International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 7
DOI: https://doi.org/10.17762/ijritcc.v11i7.7958
***Article Received**: 05 May 2023 **Revised**: 20 June 2023 **Accepted**: 18 July 2023*

Robust Deep Learning Based Framework for Detecting Cyber Attacks from Abnormal Network Traffic

K. Swathi[1], G. Narsimha[2]
[1]Department of Computer Science and Engineering, CVR College of Engineering
Hyderabad, India
swathireddykathi@gmail.com
[2]Department of Computer Science and Engineering, Jawaharlal Nehru Technological University Hyderabad,
Hyderabad, India
narasimha06@gmail.com

Abstract— The internet's recent rapid growth and expansion have raised concerns about cyberattacks, which are constantly evolving and changing. As a result, a robust intrusion detection system was needed to safeguard data. One of the most effective ways to meet this problem was by creating the artificial intelligence subfields of machine learning and deep learning models. Network integration is frequently used to enable remote management, monitoring, and reporting for cyber-physical systems (CPS). This work addresses the primary assault categories such as Denial of Services(DoS), Probe, User to Root(U2R) and Root to Local(R2L) attacks. As a result, we provide a novel Recurrent Neural Networks (RNN) cyberattack detection framework that combines AI and ML techniques. To evaluate the developed system, we employed the Network Security Laboratory-Knowledge Discovery Databases (NSL-KDD), which covered all critical threats. We used normalisation to eliminate mistakes and duplicated data before pre-processing the data. Linear Discriminant Analysis(LDA) is used to extract the characteristics. The fundamental rationale for choosing RNN-LDA for this study is that it is particularly efficient at tackling sequence issues, time series prediction, text generation, machine translation, picture descriptions, handwriting recognition, and other tasks. The proposed model RNN-LDA is used to learn time-ordered sequences of network flow traffic and assess its performance in detecting abnormal behaviour. According to the results of the experiments, the framework is more effective than traditional tactics at ensuring high levels of privacy. Additionally, the framework beats current detection techniques in terms of detection rate, false positive rate, and processing time.

Keywords-Deep Learning, Cyber Attacks, Cyber-physical systems (CPS), Recurrent Neural Networks (RNN), Linear Discriminant Analysis (LDA), NSL-KDD.

I. INTRODUCTION

Criminal activity and penetration attempt increasingly threaten local and satellite networks' security. The Strategies and methods for intrusion detection have been deemed essential to protect online resources. Because more devices are connecting to the internet, cyber security is becoming more critical. Security norms like integrity and confidentiality are among the things that are violated during intrusions. The adversary employs sophisticated programming tools to attack a network and look for weaknesses. As a result, the intrusion detection approach is crucial for keeping track and averting intrusions in a computing network atmosphere.

According to the Chinese national Data Security Vulnerability Sharing Platform, security-related vulnerabilities are growing by 1% annually. High-risk defects make up 34.5% of the 14,201 fundamental security flaws. There are often more than 4,000 dispersed denial of the facility (DDoS) attacks every year, according to the Chinese Internet emergency center. An attack that causes a denial of service (DoS) overloads the organization. Aggressors take advantage of well-known and popular servers, including those used by banks, to bring the system down and inflict significant financial damage. The logs also reveal that 1 million broilers, 90 000 IP addresses, and more than 2000 resources are utilized to launch DDoS attacks. This could mean that network attackers have illegally seized control of close to a million computers or mobile devices.

The advent of the internet, numerous improvements and technological advancements have significantly altered human existence, interpersonal relationships, and the environment. Interaction, collaboration, and data access were informal by the volume to attach computers anywhere in the wireless technology[1]. Complex, clever, intelligent, and self-aware CPSs have appeared recently. These contain robotics, transportation systems, hospital and medical areas, smart networks in electrical manufacturing, and manufacturing 4.0 in the industrial subdivision. [2] Due to the intricate interplay of numerous cyber and physical components and the fact that CPS activity is susceptible to significant disruptions brought on by unintentional events. Also, it is challenging to predict CPS activity. Meanwhile, researchers in business and academia are focused on cyber security for CPS because of the rise in the frequency and sophistication of cyberattacks, also known as zero-day vulnerabilities[3]. Figure.1 shows the basic CPS structure.

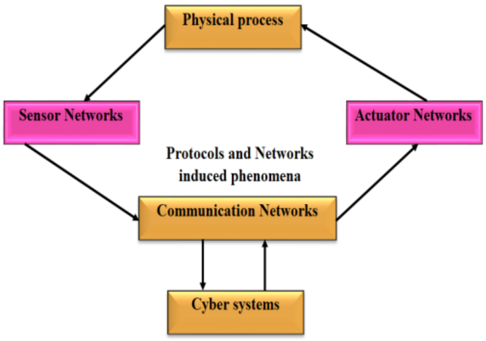

Figure 1: Basic CPS structure

The CPS can accomplish numerous psychological objectives like Grids, transportation systems, healthcare, and water or gas distributors are just a few applications. CPS also incorporates wireless industrialized sensor systems, wireless sensing networks, and networked control mechanisms. CPS can now manage tasks that formerly required much human effort via the Internet as expert machines. Numerous writers say these are physical and engineering systems where all processes are coordinated, controlled, and integrated by a central computer or communication core. The features of zero-day attacks are not stored in the security systems' databases. Therefore, access control and intrusion detection and anticipation schemes (IDS/IPS) cannot detect, prevent, or block them. CPSs are shielded against zero-day threats by artificial intelligence-based cyber security solutions [4]. Cybersecurity leverages ML technology to accomplish a considerable volume of varied information from multiple foundations to swiftly build distinct occurrence and accurately estimate upcoming hacker behavior. This enables a more precise forecast of the hackers' possible forthcoming behavior.

Collaboration across various AI and ML technologies is required to prevent zero-day vulnerability attacks in addition to the help of security experts [6]. Human decision-making enhances detection methods because human-machine collaboration aims to decrease the frequency of false convictions. Cyberattacks can disrupt the regular running of corporal procedures on CPS, which is why classical machine-learning algorithms are used to detect them. Definitive assessment of cyber-attacks against CPS gets challenging when networks are complex, and there needs to be more knowledgeable about the thing being studied. Artificial intelligence concepts and techniques can significantly enhance the capabilities and effectiveness of neural networks. The neural network may surprise with unassuming qualities and advance to more complicated ones because it has many layers.

The fact that so many people utilize the Internet makes it crucial to find a solution to the possible risks posed by network attacks as soon as possible. Researchers used various anomaly detection techniques to find attacks hidden in dense network data [7]. Based on how irregularities are identified, traditional methodologies for detecting malicious traffic can be allocated into three groups like statistical analysis approaches, machine learning methods, and signal processing approaches.

According to the results of their practical use, these techniques have partially solved security issues. However, compared to the typical network environment, the regulated network domain does not respond to these basic approaches to anomalous traffic detection. The Internet of Things (IoT) is a vast network of linked intelligent devices that provide online services to customers and companies [8–9]. The Internet of Things (IoT), which consents to real-time information collected from various sensors and actuators is significant in modern industry. Internet technology adoption is changing current industry trends regarding obtaining and analysing effectively to monitoring data about industrial processes. The internet technology may increase the efficiency and competence of the innovative subdivision through informed decision-making and remote management.

