STUDY OF MACHINE AND DEEP LEARNING ALGORITHMS FOR INTRUSION DETECTION IN IoT

A PROJECT REPORT

SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE AWARD OF THE DEGREE
OF

MASTER OF TECHNOLOGY
IN
ARTIFICIAL INTELLIGENCE

Submitted by

VINEET TOMAR

(2K21/AFI/25)

Under the supervision of

Dr. Pawan Singh Mehra

DEPARTMENT OF COMPUTER SCIENCE ENGINEERING
DELHI TECHNOLOGICAL UNIVERSITY
(Formerly Delhi College of Engineering)
Bawana Road, Delhi 110042

MAY, 2023

DEPARTMENT OF COMPUTER SCIENCE ENGINEERING
DELHI TECHNOLOGICAL UNIVERSITY
(Formerly Delhi College of Engineering)
Bawana Road, Delhi-110042

CANDIDATE'S DECLARATION

I, **Vineet Tomar**, Roll No – **(2K21/AFI/25)** student of M.Tech (**Department of Computer Science Engineering**), hereby declare that the project Dissertation titled "**MACHINE AND DEEP LEARNING ALGORITHMS FOR INTRUSION DETECTION IN IoT**" which is submitted by me to the **Department of Computer Science Engineering**, Delhi Technological University, Delhi in partial fulfilment of the requirement for the award of degree of Master of Technology, is original and not copied from any source without proper citation. This work has not previously formed the basis for the award of any Degree, Diploma Associateship, Fellowship or other similar title or recognition.

Place: Delhi **Vineet Tomar**

Date: 31.05.2023

DEPARTMENT OF COMPUTER SCIENCE ENGINEERING
DELHI TECHNOLOGICAL UNIVERSITY
(Formerly Delhi College of Engineering)
Bawana Road, Delhi-110042

CERTIFICATE

I hereby certify that the Project Dissertation titled "**MACHINE AND DEEP LEARNING ALGORITHMS FOR INTRUSION DETECTION IN IoT**" which is submitted by **Vineet Tomar**, Roll No – **(2K21/AFI/25)**, **Department of Computer Science Engineering**, Delhi Technological University, Delhi in partial fulfilment of the requirement for the award of the degree of Master of Technology, is a record of the project work carried out by the student under my supervision. To the best of my knowledge this work has not been submitted in part or full for any Degree or Diploma to this University or elsewhere.

Place: Delhi **Dr. Pawan Singh Mehra**

Date: 31.05.2023 **SUPERVISOR**

DEPARTMENT OF COMPUTER SCIENCE ENGINEERING
DELHI TECHNOLOGICAL UNIVERSITY
(Formerly Delhi College of Engineering)
Bawana Road, Delhi-110042

ACKNOWLEDGEMENT

I wish to express my sincerest gratitude to **Dr. Pawan Singh Mehra** for his continuous guidance and mentorship that he provided me during the project. He showed me the path to achieve my targets by explaining all the tasks to be done and explained to me the importance of this project as well as its industrial relevance. He was always ready to help me and clear my doubts regarding any hurdles in this project. Without his constant support and motivation, this project would not have been successful.

Place: Delhi **Vineet Tomar**

Date: 31.05.2023 **(2K21/AFI/25)**

Abstract

In this era of exponential internet boom IoT devices are also increasing with a rapid growth. This rapid growth also increases the risk of intrusion such as phishing at application layer, Dos & spoofing at network layer and node capture, malicious code injection & eavesdropping at physical layer. So, to prevent systems from these attacks it has become the desired need of time to implement an Intrusion Detection system model. In this paper we have briefly compared various global datasets and used most recent CSECICIDS-2018 dataset having 1.04 million of samples. We have implemented a Bi-LSTM model having an Input layer, a reshape layer, two Bi-LSTM layers, a dense layer, a dropout layer and lastly an output layer. Proposed model is used for the prediction of a packet whether it is Benign and Not Benign using 11 important features 'Timestamp', 'Fwd Pkt Len Std', 'Fwd Pkt Len Mean', 'Fwd Pkt Len Max', 'Fwd Seg Size Avg', 'Pkt Len Std', 'Flow IAT Std', 'Bwd Pkt Len Std', 'Bwd Seg Size Avg', 'Pkt Size Avg', 'Subflow Fwd Byts' for training the model.

