-point communication. RPL outperforms other protocols when it comes to use in low-power and lossy networks (LLN). Recently, the implementation of the RPL protocol has increased in applications such as smart grids. Also, RPL is included in various operating systems to facilitate its further use [6].

Citation: Mustafa T Mohammed Alhashimi. Intrusion Detection System in Wireless IoT Network using feature selection and Bi-LSTM. J Mat Sci Eng Technol. 2024. 2(4): 1-8. DOI: doi.org/10.61440/JMSET.2024.v2.30

On the other hand, the growth of smart devices and applications connected to the Internet in the last ten years has led to a sharp increase in cyber attacks that target the Internet of Things. As a result, the RPL protocol is vulnerable to a number of attacks that negatively affect data transmission and severely degrade the topology by consuming resources.

Also, some attacks pose a serious threat to the reliability and security of RPL-based IoT networks. Several types of IoT attacks, including selective forwarding, Sybil attacks, and black hole attacks, have the ability to disrupt network integrity and interfere with node-to-node communications. The energy consumption of network nodes can be severely affected by the attack on the routing protocol for lossy low-power networks. Since the purpose of the RPL protocol is to establish effective communication in order to increase the battery life of these nodes with limited resources.

If the RPL routing protocol is compromised, nodes are forced to travel longer paths, which requires more data to be transmitted and received, resulting in more energy consumption. Nodes may also need to resend data packets, further reducing their battery life. In the end, this can lead to a shorter network lifetime. In addition, the involvement of a malicious node in multiple attacks, such as black hole and selective attacks, may lead to increased energy consumption throughout the network. This can lead to a decrease in the overall efficiency of the network and worsening of power consumption problems. As a result, it is very important to build strong defenses against routing attacks in IoT networks.

Therefore, the design of an intrusion detection system (IDS) plays an important role in the security of computer networks and protects user data and IoT devices from malicious activities. An efficient detection system is usually created through the use of data mining techniques due to the fact that they can detect intrusions exceptionally well. Considering that an intrusion detection system usually deals with a large amount of data, the feature selection step that exists in most of these systems is a challenging step because it considers only the selected features as a subset of the total features for classification which increases the classification speed. Therefore, one of the most important challenges in designing an intrusion detection system is to select and reduce the subset of suitable features [7].

In recent years, various techniques have been proposed to detect attacks in the Internet of Things, including anomaly detection. These techniques use feature extraction and selection methods based on artificial intelligence methods to identify and classify attacks [4].

This research focuses on attacks in Internet of Things that target the routing protocol for low-power and lossy networks (RPL), and a new method for feature selection coupled with machine learning classification is proposed. Machine learning (ML) techniques can be used to detect anomalies with a high positive rate, as they have proven to be effective in the field of cyber security and also provide these systems with acceptable flexibility and learning ability [8].

Therefore, in order to predict and stop attacks on RPL protocol in Internet of Things networks, a network model based on machine learning is presented in this work. In addition, machine learning and artificial intelligence methods can handle huge amounts of data, making it a potentially suitable method for the massive data provided by sensor networks [6,9]. Therefore, in order to implement an intrusion detection system and detect attacks in the RPL-based network using a hybrid methodology, this research proposes a machine learning model called ML-DT to detect attacks in an RPL-based Internet of Things network. The innovation of this research is to present a hybrid model that uses pre-processing and feature selection using two methods of principal component analysis (PCA) and neighborhood component analysis (NCA) along with decision tree classifier to detect RPL attacks.

Long-term short-term memory (LSTM) network is one of the types of deep recurrent neural networks and has achieved significant performance in classification. Bi-directional short-term memory (BiLSTM) is a further extension of LSTM, and BiLSTM combines a forward hidden layer and a backward hidden layer, which can access both previous and next contexts. Compared to BiLSTM, LSTM only exploits historical context. Hence, BiLSTM can solve the sequential modeling task better than LSTM. In particular, it was observed that BiLSTM models provide better predictions compared to ARIMA and LSTM models. It was also observed that the BiLSTM models reach equilibrium much slower than the LSTM-based models, so Bi-Directional LSTM (BiLSTM) is used in the future for prediction and classification. Unlike the standard LSTM, the input flows in both directions and can use information from both sides.

The method for identify RPL routing attacks will include four basic steps. In the first step, the RPL-Attack dataset is received, which is a dedicated dataset for RPL protocols, then these data are preprocessed in the second step, and in the third step, features are selected using PCA and NCA method and its extra features are removed [9]. In the last step, the optimized datasets are classified using Bi-LSTM machine learning methods.

The rest of the paper is organized as follows: The second section presents the background of the RPL protocol and some recent research work related to IDS for the Internet of Things. In the third part, an overview of the proposed approach based on machine learning algorithm is provided. The experimental results obtained from the proposed approach are shown in the fourth section. Finally, the general result is presented in the fifth section.