The information security is still a problem, even though these technologies improve our quality of life. Information security breaches caused by cyberattacks can happen for several reasons. DoS and dispersed denial of examination are the most significant corporate cyberattacks (DDOS). Attack and incursion are the dual stages of DDOS attacks. DDOS attack tools are installed on several network hosts during the intrusion phase. The attack phase involves an assault on the target network [10]. Attackers utilize these hosts to generate traffic to force the target routers. This synthetic traffic consumes significant bandwidth and resources on the target PC. A legal denial of service results from the objective system's are inability to offer good services to its users [11]. The main contribution of this research are

- The main attack categories are covered in this study (DDoS, Probe, U2R, R2L). As a result, we combine AI and ML (ML) approaches in this research to provide a novel Recurrent Neural Networks (RNN) cyberattack detection scheme.

- The CPS design comprises three layers: the physical layer, the network layer, and the application layer. To increase the clarity of functionality, layered structures are used. Then, with a focus on the physical system, CPS attacks on each layer are addressed.
- We assessed the current system using information from the Network Security Laboratory-Knowledge Discovery Database (NSL-KDD), including all essential issues.
- A time-ordered sequence of network flow traffic is learned using the suggested RNN-LDA model, and its performance in identifying anomalous behavior is evaluated.

The essay is organized as follows for the remaining portions. Section.2 of the paper discusses the significant works relevant to the current topic. Section.3 deliberates the datasets, suggested frameworks, and models for machine learning. Section.4 covers the experimental results, and finally the outcomes are discussed in the conclusion section.

II. LITERATURE SURVEY

A deep learning-based network intrusion detection technique is provided by Peng et al[12]. The Back propagation Neural Networks to classify the types of incursion and deep neural networks to extract topographies from network monitoring information. The approach is examined using the KDDCup99 dataset. The results demonstrate that the methodology outperforms the conventional machine learning method by 95.45%, a significant improvement.

Ludwig et al.[13] utilizes an ensemble network to categorize different risks. Neural network learning identifies targets using several classifiers and combines their outputs to produce reliable results. The system they suggest incorporates AE, BNN, DNN, and extreme learning machines for improved presentation in differentiating between normal and abnormal behaviors. Compared to utilizing a single classifier for detection and their suggested ensemble technique achieves more accurate results.

Lu et al. [14] created Deep Belief Networks (DBN) with Population Extremal Optimization (PEO) for SCADA-based industrial control systems' anomaly detection. The suggested method selects the best neural network settings using PEO. The proposed model is tested using data from the SCADA network traffic for the water storage tank and gas pipeline systems. DDoS assaults transmit malicious queries over the network, which can destabilize the entire IoT system.

A novel advantage cloud framework for edge layer occurrence discovery was put out by Huong et al. [15]. The recommended multi-attack detection method, LocKedge, maintains excellent accuracy while minimizing complexity for deployment in devices with limited resources. Locked was implemented in both federated and centralized learning modes to assess the efficacy of the proposed paradigm from multiple perspectives. The researchers evaluated the performances of the proposed system using the BoT-IoT dataset.

Kim et al. [16] Using the KDD 99 dataset, the DBN beats the SVM and ANN organization representations presently in use. Also they described an impression of a DNN-based intrusion discovery organization that can categorize attacks. The data shows that the suggested model performs better at identifying DoS and probe object classes like R2L and U2R object classes. Ferran et al.[17] described the classification of 35 well-known network datasets into seven classes according to their significance for intrusion detection. Based on actual traffic information, such as CSE CIC-IDS2018 and Bot-IoT, they provide seven presentative representations for the respective category, evaluating and comparing the effectiveness via accuracy and false anxiety rate.

Vinayakumar et al[18]. Claim that the architecture provides real-time internet traffic monitoring and enables system administrators to receive notifications of potentially hazardous network activity. According to predictions, the system would have a reliable and diverse DNN structure and be capable of instantly handling and analysing enormous amounts of data. NSL-KDD and KDD'99 were used as supplementary data sources for the technique's evaluation. On NSL-KDD, the best F-measure for binary classification was 80.7%, and for multiclass type, it was 76.5%.

Based on attack data, Erpek et al. [19] describe a Generative Adversarial Networks(GAN)-based method to recognize and counteract attempts to jam wireless communications. A receiver and a jammer make up their model. The jammer gathers station state and ACKs to develop a classifier to anticipate the subsequent transmission and successfully block it. In contrast, the transmitter utilizes a pretrained classifier to forecast the present station state and select whether to direct founded on the most current sensing outcomes. The jammer using the classification score, controls the power under the average power restriction.

For various cybersecurity applications, Yousefi-Azar et al. [20] provide learning feature representation which includes two training phases: pretraining and fine-tuning. Finding a good place to start for the fine-tuning stage is the goal of the first step. The fine-tuning phase will provide feature descriptions for the contribution information once the pretraining step determines the parameters. Their recommended feature learning method can significantly decrease feature sizes, which lowers storage necessities.

Javaid et al. [21] used scant Auto Encoder(AE), a softmax-regression layer and Self-Taught Learning (STL) to create their models. The suggested STL may be divided into two parts with softmax regression employed for classification following feature extraction and sparse AE. The application of STL could significantly enhance a created network's learning ability when faced with unexpected threats because unique groupings of occurrences can be incrementally analysed throughout real-time without the difficulties of training from scratch.

Farahnakian et al. [22] construct classification models to identify anomalous behaviors while concentrating on essential and instructive feature representations using a deep-stacked autoencoder. Using four AEs in sequential instruction, their suggested network will be trained using a greedy layer wise method. The KDDCup99 dataset experiment consequences determine that even in the face of unbalanced data and it can identify anomalies with a high degree of accuracy, 94.71%.

Marteau et al.[23] calculated covering similarity on symbolic sequences to identify attacks from standard system call sequences. They investigated three similarity metrics and showed that protecting similarity in host-based intrusion detection systems is a crucial predictor of an anomaly. A grouping of Support Vector Machines, Particle Swarm Optimization (PSO), and K-Nearest Neighbor(KNN) search were employed by Aburomman et al.[24]. The combination of these techniques significantly improved categorization accuracy. However, the advantage of such a mixture is constrained and cannot be maximized.

Rehman et al. [25] developed a brand-new attack detection technique to distinguish DDoS occurrences in real time. The researchers detected and classified real-time DDoS circumstances on IoT networks using Recurrent Neural Networks (RNN), gated recurrent units, and minimum sequential optimization. Accuracy, precision, recall, and F1 scores are a few performance indicators used to assess the suggested framework.

An original edge-centric ML-based IoT defensive solution against IoT DDoS threats was created by Jia et al. [26]. The suggested method is to locate, recognize, and categorize DDoS attacks in IoT surroundings. The authors used the Slow HTTP test, bones, and the CICDoS2019 dataset to construct a sizable dataset using DDoS simulators. They contrasted the suggested strategy with four popular ML models. Experimental results showed that the recommended technique outclassed current state-of-the-art DDoS assault detection approaches.

A comparison of the various methods is shown in Table 1

Author	Contribution	Methods	Dataset	Limitations
Virupakshar et al.[27]	Socket programming and OpenStack firewall	NB, DNN, KNN, and DT	OpenStack Cloud, KDD-CUP 99	Only detects a limited number of DDoS attacks.
Lian et al.[28]	stacked strategy	DT-RFE	NSLKDD and KDD-CUP 99	U2R could be more accurate.
Gu et al.[29]	The ratio of the logarithmic marginal density	SVM	NSL-KDD	It is challenging to configure for diverse datasets.
Jiang et al.[30]	Data balancing using SMOTE	Deep hierarchical	NSL-KDD & UNSWNB15	SMOTE is used to balance data.
Andresini et al.[31]	Multi-channel for deep feature learning	MINDFUL	2017 KDD-CUP 99, UNSWNB 15, and CICIDS	Low accuracy results from class imbalance.
Alsirhani et al.[32]	Dynamic DDoS attack detection	Fuzzy logic	DDoS assault (T-shark)	Iterations for T time are manually set.
Yao, et al.[33]	Multi-level intrusion detection	MSML	KDD-CUP 99	Optimization for the detection of unknown patterns

Wherever Times is specified, Times Roman or Times New Roman may be used. If neither is available on your word processor, please use the font closest in appearance to Times. Avoid using bit-mapped fonts if possible. True-Type 1 or Open Type fonts are preferred. Please embed symbol fonts, as well, for math, etc.