This Bi-LSTM model have provided an accuracy of 99.554%, precision of 99.227% and F1 score of 99.612%. Further this model can be tested and improved on real-time intrusion scenario to provide improved results.

Keywords: IoT, IoT security, Deep learning, Machine learning, Intrusion Detection, Bi-LSTM, CSECICIDS-2018

Contents

List of Tables

List of Figures

List of Abbreviations

ML	-	Machine learning
DL	-	Deep learning
IoT	-	Internet of things
IDS	-	Intrusion detection system
CNN	-	Convolutional neural network
NLP	-	Natural language processing
LSTM	-	Long-Short Term Memory
Bi-LSTM	-	Bi directional Long-Short term memory
SVM	-	Support vector machine
ANN	-	Artificial neural network
AI	-	Artificial intelligence
RNN	-	Recurrent neural network
KNN	-	K-nearest neighbor
RF	-	Random Forest
PCA	-	Principal component analysis
GAN's	-	Generative adversarial network

CHAPTER 1

INTRODUCTION

1.1 Machine Learning

Machine learning is a fast developing discipline that has transformed several sectors and companies. It includes many different methods and strategies, such as deep learning, reinforcement learning, unsupervised learning, and supervised learning. These techniques provide computers the ability to process and analyse data, spot patterns, and come to reliable conclusions or predictions. Algorithms that use ML may also continually pick up new skills and enhance their performance over time [1]. Models may modify and improve their predictions or behaviours by incorporating user input and fresh data. Systems may improve their accuracy and efficiency through this iterative learning process, which also improves the performance and efficacy of the system as a whole.

IDS in the context of IoT leverage the power of ML to protect IoT networks and devices from security threats. ML algorithms play a crucial role in analyzing network traffic, identifying patterns, and detecting anomalies that may indicate unauthorized access or malicious activities.

The application of ML in IDS for IoT brings several advantages. One of the key benefits is the ability to process and analyze massive amounts of IoT data in real-time. IoT networks generate a vast volume of data from interconnected devices, making it challenging for traditional manual analysis methods. ML algorithms can efficiently handle this data influx, extracting meaningful insights and detecting potential intrusions with high accuracy.

ML models used in IDS for IoT can adapt and learn from evolving threats and changing network conditions. By continuously analyzing network traffic and monitoring device behavior, these models can update their knowledge and improve their detection capabilities over time [2]. This adaptability ensures that IDS systems remain effective in countering emerging and sophisticated threats in dynamic IoT environments.

The versatility of ML also allows IDS to handle the diverse characteristics and

complexities of IoT networks. Different types of IoT devices, each with unique data patterns and communication protocols, can be effectively monitored and protected using ML techniques. These models can be trained on labeled datasets that include various IoT attack scenarios, enabling them to recognize and respond to new and unseen threats.

1.2 Categories of Machine Learning

Machine learning being a vast field is further subclassified into subcategories. It basically has four subcategories. These four subcategories are: Supervised learning, Unsupervised Learning, Semi-Supervised Learning, Reinforcement Learning.

1.2.1 Unsupervised Learning

Unsupervised learning is an essential technique for intrusion detection systems that do not rely on labeled data. Instead, unsupervised learning algorithms analyze the inherent structure and characteristics of the network traffic to detect anomalies and potential intrusions. By identifying patterns that deviate from the expected normal behavior, unsupervised learning algorithms can flag suspicious activities or anomalies that may indicate an intrusion [1]. These algorithms utilize clustering, outlier detection, and statistical analysis techniques to identify patterns that are significantly different from the norm. Unsupervised learning is particularly valuable in detecting unknown attacks or zero-day.