Background and related works

RPL is a unique routing solution for low-power, lossy networks and, in fact, was created specifically to meet the needs of constrained systems. This protocol is designed for networks where nodes send point-to-point communication from the sink node to devices within the LLN as well as measurements that are regularly collected to a single point. This section provides a background of RPL: its architecture, routing policies, and IDS-related tasks.

Architecture

RPL generates a destination-directed acyclic graph (DODAG), as illustrated in Figure 1. The graph begins with a root node that is linked to a few low-power devices as its offspring and the global Internet as its parent. 6BR is the name of this root node.

An IPv6 address known as the node ID, one or more parents, a list of neighbors, and a rank that establishes the node's location in relation to the root and other nodes are all contained in each DODAG node.

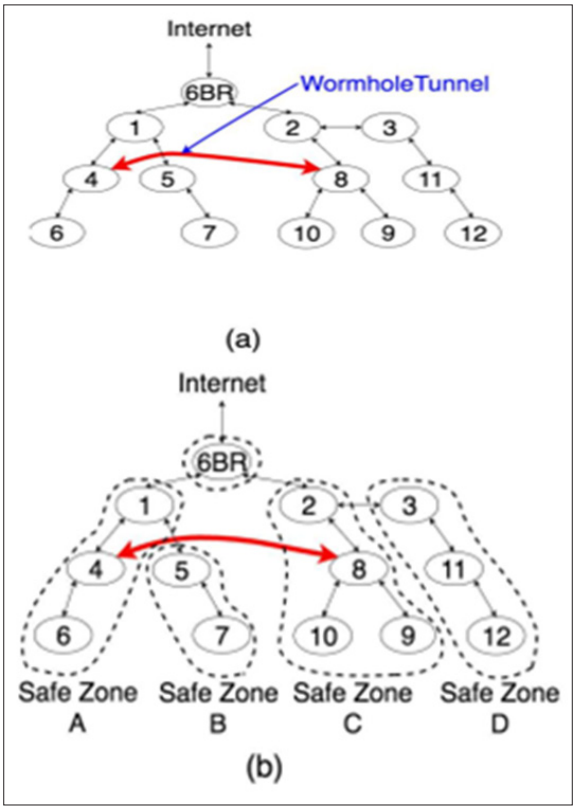

Figure 1: RPL network: (a) wormhole, (b) safe zones.

Routing policies

Source routing and table-based routing are the two types of routing policies that are typically supported by the RPL protocol. The packet contains the whole route to the destination(s) in source routing. In contrast, all nodes in a table-based routing maintain a routing table in order to determine the destination of incoming and outgoing packets. When nodes find new, shorter paths, table-based routing continuously adds new neighbors to the connection tables so that the destination is reached more quickly.

Related works

From the past decades to today, many researches and developments in the field of intrusion detection and anomaly detection in various networks continue due to its transformation. Many researches have been presented using classical ML algorithms, such as Decision Tree (DT), Naïve Bayesian (NB), Logistic Regression (LR), Expectation Maximum Clustering (EM), ANN, FL, GA, and Support Vector that in the following, some recent studies in this field, have discussed.

Mane et al. presented a novel assault known as the Flooding attack and a counter detection technique based on the RPL protocol in their research effort [10]. A flooding attack overloads both the motes and the network by having hostile motes directly bombard them with huge packets. Distributed denial of service, or DDoS, is another term for this kind of attack. The authors' research proposes an intrusion detection system prevention technique that counts the amount of packets delivered and received from each neighbor in order to identify and counteract flooding attacks. Based on a performance assessment carried out in the Cooja simulator in various real-time scenarios. With a high degree of accuracy and true positive rate, the IDS algorithm can identify flooding attacks.

Mozamel et al. addresses Rank and Blackhole threats in RPL while taking into account both stationary and mobile Internet of Things nodes [11]. The authors' suggested Security, Mobility, and Trust (SMTrust) model is predicated on a thorough selection of measurements and trust elements, including mobility-based indicators. Simulation experiments evaluating the proposed model demonstrate that SMTrust outperforms current trust-based approaches in protecting RPL.

A novel intrusion detection system for Internet of Things infrastructure was presented by Elazab et al. [12]. It makes use of a number of features that are especially made to identify various kinds of routing assaults. The suggested solution is built on a hybrid methodology that enhances detection accuracy by combining two detection processes with the cumulative set method. The experimental findings demonstrate that, in terms of accuracy and false alarm rate.