III. PROPOSED SYSTEM

An individual, a group of people, or an organization could launch a cyberattack which could be related to cyberterrorism or interstate cyber warfare. Many agencies have used cyberattacks in the modern era, including autonomous states, people, businesses, the general public, communities, and gangs. Anyone can also carry out these assaults. A specified target can be stolen, altered, or even destroyed with unauthorized access to a protected network. A cyberattack may aim to do various things from infecting a personal computer with malware to trying to take down an entire country's infrastructure. Control issues are raised because CPS is vulnerable to a substantial number of cyberattacks without displaying any signs of organizational failure. The physical system may become unstable as a result of attacks. CPS is at risk if cyberattacks are launched against it repeatedly and seem to be unsuccessful. If no hardware or software safeguards are protecting the dynamics of the program, the hacker is free to wreak whatever havoc they like. It can be challenging to control systems that have been the victim of cyberattacks, particularly when it comes to electrical systems. Physical and digital attacks may be directed toward CPS. While cyber-physical links are broken by cyber-attacks, which also weaken CPS, physical attacks result in the immediate suspension of dynamic response[34].

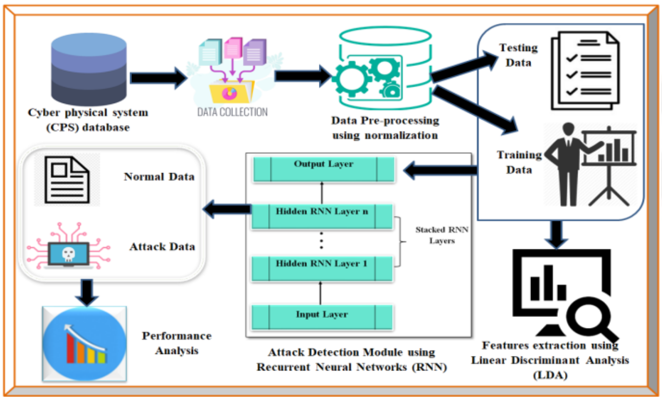
Cyber physical system (CPS) database
DATA COLLECTION
Data Pre-processing using normalization
Testing Data
Training Data
Normal Data
Attack Data
Output Layer
Hidden RNN Layer n
Stacked RNN Layers
Hidden RNN Layer 1
Input Layer
Performance Analysis
Attack Detection Module using Recurrent Neural Networks (RNN)
Features extraction using Linear Discriminant Analysis (LDA)

Figure 2: The proposed method of RNN-LDA

A. *The goal of the study*

CPS, or cyber-physical systems, integrate components from the digital and physical worlds to enhance functionality. As a result of the exponential growth of cyberattacks and threats, more complaints about them are being made. The main culprit is the increased usage of cyber-physical systems (CPS) to supply cutting-edge tools. Concerns concerning the safety of these schemes have increased due to the exponential growth of cyber-physical systems (CPS). New threats, dangers, assaults, and defences are now a part of the CPS's next generation. However, a thorough inquiry into the CPS safety problems has yet to be carried out. Due to the extensive range of CPS systems and components, it has been difficult to explore this subject using a single generalized model. It is challenging because of these two factors together. Creating a suitable CPS architecture is essential since CPS security has emerged as a global problem.

The foremost process as input data is extracted from the CPS database, after which it is normalized to eliminate errors and duplicate entries. The pre-processed data has been splitted into training and testing dataset. This training and testing process will make it easier for the proposed RNN classification algorithm to obtain high accuracy. The characteristics are obtained by utilizing the LDA approach in the CPS database. LDA is can simplify intrusion detection by splitting data into normal attacks and intrusion. LDA is used to analysing the normal classes and cyber-attack classes. Data pre-processing and cleansing of input were performed to meet the criteria for neural networks. The RNN technology is combined with it to optimize the system by including different hidden layer process. Finally, the proposed classifier model RNN extract the features from the NSL-KDD dataset and distribution of cyber-attacks as normal or attacks. The performance evaluation done with by applying different parameters. The efficiency of the suggested technique RNN is designed in Fig.2.

B. *Normalization of Cyber-attack Data*

This paper discusses a methodology to assess the presence of a cyber-attack based on the relationship between attacks, consequences, and cyber-physical parameters using probability theory and mathematical statistics. The likelihood of consequences occurring by violating cyber-physical parameters defines an attack, and changes in cyber-physical parameters indicate the possibility of an attack. Bayes' theorem is used to determine the probability of a particular attack being responsible based on the conditional probabilities of the occurrence of events. The probability of an attack given a change in parameters can be expressed using the equation provided.

Mathematically, Bayes' theorem shows the relationship between the probability of event R and the probability of event S, P(R) and P(S), the conditional probability of the occurrence of event R with existing S and the occurrence of event S with existing R, P(R|S) and P(S|R) . For example, we need to determine the relationship between the probability of an attack, given a change in the parameters. Then, we can express the probability with the following equation:

$$P\left({}^{R_n}/_{S_n}\right) = \frac{P(R_n/\, S_n)P(R_n)}{P\ (S_n)} \qquad (1)$$

where P(Rn) is the a priori probability of the occurrence of an event that is described as an attack, P(Rn|Sn) is the probability of an attack A occurring when parameter P changes (a posteriori probability), P(Rn|Sn) is the probability of changing parameter P when attack R occurs and P(Rn) is the total probability of the occurrence of a change in parameter P. Specifically, we believe a priori that an attack has occurred, and we need to understand which parameters are affected and with what probability they indicate its occurrence. In problems and statistical applications, P(Rn) is usually calculated using the formula for the total probability of an event depending on several inconsistent hypotheses that have a total probability. In our case, as a rule, the attack depends on changing several parameters at once, so it is rational to use the following equation:

$$P\left({}^{S_{in}}/_{R_n}\right) = \frac{P(R_n/\, S_{in})P(R_n)}{\sum_{j=1}^{N} P({}^{R_n}/_{S_{in}})\ P(S_{in})\ P(S_{in})} \qquad (2)$$

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 7
DOI: https://doi.org/10.17762/ijritcc.v11i7.7958
***Article Received**: 05 May 2023 **Revised**: 20 June 2023 **Accepted**: 18 July 2023*

The system is trained using data that distinguishes normal behavior from anomalies, and the classifier uses a set of metrics that are assumed to be independent of each other. The score for each class is calculated, and the classifier chooses the class with the highest score.

C. Recurrent Neural Network

RNNs are a subclass of Artificial Neural Networks that, unlike Feed-forward Neural Networks, have a hidden internal state $r^{(t)}$ that is engaged in the computation at the stage $t+1$ (FNN). RNN is trained to process sequences of varying lengths[35].