1.2.2 Supervised Learning:

It Supervised learning is a powerful approach in intrusion detection systems that relies on labeled data to train models. In the context of network security, supervised learning algorithms can be trained using historical data that has been meticulously labeled as normal or malicious. These algorithms analyze the input features extracted from network traffic, such as packet headers, payload content, and behavior patterns, along with their corresponding labels. By learning from this labeled data, the supervised learning model can identify patterns and correlations that differentiate normal network behavior from potential intrusions [3]. This enables the model to accurately classify and predict whether incoming network traffic is benign or malicious based on the learned patterns.

exploits, as it does not rely on predefined labels and can adapt to new types of threats.

Table 1.1: Supervised ML models described w.r.t IDS

Method	Working principle	Advantages	Drawbacks	Potential Application
Decision Tree	It follows the principle of the Sum of the product. The attributes having lower entropy are selected and using this splitting is done.	Data preparation for pre-processing requires significant effort. Normalization and scaling of data are not required.	The decision tree is highly unstable for any changes in the provided data. Model prediction and training time is usually very high.	Beneficial to detect any red alert traffic (suspicious) and intrusion within a device.
Support Vector Machine	SVM produces a hyper plain that is a line when we talk about 2D and changes with dimensions and this line separates the data points.	In a dataset, if data attributes are lesser than dimension then SVM will work more efficiently. It works more efficiently in higher-space dimensions.	The efficiency of SVM is comparatively low when the dataset is larger. Its performance decreases in noisy dataset	In smart grids attacks can be predicted with better accuracy. Used in the detection of malware & intrusion.
Naïve Bayes	The underlying principle is based on conditional probability based on the Bayesian theorem.	It converses very fast as there is no iterations involved. It is only based on calculating probabilities.	Accuracy is relatively very less in the cases of zero probability problems or failed conditional assumptions	Beneficial to detect network intrusion.
K-Nearest Neighbour	KNN's underlying principle is based on the fact that if two things are similar, they will exist together.	KNN learns during the time of prediction hence there is no training period. New data can be added seamlessly.	Accuracy reduces while working with large datasets and higher dimensions. Scaling of features is needed.	Beneficial to detect U2R-R2L attacks. Beneficial to detect any red alert traffic (suspicious) and intrusion
Random Forest	In random forests, multiple decision trees is made and then they are combined together for better accuracy.	Random forest is easier to use. It is highly versatile with better efficiency and accuracy.	It has a slow convergence speed due to the use of numbers of decision trees.	Beneficial to detect unauthorized device nodes. It can also detect DDoS.

Table 1.2: Unsupervised ML models described w.r.t IDS

Method	Working Principle	Advantages	Drawbacks	Potential Applications
k-Means Clustering	In the K-mean clustering, task is to add new k-points into the data and adjust those points in k clusters we have.	It is flexible to new datasets or more features when added to an existing one. Convergence is guaranteed. For large datasets also it is scalable	We have to decide the optimal value of k manually. It also depends on initial values.	Beneficial to detect U2R and R2L attacks and added to it detection of intrusion and anomaly.
Principal Component Analysis	PCA is a technique that is used to reduce the dimensions of a dataset that still holds the utmost of the information of the previous dataset.	It reduces the complexity of the given dataset at a very high rate. It reduces the computation associated with the analysis.	A variable that is independent becomes less explainable. Data Standardization is compulsory.	Real-time intrusion detection is made easier due to reduced dimensions.