Othman et al. describe a learning-based intrusion detection system (ELG-IDS) that employs extreme parameter optimization and clustering to identify three internal RPL attacks in response to the shortcomings of anomaly-based intrusion detection systems [13]. Version number, upgrade, and deluge of assaults. Genetic algorithm-based feature selection and sophisticated feature extraction are used in their suggested ELG-IDS system. ELG-IDS has demonstrated outstanding accuracy in experimental results on a dedicated dataset. It has a relative improvement for version number, rank attack, DIS flood, and accuracy in multi-classification mode.

An unsupervised group-based anomaly detection (OEAD) method was presented by Budania and Shenoy for anomaly identification. With the most recent and representative traffic that reflects the state of the network, this live model can be modified and retrained [14]. In the OEAD design, a drift detector unit is utilized to identify notable changes in network traffic, which can be used to update the machine learning model for network drift detection. A publicly accessible RADAR dataset is used to test the suggested OEAD method, and the findings indicate that the technology holds promise for anomaly identification in real-time applications.

AMI attack-aware intelligent machine learning identities is a novel intrusion detection method that Savitha et al. proposes to provide secure and dependable AMI-RPL [15]. Predicting, detecting, and mitigating different kinds of RPL security attacks in an AMI system are the main objectives of AIMS. A new AMI-RPL attack dataset with suitable preprocessing and feature selection based on Spider Monkey Optimization (SMO) is developed by the Cooja simulator in order to forecast RPL attacks using the Ensemble Machine Learning (SE) model.

The basic article which is presented in this research to compare and evaluate the performance of the results of the proposed method, which uses a machine learning-based Secure RPL Routing Protocol (MLRP) to detect attacks before they have

a large impact on the Internet of Things [16]. First, the MLRP protocol generates a complex dataset including normal and attack behavior using the Cooja simulator. Second, the dataset is machine-learned to effectively identify attack behaviors that are version, rank, and denial of service (DoS). In addition, MRLP classifies the types of attacks using Support Vector Machine (SVM) classifier. To improve the performance of SVM, an improved principal component analysis (PCA) is introduced to effectively reduce the dimensionality of the dataset. Finally, the simulation results show that the proposed MLRP protocol accurately increases the attack detection accuracy and is suitable for the IoT environment. Studies and reviews have shown that IoT wireless sensor networks are networks with special and unique characteristics that distinguish these networks from other wireless networks. One of the most important distinctions is related to routing and data transmission. With the investigations carried out in connection with the various applications of wireless sensor networks, it was found that in all these fields (due to the very high importance of the sent information), the category of maintaining the quality of service in routing and data transmission along with reducing energy consumption is considered the most important issue and improving It plays a significant role in improving network efficiency. For this purpose, combined machine learning algorithms such as Bi-LSTM neural network have been used in feature classification. In this study, the ideal range value for detecting assaults in an RPL-based IoT network is chosen by utilizing a machine learning model known as ML-BiLSTM. Then calculate the detection rate based on this range. For the first time, IDS is using machine learning to identify assaults in IoT networks.

The proposed method

In this research, an approach based on feature selection and machine learning to detect attacks has been developed. In the first step, the proposed approach receives an RPL-Attack dataset and divides it into training and testing datasets [9]. In the second step, the proposed approach selects an appropriate subset of features using two algorithms, principal component analysis (PCA) and neighborhood component analysis (NCA). The purpose of this combination is to free the data set from as much complexity as possible and simplify the work for the classification algorithms. The existence of machine learning algorithms help a lot in classification. These smart algorithms can learn patterns in the past and predict them in the future. Therefore, in the next step, the proposed method uses the Bi-LSTM neural network algorithm to classify the data and detect RPL attacks based on the patterns that the neural network extracts from past data. Next, each of these steps is explained in Figure 2.

Step 1: Data entry

In the first step, the data is first entered into the system to implement the proposed method. In this research, the RPL-attack dataset is used, which is a dedicated dataset for RPL protocols. A specialized dataset developed to investigate routing RPL protocol, with the primary goal of developing a broad and diverse dataset to facilitate in-depth analysis and complement existing resources.

Step 2: Data preprocessing

Data pre-processing includes steps of numbering, cleaning and normalization, and Figure 3 shows the pseudo-code of the data pre-processing process. Data cleaning is used to remove and fix corrupt, duplicate and incorrect data in a dataset. The main goal is to clean the data set, which standardizes the data analysis and is easily accessible to find the correct data for the query.