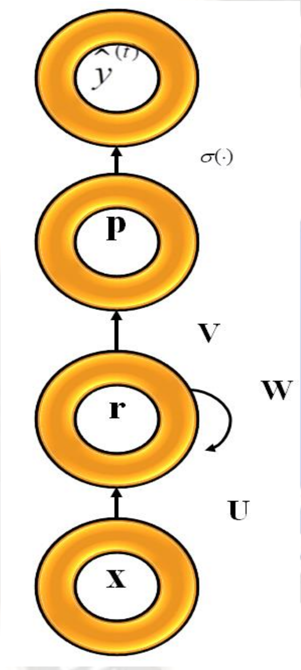

Figure 3: Computation graph of RNN

$$r^{(t)} = \sigma_h(b + Wh^{(t-1)} + Ux^{(t)}) \qquad (3)$$

$$p^{(t)} = c + Vh^{(t)} \qquad (4)$$

$$y^{(t)} = \sigma(p^{(t)}) \qquad (5)$$

Where U, V, and W are the weight matrices for the input-to-hidden, hidden-to-output, and hidden-to-hidden connections, respectively; $r^{(t)}$ is the hidden internal state at the time step $t, \sigma_r(t)$ is the activation function of the hidden state; b and c are the bias vectors for the state and output; $\sigma(\cdot)$ is the activation function of the output; and $y^{(t)}$ is the production at t.

The prototype of cyber-attack classification has the well-known issue where the ascent vanishes through back-propagation through time (BPTT), which makes training challenging. Recently, methods to avoid gradient vanishment and let RNN learn long-term dependability have been included in some RNN versions, including LSTM and Gated Recurrent Unit (GRU). They perform better than the RNN prototype overall. Therefore, we employ stacked RNN to project upcoming measures [36].

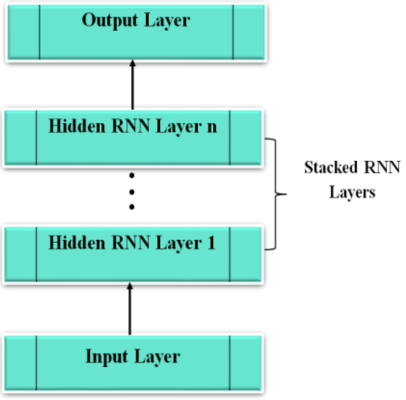

Figure 4: The structure of stacked RNN

D. Network architecture

We utilize the loaded RNN, which comprises many RNN hidden layers, as shown in Fig.4. since a deep neural network architecture performs better than a shallow one. In our example, a hyperbolic tangent function activates hidden layers. A full-connected layer with a linear activation function is placed on these concealed RNN layers. The neural network's depth facilitates extracting high-level temporal patterns from information making it generally more challenging to change the parameter.

i. L1 Regularization:

L1 regularization is the preferred choice when having a high number of features as it provides sparse solutions. Even, we obtain the computational advantage because features with zero coefficients can be avoided. The regression model that uses L1 regularization technique is called Lasso Regression.

For instance, we define the simple linear regression model Y with an independent variable to understand how L1 regularization works.

For this model, W and b represents "weight" and "bias" respectively, such as

$$W = w_1, w_2, w_3 \ldots \ldots . w_n \qquad (6)$$

And,

$$b = b_1, b_2, b_3 \ldots \ldots . b_n \qquad (7)$$

And Ŷ is the predicted result such that

$$Y^{\wedge} = w_1x_1 + w_2x_2 + w_3x_3 \ldots \ldots w_nx_n + \mathrm{b} \qquad (8)$$

The below function calculates an error without the regularization function

Loss = E (Y, Y^) (9)

And function that can calculate the error with L1 regularization function,

Where **λ** is called the regularization parameter and **λ**> 0 is manually tuned. Also, **λ**=0 then the above loss function acts as Ordinary Least Square where the high range value push the coefficients (weights) 0 and hence make it underfits.

ii. L2 regularization:

L2 regularization can deal with the multi-collinearity (independent variables are highly correlated) problems through constricting the coefficient and by keeping all the variables. L2 regression can be used to estimate the significance of predictors and based on that it can penalize the insignificant predictors. A regression model that uses L2 regularization techniques is called Ridge Regression.

Here, **λ** is known as Regularization parameter, also if the lambda is zero, this again would act as OLS, and if lambda is extremely large, it leads to adding huge weights and yield as underfitting. Substituting the formula of Gradient Descent optimizer for calculating new weights;

E. Linear Discriminant Analysis(LDA)

A well-known arithmetical technique called LDA is frequently used as a dimensionality decrease tool in machine learning and design appreciation presentations. LDA reduces an n-dimensional dataset into a lesser k-dimensional dataset though maintaining the important class discrimination data (kn). But in this study, we employ the LDA as an organization method and investigate how it might be used to grow an intrusion discovery perfectly [37]. Instead of using the LDA as a feature decrease method. LDA offers several useful features, making it a fantastic technique for creating intrusion detection models. To begin with, it is clear-cut and easy to use. Second, it is more effective and requires less calculation. The LDA-based model, in the end, performs better than several other well-known intrusion-finding methods. We define the LDA's essential operation in this division and offer an illustration.

Take a look at a dataset with k different class labels. Let $S = \{S_1, S_2, S_3, ..., S_k\}$ represent the collection of these k classes. We describe the period matrix Slc_i for period $S_i \in S$ as follows if the dataset is d-dimensional (without the period make):

$$Sl_{ci} = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1d} \\ a_{21} & a_{22} & \cdots & a_{2d} \\ \cdots & \cdots & \cdots & \cdots \\ a_{n1} & a_{n2} & \cdots & a_{nd} \end{bmatrix} \quad (10)$$

Wherein every row admission in the $Slc_i \in Class\ S_i$ period matrix. The period matrix Slc_i mean vector is identified by the following symbols and is an array made up of the means of each of its column vectors:

$$\mu_{Slc_i} = [m_{a1} \;\; m_{a2} \;\; m_{a3} \;\; \cdots \;\; m_{ad}] \quad (11)$$

Where m_{a1} is the class matrix Slc_i mean for the i-th column attribute. The worldwide unkind vector is. Therefore the average of all of the period means vectors, which is represented by:

$$\mu = \frac{1}{k}\sum_{i=1}^{k} \mu Sl_{c_i} \quad (12)$$

A mean-adjusted class matrix is then designed. This is the period matrices whose columns components for each row are deducted from the corresponding area defined by the worldwide mean vector (μ) and is represented by:

$$Slc_i^{mc} = Sl_{ci}[a][b] - \mu[b] \quad (13)$$

Where the row and support directories of the period matrix Slc_i are represented by a = 1, 2,..., n and b = 1, 2,..., d, respectively. The class matrix Slc_i a covariance matrix is therefore defined as

$$Slc_i^{\text{cov}} = \frac{Slc_i^{mc^T} * Slc_i^{mc}}{n_i} \quad 14)$$

Where in the mean corrected class matrix Slc_i^{mc} , $Slc_i^{mc^T}$ and n_i , respectively,. stand for the transposition and the number of row entrances. The combined collection covariance matrix of the dataset is distinct as follows if there are k classes in the dataset:

$$S = \frac{1}{N}\left(\sum_{j=1}^{k} n_j Slc_j^{\text{cov}}\right); \;\; where\ N = \sum_{j=1}^{k} n_j \quad (15)$$

We now generate the linear discriminant function (LDF) of each of the k classes to classify a new data item instance

***Article Received**: 05 May 2023 **Revised**: 20 June 2023 **Accepted**: 18 July 2023*

$x = \{x_1, x_2, \ldots x_d\}$ into one of the k classes. The LDF for class S_i is as follows:

$$f_i = \mu_{Slc_i} S^{-1} x^T - 0.5_{\mu Slc_i} S^{-1} \mu_{Slc_i}^T + \ln(p_i) \qquad (16)$$

S is the pooled subgroup covariance matrix, and C1 is its inverse. x^T and $\mu_{Slc_i}^T$ are the transfers of the category mean direction μ_{Slc_i} and the input data points vector x, respectively. The category with the maximum LDF assessment receives the data point x.