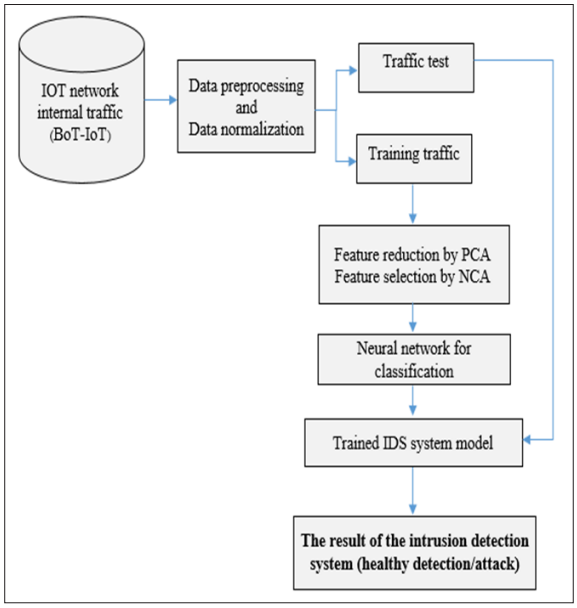

Figure 2: A schema of ML-BiLSTM method

```
Function: ReadDataAndPreproccess (a)
Output: data
        all_data ← read data from dataset
        Data ← select a precent of all_data randomly
        Data ← transform non numeric columns to numeric
        Data ← remove NAN values of data
        Data ← remove Miss Values
        Data ← Normalize data
End Function
```

Figure 3: Preprocessing phase pseudocode

Data are normalized upon entry. Data normalization is a method used in data mining to convert the values of a data set to a standard scale. Normalization is an important process because many machine learning algorithms are sensitive to the magnitude of input features and benefit from data normalization by producing more accurate results.

Normalization should only be applied to the input features and not the target feature. In this research, the feature vector is performed using min-max normalization. Normalization is calculated using equation 1.

$$X(:,i)^N = \frac{X(:,i) - \min(X(:,i))}{\max(X(:,i)) - \min(X(:,i))} = \{1, 2 \ldots N\} \quad (1)$$

Where X is the final feature vector of size N.

Step 3: PAC and NCA feature selection

Basically, the main goal of feature selection approaches is to select an approximately optimal feature subset in order to maximize the classification ability of a learning model. Therefore, to select an efficient subset of features, a reliable algorithm that accurately

determines the subset of features needed to classify the studied patterns is needed.

At this stage, by removing additional features in the data set using NCA and PCA approaches, data refinement and improvement are done. This approach makes the data refined with higher quality as much as possible and redundant features are removed. Removing extra features reduces the complexity of the data. The PCA method works well with the dimensionality of the space, does not reduce the number of primary features, and requires all the primary features to create new hidden features (principal components). Reducing the number of features is also very useful for potential evaluation or testing purposes. Selecting a function (feature) is possible by selecting knowledge features or removing additional features.

PCA feature selection

The algorithm (PCA) reduces the features of the data set from "n" to "p" where p < n. In order to select the feature, the following steps are implemented in this research:

In the first stage of standardization, the range of features in this process is calculated and standardized so that the contribution of each feature is equally analyzed. Calculation of primary features helps to categorize features that dominate other small range features. To convert the characteristics of a standard, can use the following formulas, where X = value in a data set and n = number of values in the data set.

$$z = \frac{\text{Value-Mean}}{\text{Standard Deviation}} \tag{2}$$

$$\text{MEAN} = \frac{\text{sum of terms}}{\text{Total number of terms}} \tag{3}$$

$$\text{Standard Deviation} = \sqrt{\frac{(\text{x-mean})^2}{\text{n}}} \tag{4}$$

In the second step of calculating the covariance matrix, the characteristics of the given data are changed by the calculated average value. Any related features can be sorted at the end of this step. In continue, use the provided formula to build the covariance matrix, which helps isolate features that are closely related to one another. An $N \times N$ symmetric matrix with the covariance of every conceivable data collection is called a covariance matrix.

$$\text{Covariance Matrix} = \begin{bmatrix} \text{COV(X,X)} & \text{COV(X,Y)} \\ \text{COV(Y,X)} & \text{COV(Y,Y)} \end{bmatrix} \tag{5}$$

$$\text{Covariance} = \frac{\text{Sum (X-(Mean of X)(Y-(Mean of Y)))}}{\text{number of data notes}} \tag{6}$$

The covariance matrix's positive value denotes a positive correlation between the features. (Y also rises when X does, and vice versa). The characteristics are inversely correlated if the covariance matrix has a negative value. (Y likewise reduces if X grows, and vice versa). As a result, eventually pairs that are related to each other will be identified and the feature classification process will be much simpler.

In the third stage of feature vector formation, to determine the main components of the features, eigenvalues and eigenvectors must be defined for them, and using the data required in training, they are placed in an array. The vectors uniformly choose their highest values and denote these vectors by *V1,.....Vm.* To put it another way, after determining the eigenvector components, list the primary components by defining the eigenvalues for each feature in descending order. As a result, the principal components which indicate the direction of the data are represented by the eigenvalues.

In the last step of feature selection, feature component j is calculated as follows:

$$c_j = \sum_{p=1}^{m} |V_{pj}| \tag{7}$$

j: V_{pj} *input,* V_{pj} *=1.2.3...N*

Finally, our large dataset is compressed into a small dataset without data loss. Next, the NCA algorithm is executed.

NCA feature selection

Next, after feature selection using PCA, the non-parametric feature selection technique, neighborhood component analysis (NCA) is used with the aim of increasing the power of the proposed approach and classification. The fscnca and fsrnca functions are used in this research to perform NCA feature selection with regularization in order to learn the weight of the features in order to minimize an objective function that improves the regression accuracy on the training data. By Using NCA, perform feature selection again on the features that were normalized and selected. The steps of the NCA algorithm in this study are as follows:

Let $T=\{(x_1,y_1),\ldots,(x_i,y_i),\ldots,(x_N,y_N)\}$ be a set of training data, with x_i being a d-dimensional feature vector, being a single class label, and N is the total number of samples, the goal is to find the weight vector w that supports the selection of the nearest subset of the neighbor classification optimization feature. The weighted distance between two samples x_i and x_j which is expressed in terms of the weight vector w, is obtained as follows:

$$D_w(x_i,y_i) = \sum_{l=1}^{d} w_l^2 |x_{il} - x_{il}| \tag{8}$$

Where the weight w_l is connected to the feature *l* Maximizing its individual classification accuracy in the training set T is a logical and efficient method for the success of nearest neighbor classification. However, since the genuine accuracy that chooses the nearest neighbor as the classification reference point is a non-separable function, an appropriate approximation is that the reference point is decided by a probability distribution. Here's how to calculate the probability that x_i chooses x_j as its reference point:

$$p_{ij} = \begin{cases} \dfrac{k(D_w(x_i,x_j))}{\sum_{k \neq i} k(D_w(x_i,x_k))}, & \text{if } i \neq j \\ 0, & \text{if } i=j \end{cases} \tag{9}$$

Based on the above definition, the probability of correct classification of the query point x_i is given as follows:

$$p_i=\sum_j y_{ij}\, p_{ij} \quad (10)$$

Where $y_{ij}=1$ if and only if $y_i=y_j$ and $y_{ij}=0$.

Step 4: classification with Bi-LSTM

A unique kind of RNN network called short-term memory, or LSTM network, addresses the long-term memory issue with RNN networks. Gates are internal mechanisms in LSTM networks. These entry points regulate the information flow. Additionally, they indicate which sequence data should be discarded and which should be maintained since they are valuable. The network uses this method to transfer crucial data along the chain of events in order to produce the intended result. Figure 4 depicts its basic architecture, which consists of input, output, and forget gates.

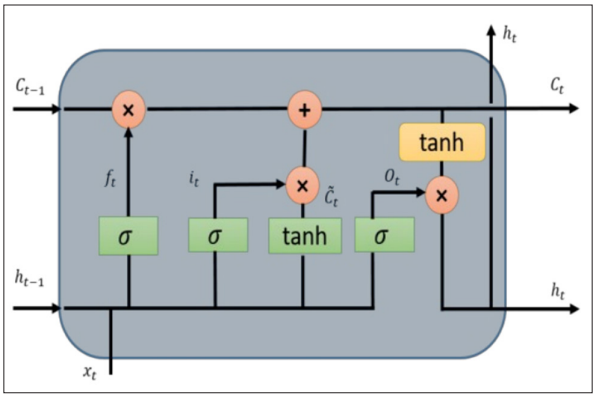

Figure 4: Structure of an LSTM network

As shown in Figure 4, the LSTM network has several different operations in its internal structure.

The network's input at time t is represented by x_t, its output at time t-1 by h_{t-1} and C_{t-1} correspondingly, and the logistic sigmoid function by σ. The input, forget, output, and cell state are represented by i, f, o, and c, respectively. Similar to a typical RNN, the LSTM architecture has a variety of repeating modules for every time step. A set of gates in Rd the forget gate f_t, the input gate I_t, and the output gate o_t. Control the module's output at each time step as a function of the previous hidden state, h_{t-1} and the input at the current time step X_t.

In this design, all vectors share a single dimension, which is represented by the symbol d for the memory dimension in an LSTM. The definition of LSTM transfer functions is as follows:

$$i_t=\sigma(W_{xi} X_t+ W_{hi} h_{t-1}+W_{ci} c_{t-1}+ b_i) \quad (11)$$

$$f_t=\sigma(W_{xf} X_t+ W_{hf} h_{t-1}+W_{cf} c_{t-1}+ b_f) \quad (12)$$

$$c_t=f_t c_t\text{-}1+i_c \tanh(W_{xc} X_t+ W_{hc} h_{t-1}+ b_f) \quad (13)$$

$$o_t=\sigma(W_{xo} X_t+ W_{ho} h_{t-1}+W_{co} c_t+ b_o) \quad (14)$$

$$h_t=O_t \tanh (c_t) \quad (15)$$

In the design of neural networks, after determining the type of network and the training method, the number of input nodes, the number of hidden (intermediate) layers and hidden nodes, and the number of output nodes should be determined. Choosing the number of inputs is also of particular importance, because each input pattern includes important information about the self-correlated and complex structure of the data.