IV. RESULTS AND DISCUSSION

A. *Simulation Environment*

The suggested IoT attack classification and detection system were developed, tested, and evaluated using the NSL-KDD dataset of significant assaults on IoT communication. It was found that the classifier model also contained two (for binary assault discovery) or five categories (multi-attack organization). Python libraries are used to implement the suggested system. A high-performance computing environment was used to assess the multicore architecture, CPU, and graphical dispensation organization effectiveness of the NVIDIA GeForce® Quadro P2000 graphics card (GPU).

B. *Performance Metrics*

This section introduces and assesses the proposed RNN-LDA model outcomes and its evaluation. Self-organizing incremental neural networks (SOINN), multi-layer perceptron neural networks (MLP), deep neural networks (DNN), convolutional neural networks (CNN), and residual neural networks are the three types of classifiers used in the studies' lowest-layer classifiers (ResNet-50).

A "true positive" (TP) is an occurrence that we predicted would happen and whose yield occurred as expected.

- True Negatives (TN): When we anticipated something would be false, it was.
- False Negatives (FN) are instances where we expected an accurate result, but the actual yield was likewise incorrect.
- A false negative is when we anticipated a false result but received a correct one (FN).
- The percentage of accurate predictions the classifier makes represents its accuracy. It details how well the classifier performed overall. According to its definition,

$$Accuracy = \frac{TP + TN}{TP + FN + FP + TN} \qquad (17)$$

Precision is the proportion of accurately predicted positive results to all anticipated positive observations. Precision and an extremely low False Positive Recall are connected.

$$\Pr ecision = \frac{TP}{TP + FP} \qquad (18)$$

Recall, which measures the fraction of correctly categorized positives, is another helpful evaluation statistic. The recall is calculated using the TP and FP values.

$$\mathrm{Re}\, call = \frac{TP}{TP + FN} \qquad (19)$$

The F1-score is calculated using the precision and recall weighted average. It acts as a statistical metric for evaluating how effectively the classifier works. Both false positives and false negatives are taken into account in this ranking.

$$F - Score = 2 * \frac{precision * recall}{precision + recall} \qquad (20)$$

"Root Mean Squared Error" is the square root of the average error between the actual data and the anticipated data (RMSE). To calculate its value, you can use formulas (35).

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\left(x_i - x'_i\right)^2} \qquad (21)$$

Where the forecast sample number is n; the value speed at the time i is x_i; the forecasted value at the time is x'_i; and the actual value at the time i is the average of x_i.

i. *Precision Analysis*

Table 2: Precision Analysis for RNN-LDA method with existing systems

Number of data from Dataset	SOINN	MLP	DNN	CNN	ResNet-50	RNN-LDA
1000	88.637	80.324	79.632	86.435	82.536	90.228
2000	88.425	81.526	81.532	87.452	83.627	90.536
3000	88.926	80.528	79.524	86.920	82.298	91.652
4000	89.926	81.327	79.063	86.213	82.738	92.873
5000	90.325	81.732	80.637	87.338	83.927	91.932
6000	89.425	82.653	80.937	87.228	84.325	92.536

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 7
DOI: https://doi.org/10.17762/ijritcc.v11i7.7958
***Article Received**: 05 May 2023 **Revised**: 20 June 2023 **Accepted**: 18 July 2023*

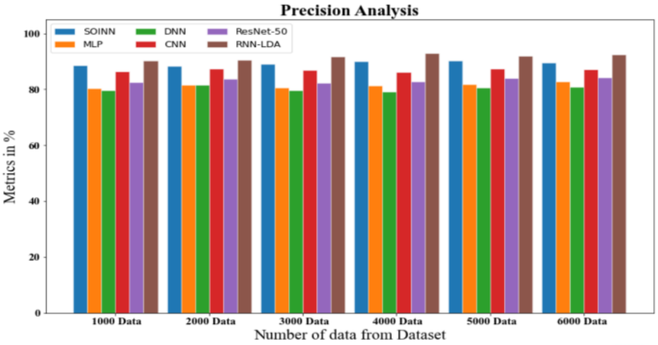

Figure 5: Precision Analysis for RNN-LDA method with existing systems

A precision comparison of the RNN-LDA technique with different well-known approaches is shown in Fig. 5 and Tab. 2. The graph shows how better a precision performance was obtained using the deep learning approach. In contrast to the SOINN, MLP, DNN CNN, and ResNet-50 models, that have precisions of 88.637%, 80.324%, 79.632%, 86.435%, and 82.536%, respectively, RNN-LDA has a precision of 90.228% while using 1000 data. The RNN-LDA model, however, performed admirably with various data sizes. RNN-LDA has a precision of 92.536% for 6000 data, while SOINN, MLP, DNN CNN, and ResNet-50 have precision values of 89.425%, 82.653%, 80.937%, 87.228%, and 84.325%, respectively.

ii. Recall Analysis

Table 3: Recall Analysis for RNN-LDA method with existing systems

Number of data from Dataset	SOINN	MLP	DNN	CNN	ResNet-50	RNN-LDA
1000	88.536	91.324	83.526	84.829	87.213	96.435
2000	88.072	91.627	83.928	84.953	86.023	96.223
3000	89.452	92.763	84.229	85.435	86.536	95.425
4000	90.322	92.733	85.083	85.627	86.220	95.053
5000	91.832	93.403	85.227	87.936	87.186	95.326
6000	90.728	94.756	84.627	87.425	87.637	94.652

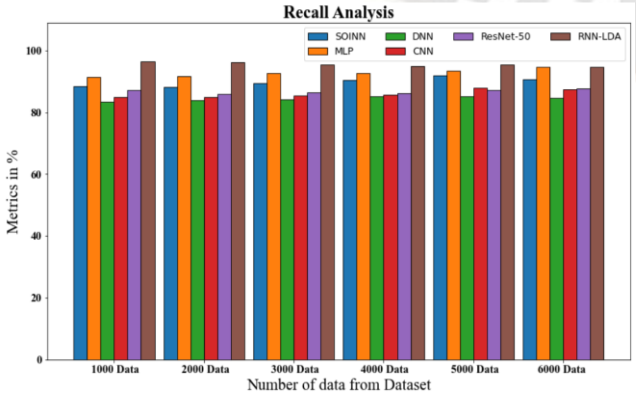

Figure 6: Recall Analysis for RNN-LDA method with existing systems

The RNN-LDA methodology is compared to other widely used methods in Fig. 6 and Tab. 3. The graph demonstrates how the deep learning approach has an improved recall performance. For instance, the RNN-LDA model, with 1000 data, has a recall of 96.435%, whereas the SOINN, MLP, DNN CNN, and ResNet-50 models have recall values of 88.536%, 91.324%, 83.526%, 84.829%, and 87.213%, respectively. With various data sizes, the RNN-LDA model nevertheless worked effectively. RNN-LDA has a recall of 94.652%, while SOINN, MLP, DNN CNN, and ResNet-50 models have recall values of 90.728%, 94.756%, 84.627%, 87.425%, and 87.637% under 6000 data, respectively.

iii. F-Score Analysis

Table 4: F-Score Analysis for RNN-LDA method with existing systems

Number of data from Dataset	SOINN	MLP	DNN	CNN	ResNet-50	RNN-LDA
1000	92.637	89.738	93.727	87.819	85.435	97.536
2000	92.397	90.425	94.637	87.425	86.935	97.425
3000	91.092	89.231	95.225	88.182	87.324	98.026
4000	91.526	89.062	94.827	87.927	87.027	97.324
5000	91.324	89.526	94.673	87.535	86.526	97.226
6000	91.028	89.425	95.781	88.652	86.103	98.536