BiLSTM, also known as bidirectional LSTM, is a kind of LSTM network that enables improved learning at each stage of the data by feeding the information from the end to the beginning and the learning data from the beginning to the end of the model. Two LSTM networks make up BiLSTM: a forward LSTM network and a backward LSTM network [17]. Extracting features from the forward direction is done by the forward LSTM hidden layer, and extracting features from the backward direction is done by the backward layer. In other words, information can be used from both sides and input can flow both ways. Additionally, it is an effective tool for simulating word and phrase sequential dependencies in both directions of the sequence, forget gate and output gate.

In short, the two unidirectional LSTMs that make up the Bi-LSTM architecture process the sequence both forward and backward. Two distinct LSTM networks, one receiving the token sequence and the other reversing the order, can be understood as part of this architecture. The output of each of these LSTM networks is a probability vector, and the combined result of these probabilities is the final output. Put simply, it indicates that in the extra layer of the LSTM, the input sequence runs backwards. The outputs of the two LSTM layers are then combined in a variety of ways, including concatenation, addition, multiplication, and average. The following are the forward and backward outputs included in the Bi-LSTM equation at time t:

$$h_t^{forward} = LSTM^{forward} (h_{t-1}, x_t, C_{t-1}) \quad (16)$$

$$h_t^{Backward} = LSTM^{forward} (h_{t-1}, x_t, C_{t-1}) \quad (17)$$

$$H_t=[h_t^{forward}, h_t^{backward}] \quad (18)$$

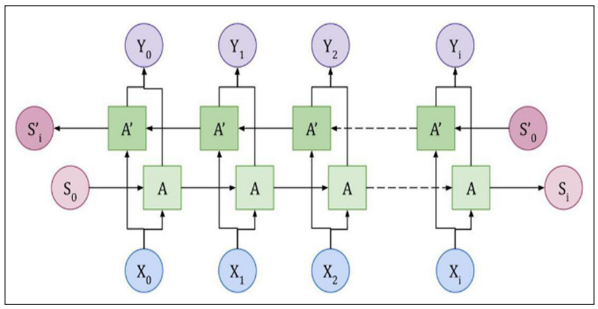

Figure 5: Bidirectional LSTM layer architecture [17]

Figure 5 describes the Bi-LSTM layer architecture, where is the input signal, is the output signal, and A and A' are LSTM nodes. The final output is a combination of LSTM nodes, A and A'.

Evaluation results

An RPL attack detection algorithm should be carefully evaluated to measure its performance in predicting RPL attacks. For this purpose, the proposed method, which is a combination of PCA and NCA feature selection algorithms for feature selection and decision tree for classification, is called PCA-NC-DC and the compared method of the basic article which supports PCA feature selection and vector machine [16]. For classification, PCA-SVM is used, implemented with MATLAB 2022a software. Since the operation performed by the systems is considered a type of

classification, a common classification criteria, i.e., Precision, Recall, Accuracy, and F-measure have used.

$$\text{Precision} = \frac{TP}{TP+FP} \quad (19)$$

$$\text{Recall} = \frac{TP}{TP+FP} \quad (20)$$

$$\text{Accuracy} = \frac{TP+TN}{TP+TN+FP+FN} \quad (21)$$

$$\text{F-Measure} = \frac{2*\text{Precision}*\text{Recall}}{\text{Precision}+\text{Recall}} \quad (22)$$

The RPL-Attack dataset is a specialized dataset created using the Contiki operating system and Cooja simulation to investigate attacks against the RPL routing protocol [18]. In this dataset, a deliberate attempt has been made to encompass various network scenarios. It involved the integration of healthy and malicious nodes, each with a specific role. Three specific types of attacks are simulated in this dataset:

1. **Flood attack:** This scenario involves deliberately generating excessive network traffic, mimicking a flood attack. The goal was to determine how RPL and network performance respond in such situations.
2. **Degraded rank attack:** In simulated attack, certain nodes were manipulated to lower their rank in the network. This manipulation was done to disrupt routing decisions and evaluate the strength of the network.
3. **Version Number Increment Attack:** Another aspect of the dataset focuses on increasing the version number of network nodes. This action measures the network's flexibility against manipulation of the version number.

Considering that the proposed approach is based on machine learning, as a result, the size of the training dataset can have a great impact on its performance. Therefore, experiments have been performed for different sizes of the training dataset. The results of the evaluations can be seen in the diagrams of Figures 6 to 9.

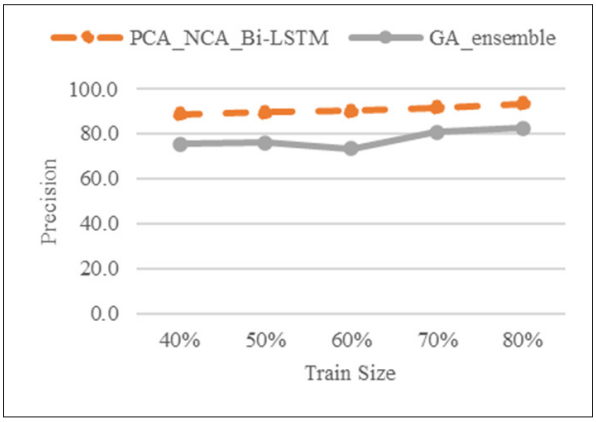

Figure 6: Precision vs. train size.

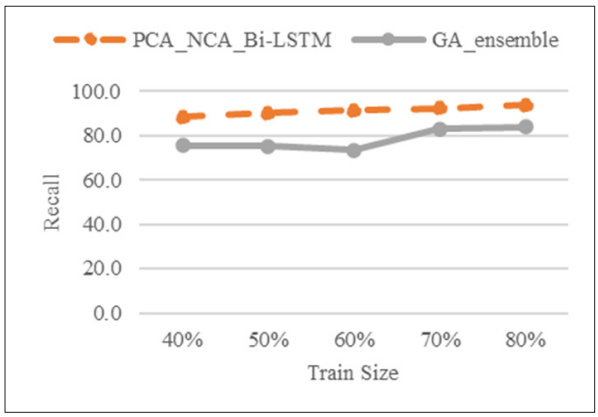

Figure 7: Recall vs. train size.

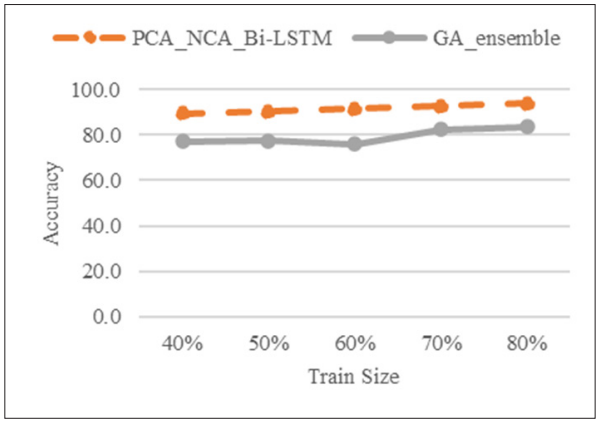

Figure 8: Accuracy vs. train size.

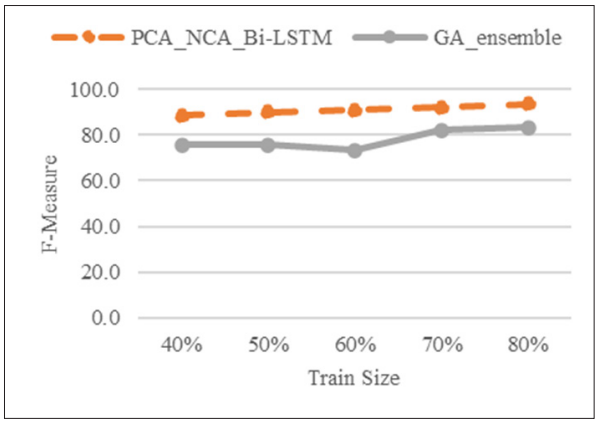

Figure 9: F-Measure vs. train size.

The results show that the proposed method provides better results in four of the presented criteria. The proposed method's superiority to the basic method applies to all sizes of the training data set. Table (1) contains the proposed method improvement compared to the GA_ensemble method.

Table 2: Proposed method improvement.

Improvement (%)	Criterion
13 to 23	Precision
11 to 25	Recall
12 to 21	F-Score
12 to 24	Accuracy

The most important reason for the superiority of the proposed method over the compared method is that the feature selection

of the proposed method is a combination of PCA and NCA algorithms. Using two strong algorithms for feature selection means better refinement of the data set and, as a result, better quality feature selection. Better data refinement has helped the neural network deal with fewer features as a classifier. As a result, it can form a network of fewer neurons and have higher accuracy. Finally, it can better classify the data set [19,20].

Conclusion

Detecting RPL attacks is one of the hot research topics in sensor networks and IoT. If RPL attacks can be recognized well, effective parameters can also be obtained. Therefore, detecting RPL attacks has recently gained many fans among research institutes and universities.