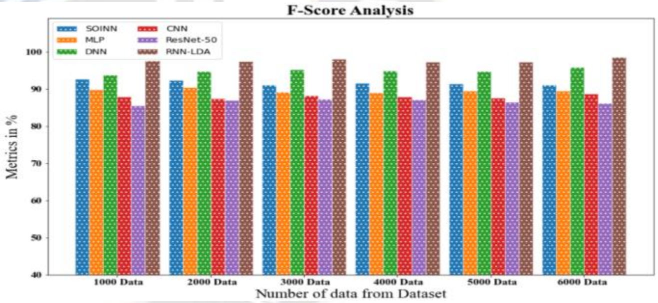

Figure 7: F-Score Analysis for RNN-LDA method with existing systems

An f-score comparison of the RNN-LDA approach with several established methods is shown in Fig. 7 and Tab. 4. The graph shows how the deep learning approach has an improved f-score performance. Compared to the SOINN, MLP, DNN CNN, and ResNet-50 models, which have f-scores of 92.637%, 89.738%, 93.727%, 87.819%, and 85.435%, respectively, RNN-LDA has an f-score of 97.536% with 1000 data. The RNN-LDA model, however, performed admirably with various data sizes. The f-score for RNN-LDA under 6000 data is 98.536%, whereas those for SOINN, MLP, DNN CNN, and ResNet-50 are 91.028%, 89.425%, 95.781%, 88.652%, and 86.103%, respectively.

iv. *Accuracy Analysis*

Table 5: Accuracy Analysis for RNN-LDA method with existing systems

Number of data from Dataset	SOINN	MLP	DNN	CNN	ResNet-50	RNN-LDA
1000	97.036	88.526	91.652	94.213	89.952	97.636
2000	96.202	89.062	91.920	94.827	90.425	98.636
3000	96.526	88.425	93.627	94.435	90.218	98.435
4000	96.213	88.213	94.637	95.637	92.435	98.219
5000	96.942	89.637	93.324	95.229	92.763	97.926
6000	96.231	89.435	93.902	96.536	91.325	99.821

The accuracy of the RNN-LDA approach is contrasted with that of other methods in Fig. 8 and Tab. 5. The graph shows that the deep learning strategy produced higher performance with accuracy. Compared to the SOINN, MLP, DNN CNN, and ResNet-50 models, that have an accuracy of 97.036%, 88.526%, 91.652%, 94.213%, and 89.952%, respectively, RNN-LDA has an accuracy of 97.636% while using 1000 data. The RNN-LDA model, however, performed admirably with various data sizes. Like RNN-LDA, SOINN, MLP, DNN CNN, and ResNet-50 models have accuracy scores of 96.231%, 89.435%, 93.902%, 96.536%, and 91.325%, respectively, under 6000 data while RNN-LDA technique has an accuracy of 99.821%.

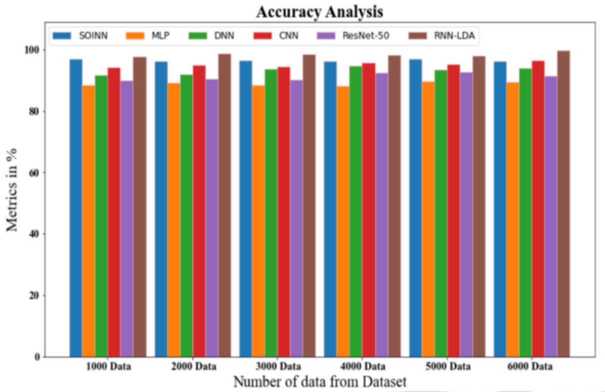

Figure 8: Accuracy Analysis for RNN-LDA method with existing systems

v. *Processing Time*

Table 6: Processing Time Analysis for RNN-LDA method with existing systems

Number of data from Dataset	SOINN	MLP	DNN	CNN	ResNet-50	RNN-LDA
1000	7.832	6.536	4.836	3.726	2.636	1.076
2000	7.425	6.324	4.732	3.963	2.038	1.452
3000	7.063	5.926	5.029	3.224	2.137	1.635
4000	7.213	5.065	5.432	4.029	2.541	1.906
5000	9.637	6.625	5.201	4.526	3.028	2.038
6000	8.536	6.063	5.139	4.213	3.627	2.536

In Tab.6 and Fig.9, the processing time comparison of the RNN-LDA methodology with existing methods is shown. The data shows that the RNN-LDA method has outperformed the alternative ways in every aspect. For instance, the RNN-LDA method has processed 1000 data in 1.076 seconds as opposed to 7.832 seconds, 6.536 seconds, 4.836 seconds, 3.726 seconds, and 2.636 seconds for existing methods such as SOINN, MLP, DNN CNN, and ResNet-50 respectively. Similarly, for processing 6000 data with the RNN-LDA method takes 2.536 seconds, whereas SOINN, MLP, DNN CNN, and ResNet-50 methods require 8.536 seconds, 6.063 sec, 5.139 seconds, 4.213 seconds, and 3.627 seconds, respectively.

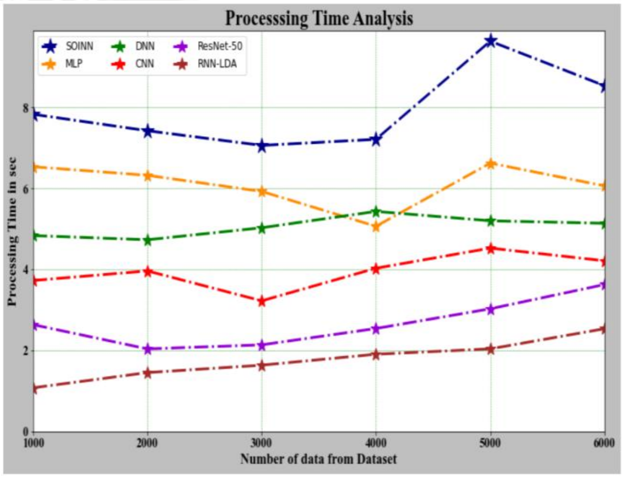

Figure 9: Processing Time Analysis for RNN-LDA method with existing systems

vi. *Training and Testing Validation*

Table 7: Training and Testing validation Analysis for RNN-LDA with the existing system

Epochs	Train	Test
0	1.53	1.34
5	1.33	1.13
10	1.09	0.83
15	0.75	0.67
20	0.65	0.53
25	0.57	0.45
30	0.49	0.35
35	0.24	0.21
40	0.15	0.19
45	0.12	0.14
50	0.07	0.09

In Tab.7 and Fig.10, the RNN-LDA technique's Training and Testing Validation Analysis is demonstrated with existing systems. In all aspects, the proposed RNN-LDA technique performed brilliantly, for the given data. The RNN-LDA's training and testing validation times with five epochs are 1.33 and 1.13, respectively. Similarly, after 50 epochs, the RNN-LDA's training and testing validation coefficients are 0.07 and 0.09, respectively. The proposed method performs best with minimum loss.

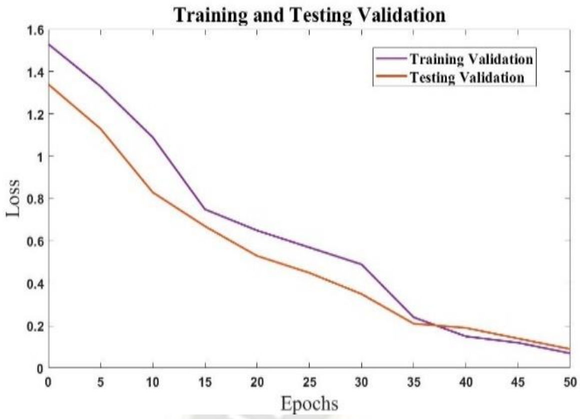

Figure 10: Training and Testing validation Analysis for RNN-LDA with the existing system

vii. *RMSE Analysis*

Table 8: RMSE Analysis for RNN-LDA method with existing systems

Number of data from Dataset	SOINN	MLP	DNN	CNN	ResNet-50	RNN-LDA
1000	26.452	25.832	22.536	21.827	18.926	16.827
2000	27.907	25.393	22.102	21.072	19.425	17.636
3000	27.627	27.602	24.736	21.627	19.324	16.029
4000	29.272	27.435	24.324	22.838	19.826	16.325
5000	31.527	26.187	24.029	23.793	21.652	17.425
6000	30.627	26.762	23.726	22.632	20.627	18.627

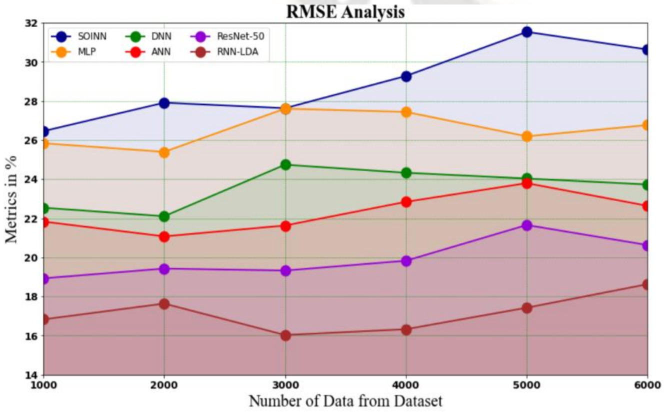

Figure 11: RMSE Analysis for RNN-LDA method with existing systems

The RMSE comparison of the RNN-LDA approach with several known methods is shown in Fig.11 and Tab.8. The graph shows that the deep learning technique yielded superior outcomes with reduced RMSE values. RNN-LDA, for example, has an RMSE of 16.827% with 1000 data, whereas the SOINN, MLP, DNN CNN, and ResNet-50 models have slightly higher RMSEs of 26.452%, 25.832%, 22.536%, 21.827%, and 18.926%, respectively. The RNN-LDA model, on the other hand, has exhibited maximum performance with low RMSE values for various data sizes. Similarly, the RMSE value of RNN-LDA under 6000 data is 18.627%, whereas SOINN, MLP, DNN CNN, and ResNet-50 models have RMSEs of 30.627%, 26.762%, 23.726%, 22.632%, and 20.627%, respectively.

viii. *Dataset Accuracy Comparison Analysis*

Table 10: Dataset Accuracy Comparison Analysis

Dataset	Accuracy Values
NSL-KDD(Wang et al. [38])	99.28
NSL-KDD (Singh et al. [39])	98.66
NSL-KDD (Bamakan et al. [40])	98.30
NSL-KDD Dataset	99.821

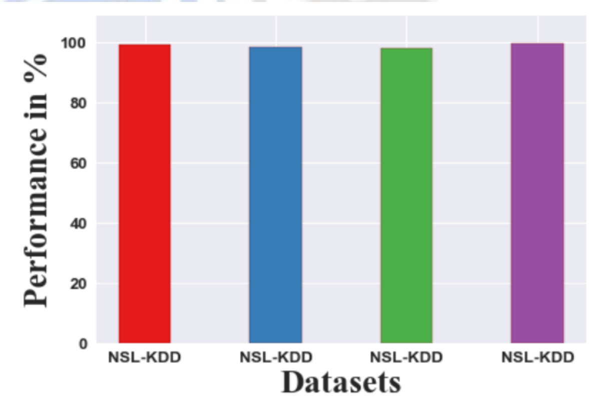

Figure 12: Dataset Accuracy Comparison Analysis

V. Conclusion

The main goal of this initiative was to apply AI and machine learning technologies to detect cyber-attacks on physical systems as soon as possible. The key assault categories are the subject of this work (DDoS, Probe, U2R, R2L). As a result, we provide a novel Recurrent Neural Networks (RNN) cyberattack detection system that integrates AI and machine learning in this study (ML). The Network Security Laboratory-Knowledge Discovery Databases (NSL-KDD) dataset, which includes all of the significant risks, was used to assess the created system in this case. The dataset was normalized beforehand to remove errors and erroneous information. Linear Discriminant Analysis (LDA) is used to extract the features (LDA). Because the RNN-LDA method is so good at resolving sequence problems, time series prediction, speech recognition, text generation, machine translation, image description generation, handwriting recognition, and other issues, it was chosen for this work. A model called RNN-LDA has been proposed for learning time-ordered orders of network flow traffic and evaluating how well it can spot unusual activity. Existing models like Self-organizing incremental neural network (SOINN), Multi-layer perceptron (MLP) neural network, Deep neural network (DNN), Convolutional neural network (CNN), and ResNet-50 had little impact on predictive performance, with the suggested framework winning out with an overall accuracy of 97.321% in determining whether a user will belong to a particular group.The sensors that compose the technical system will use the methodologies outlined in this study to collect new data. It will then be compared to various ways of notion generation. Following the construction of a robust prediction model, the researchers will devise action plans for staying safe in potentially hazardous settings.

REFERENCES

[1] W. Duo, M. Zhou, and A. Abusorrah, “A Survey of Cyber Attacks on Cyber Physical Systems: Recent Advances and Challenges,” IEEE/CAA Journal of Automatica Sinica, vol. 9, no. 5, pp. 784–800, May 2022.

[2] C. Kwon and I. Hwang, “Reachability Analysis for Safety Assurance of Cyber-Physical Systems Against Cyber Attacks,” IEEE Transactions on Automatic Control, vol. 63, no. 7, pp. 2272–2279, Jul. 2018.

[3] S. Tepjit, I. Horváth, and Z. Rusák, “The state of framework development for implementing reasoning mechanisms in smart cyber-physical systems: A literature review,” Journal of Computational Design and Engineering, vol. 6, no. 4, pp. 527–541, Apr. 2019.

[4] M. Yildirim, “Artificial Intelligence-Based Solutions for Cyber Security Problems,” Artificial Intelligence Paradigms for Smart Cyber-Physical Systems, pp. 68–86, 2021.

[5] H. Yang, K. Zhan, M. Kadoch, Y. Liang, and M. Cheriet, “BLCS: Brain-Like Distributed Control Security in Cyber Physical Systems,” IEEE Network, vol. 34, no. 3, pp. 8–15, May 2020.

[6] R. Prasad and V. Rohokale, “Artificial Intelligence and Machine Learning in Cyber Security,” Cyber Security: The Lifeline of Information and Communication Technology, pp. 231–247, Oct. 2019.

[7] Ahmed, M., Mahmood, A.N., Hu, J.: ‘A survey of network anomaly detection techniques’, J. Netw. Comput. Appl., 2016, 60, pp. 19–31

[8] R. Kumar and R. Tripathi, “DBTP2SF: A deep blockchain-based trustworthy privacy-preserving secured framework in industrial Internet of Things systems,” Trans. Emerg. Telecommun. Technol., vol. 32, no. 4, 2021, Art. no. e4222.

[9] S. Latif, Z. Idrees, J. Ahmad, L. Zheng, and Z. Zou, “A blockchain-based architecture for secure and trustworthy operations in the industrial Internet of Things,” J. Ind. Inf. Integr., vol. 21, 2021, Art. no. 100190.

[10] Singh, K., Dhindsa, K. S., & Nehra, D. (2020). T-CAD: A threshold based collaborative DDOS attack detection in multiple autonomous systems. Journal of Information Security and Applications, 51, 102457.