Proposed methods to detect RPL attacks often focus on using machine learning algorithms. These methods are trying to detect RPL attacks using machine learning capabilities. However, a critical aspect of RPL attack detection, feature processing, must be addressed. In this research, an approach for predicting RPL attacks using intelligent algorithms is presented. The proposed approach selects an appropriate subset of features by using two algorithms, PCA and NCA. The purpose of this combination is to free the data set from the complexity as much as possible and simplify the work for the classification algorithms. The existence of machine learning algorithms helps classification algorithms a lot. These intelligent algorithms can learn past patterns and predict the future. Therefore, in the continuation of the proposed method, the data is classified using the Bi-LSTM algorithm and RPL attacks are identified. With the help of MATLAB 2022a, the proposed method was evaluated and the evaluation results based on the criteria of accuracy, recall, precision and F measure show that the proposed method is improved by 23, 25, 21 and 24% respectively.

References

1. Evans D. The internet of things: How the next evolution of the internet is changing everything. CISCO white paper. 2011. 1: 1-11.
2. Umamaheswari S, Negi A. Internet of Things and RPL routing protocol: A study and evaluation. In Proc. 2017 International Conference on Computer Communication and Informatics (ICCCI). 2017. 1-7.
3. Cam-Winget N, Sadeghi A-R, Jin Y. Can iot be secured: Emerging challenges in connecting the unconnected. in DAC. 2016. 3: 1-6
4. Pongle P, Chavan G. A survey: Attacks on RPL and 6LoWPAN in IoT. In 2015 International conference on pervasive computing (ICPC). 2015. 1-6.
5. Kharrufa H, Al-Kashoash HA, Kemp AH. RPL-based routing protocols in IoT applications: A Review. IEEE Sensors Journal. 2019. 19: 5952-5967.
6. Haque KF, Abdelgawad A, Yanambaka VP, Yelamarthi K. An Energy-Efficient and Reliable RPL for IoT. In 2020 IEEE 6th World Forum on Internet of Things (WF-IoT). 2020. 1-2.
7. Venkatesan S. Design an intrusion detection system based on feature selection using ML algorithms. Mathematical Statistician and Engineering Applications. 2023. 72: 702-710.
8. Krishna EP, Thangavelu A. Attack detection in IoT devices using hybrid metaheuristic lion optimization algorithm and firefly optimization algorithm. International Journal of System Assurance Engineering and Management. 2021. 1-14.
9. Osman M, He J, Mokbal FMM, Zhu N, Qureshi S. Ml-lgbm: A machine learning model based on light gradient boosting machine for the detection of version number attacks in rpl-based networks. IEEE Access. 2021. 9: 83654-83665
10. Manne VRJ, Sreekanth S. Detection and mitigation of RPL routing attacks in Internet of Things. In 2022 9th International Conference on Computing for Sustainable Global Development (INDIACom). 2022. 481-485.
11. Muzammal SM, Murugesan RK, Jhanjhi NZ, Humayun M, Ibrahim AO, et al. A trust-based model for secure routing against RPL attacks in internet of things. Sensors. 2022. 22: 7052.
12. Alazab A, Khraisat A, Singh S, Bevinakoppa S, Mahdi OA. Routing Attacks Detection in 6LoWPAN-Based Internet of Things. Electronics. 2023. 12: 1320.
13. Osman M, He J, Zhu N, Mokbal FMM. An ensemble learning framework for the detection of RPL attacks in IoT networks based on the genetic feature selection approach. Ad Hoc Networks. 2024. 152: 103331.
14. Budania S, Shenoy MV.OEAD: An Online Ensemble-based Anomaly Detection technique for RPL network. In Proceedings of the 25th International Conference on Distributed Computing and Networking.2024. 316-321.
15. Savitha MM S, PI B. Securing AMI-IoT networks against multiple RPL attacks using ensemble learning IDS and light-chain based prediction detection and mitigation mechanisms. Information Security Journal: A Global Perspective. 2024. 73-95.
16. Osman Musa, He J, Zhu N, Mokbal FM. An ensemble learning framework for the detection of RPL attacks in IoT networks based on the genetic feature selection approach. Ad Hoc Networks. 2024. 152: 103331.
17. Jiang K, Wang W, Wang A, Wu H. Network Intrusion Detection Combined Hybrid Sampling With Deep Hierarchical Network. in IEEE Access. 2020. 32464-32476.
18. https://www.kaggle.com/datasets/azizkarimy/rpl-attacks.
19. https://www.geeksforgeeks.org/bidirectional-lstm-in-nlp/
20. Momand, Mohammad Dawood, and Mohabbat Khan Mohsin. Machine learning-based multiple attack detection in RPL over IoT. 2021 International Conference on Computer Communication and Informatics (ICCCI). IEEE. 2021.