[11] Sharma, A., Agrawal, C., Singh, A., & Kumar, K. (2020). Real-Time DDOS Detection Based on Entropy Using Hadoop Framework. In Computing in Engineering and Technology (pp. 297-305).

[12] D. K. Jain, S. K. S. Tyagi, M. Prakash and L. Natrayan, "Metaheuristic Optimization-Based Resource Allocation Technique for Cybertwin-Driven 6G on IoE Environment," in *IEEE Transactions on Industrial Informatics*, vol. 18, no. 7, pp. 4884-4892, July 2022, doi: 10.1109/TII.2021.3138915.

[13] S. A. Ludwig, “Intrusion detection of multiple attack classes using a deep neural net ensemble,” in Proceedings of 2017 IEEE Symposium Series on Computational Intelligence (SSCI), IEEE, Honolulu,HI, USA, November 2017.

[14] K.-D. Lu, G.-Q. Zeng, X. Luo, J. Weng, W. Luo, and Y. Wu, “Evolutionary deep belief network for cyber-attack detection in industrial automation and control system,” IEEE Trans. Ind. Informat., vol. 17, no. 11, pp. 7618–7627, Nov. 2021.

[15] T. T. Huong et al., “Lockedge: Low-complexity cyberattack detection in IoT edge computing,” IEEE Access, vol. 9, pp. 29696–29710, 2021

[16] J. Kim, N. Shin, S. Y. Jo, and S. H. Kim, “Method of intrusion detection using deep neural network,” in 2017 IEEE International Conference on Big Data and Smart Computing (BigComp), pp. 313–316, Jeju, 2017.

[17] M. A. Ferrag, L. Maglaras, S. Moschoyiannis, and H. Janicke, “Deep learning for cyber security intrusion detection: approaches, datasets, and comparative study,” Journal of Information Security and Applications, vol. 50, p. 102419

[18] R. Vinayakumar, M. Alazab, K. P. Soman, P. Poornachandran, A. Al-Nemrat, and S. Venkatraman, “Deep learning approach for intelligent intrusion detection system,” IEEE Access, vol. 7, pp. 41525–41550, 2019.

[19] T. Erpek, Y. E. Sagduyu, and Y. Shi, “Deep learning for launching and mitigating wireless jamming attacks,” IEEE Transactions on Cognitive Communications and Networking, vol. 5, no. 1, pp. 2–14, 2018

[20] M. Yousefi-Azar, V. Varadharajan, L. Hamey, and U. Tupakula, “Autoencoder-based feature learning for cyber security applications,” in Proceedings of 2017 International Joint Conference on Neural Networks (IJCNN), IEEE, San Diego, CA, USA, pp. 3854–3861, June 2017.

[21] A. Javaid, Q. Niyaz, W. Sun, and M. Alam, “A deep learning approach for network intrusion detection system,” in Proceedings of the 9th EAI International Conference on Bio-Inspired Information and Communications Technologies (formerly BIONETICS), pp. 21–26, New York, NY, USA, December 2016.

[22] F. Farahnakian and J. Heikkonen, “A deep auto-encoder based approach for intrusion detection system,” in Proceedings of 2018 20th International Conference on Advanced Communication Technology (ICACT), IEEE, Chuncheon, South Korea, pp. 178–183, July 2018.

[23] Marteau, P.F. Sequence covering for efficient host-based intrusion detection. IEEE Trans. Inf. Forensics Secur. 2018, 14, 994–1006.

[24] Aburomman, A.A.; Reaz, M.B.I. A novel SVM-kNN-PSO ensemble method for intrusion detection system. Appl. Soft Comput. 2016, 38, 360–372.

[25] S. ur Rehman et al., “DIDDOS: An approach for detection and identification of distributed denial of service (DDoS) cyberattacks using gated recurrent units (GRU),” Future Gener. Comput. Syst., vol. 118, pp. 453–466, 2021.

[26] Y. Jia, F. Zhong, A. Alrawais, B. Gong, and X. Cheng, “Flowguard: An intelligent edge defense mechanism against IoT DDoS attacks,” IEEE Internet Things J., vol. 7, no. 10, pp. 9552–9562, Oct. 2020.

[27] Virupakshar, K.B.; Asundi, M.; Channal, K.; Shettar, P.; Patil, S.; Narayan, D. Distributed denial of service (DDoS) attacks detection system for OpenStack-based private cloud. Procedia Comput. Sci. 2020, 167, 2297–2307.

[28] Lian, W.; Nie, G.; Jia, B.; Shi, D.; Fan, Q.; Liang, Y. An Intrusion Detection Method Based on Decision Tree-Recursive Feature Elimination in Ensemble Learning. Math. Probl. Eng. 2020, 2020, 2835023.

[29] Gu, J.; Wang, L.; Wang, H.; Wang, S. A novel approach to intrusion detection using SVM ensemble with feature augmentation. Comput. Secur. 2019, 86, 53–62.

[30] Jiang, K.; Wang, W.; Wang, A.; Wu, H. Network intrusion detection combined hybrid sampling with deep hierarchical network. IEEE Access 2020, 8, 32464–32476.

[31] Andresini, G.; Appice, A.; Di Mauro, N.; Loglisci, C.; Malerba, D. Multi-channel deep feature learning for intrusion detection. IEEE Access 2020, 8, 53346–53359.

[32] Alsirhani, A.; Sampalli, S.; Bodorik, P. DDoS detection system: Using a set of classification algorithms controlled by fuzzy logic system in apache spark. IEEE Trans. Netw. Serv. Manag. 2019, 16, 936–949. [CrossRef]

[33] Yao, H.; Fu, D.; Zhang, P.; Li, M.; Liu, Y. MSML: A novel multilevel semi-supervised machine learning framework for intrusion detection system. IEEE Internet Things J. 2018, 6, 1949–1959.

[34] Sharath, M.N., Rajesh, T.M. & Patil, M. Design of optimal metaheuristics-based pixel selection with homomorphic encryption technique for video steganography. Int. j. inf. technology. 14, 2265–2274 (2022). https://doi.org/10.1007/s41870-022-01005-9

[35] Zhenwei Zhao, Xiaoming Li, Bing Luan, Weining Jiang, Weidong Gao, Secure Internet of Things (IoT) using a Novel Brooks Iyengar Quantum Byzantine Agreement-centered blockchain Networking (BIQBA-BCN) Model in Smart Healthcare, Information Sciences, 2023,https://doi.org/10.1016/j.ins.2023.01.020

[36] Alharbi, M.,Gupta, S. et al. Mobility aware load balancing using Kho–Kho optimization algorithm for hybrid Li-Fi and Wi-Fi network. Wireless Networks (2023). https://doi.org/10.1007/s11276-022-03225-0

[37] D. Paulraj, P. Ezhumalai & M. Prakash (2022) A Deep Learning Modified Neural Network(DLMNN) based proficient sentiment analysis technique on Twitter data, Journal of Experimental & Theoretical Artificial Intelligence, DOI: 10.1080/0952813X.2022.2093405

[38] Harinder Singh, D. Ramya, Nayani Sateesh, Rohit Anand, Swarnjit Singh, Artificial intelligence-based quality of transmission predictive model for cognitive optical networks, Optik, Vol. 257, 2022,https://doi.org/10.1016/j.ijleo.2022.168789

[39] B.T. Geetha, P. Santhosh Kumar, B. Sathya Bama, Chiranjit Dutta, D. Vijendra Babu, Green energy aware and cluster-based communication for future load prediction in IoT, Sustainable Energy Technologies and Assessments, Vol.52,2022,102244, https://doi.org/10.1016/j.seta.2022.102244.