1.2.3 Semi-supervised Learning

Semi-supervised learning combines the benefits of both supervised and unsupervised learning in intrusion detection systems. This approach utilizes a small portion of labeled data, typically representing known malicious or normal network traffic, along with a larger portion of unlabeled data. The labeled data helps the model understand the basic concepts of normal and malicious behavior, while the unlabeled data provides a more comprehensive representation of the network traffic. By leveraging this combined dataset, semi-supervised learning algorithms can effectively identify anomalies and potential intrusions by comparing the unlabeled data against the learned patterns from the labeled data [4]. This approach is particularly useful in scenarios where obtaining labeled data is expensive or time-consuming, as it reduces the reliance on large-scale labeling efforts while still maintaining reasonable accuracy.

1.2.4 Reinforcement Learning

A dynamic approach to intrusion detection systems called reinforcement learning teaches an agent to decide what to do and how to do it depending on feedback from its surroundings. The agent interacts with the network environment in the context of network security, making choices such as disabling suspect connections, modifying security settings, or implementing countermeasures. Based on the results of its activities, the agent receives feedback in the form of incentives or penalties. By maximising the cumulative rewards over time, the reinforcement learning agent discovers the best methods for identifying and minimising intrusions through trial and error [5]. Through an adaptive learning process, the intrusion detection system may continually strengthen its defences, respond to new attack methods, and dynamically modify its tactics to maintain network security.

1.3 Deep Learning

Deep learning is a cutting-edge approach in the field of intrusion detection in the IoT that utilizes artificial neural networks with multiple hidden layers to extract complex and hierarchical representations from data. In the context of IoT security, deep learning models are trained using a large amount of labeled data to effectively identify and classify various types of intrusions and attacks. These models can analyze diverse and high-dimensional features extracted from IoT network traffic, sensor data, or device behaviors to uncover hidden patterns and anomalies associated with malicious activities. By leveraging the deep hierarchical structure of neural networks, deep learning-based intrusion detection systems can automatically learn and adapt to different types of attacks, including both known and unknown threats [6]. This enables them to provide robust and accurate detection capabilities, improving the overall security and resilience of IoT environments. Furthermore, the ability of deep learning models to perform feature extraction and representation learning on their own alleviates the burden of manual feature engineering.

DL algorithms can effectively handle and analyse enormous volumes of data in real-time because to the availability of strong hardware and processing resources, enabling quick identification and reaction to possible security issues. Intrusion detection systems in the IoT can improve overall security posture and offer a more proactive approach to securing IoT infrastructures and protecting

sensitive data by utilising the benefits of deep learning.

In table 1.3 we have briefly described major deep learning models and their working principles, advantages, disadvantages and potential applications in IDS in IoT.

Table 1.3: Deep learning models described w.r.t IDS

Deep learning Method	Working Principle	Advantages	Disadvantages	Application
Recurrent-Neural Networks (RNNs) [7]	They are the category of neural networks that can model sequential data by using feedback connections to maintain state information over time	Can capture temporal dependencies and sequential patterns, can handle variable-length sequences	Can suffer from vanishing/exploding gradients, may require extensive tuning and optimization	Detecting complex attack patterns that span over multiple network packets and sessions
Convolutional-Neural-Networks (CNNs) [7]	They are the category of neural-networks that use convolutional layers to extract spatial features from input data	Can detect patterns and features in image-based network traffic data, can be computationally efficient	May not be suitable for detecting attacks that do not exhibit clear spatial patterns or features	Detecting network-based attacks that involve packet payloads or headers
Generative-Adversarial-Networks (GANs) [8]	They are the category of propagative model that uses a two-player game between a generator and discriminator to produce unreal data.	Can generate synthetic network traffic data for training and testing IDS, can improve detection performance with limited labelled data	May suffer from mode collapse or instability during training, can be computationally expensive	Data augmentation and synthetic data generation for IDS
Autoencoders [9]	They learn a compressed representation of input data by encoding and decoding it through a bottleneck layer	Can learn a compressed representation of network traffic data, can detect anomalies and deviations	May not be suitable for detecting unknown or novel attacks, may require extensive feature engineering.	Anomaly detection and intrusion detection based on deviations
Long Short-Term Memory (LSTM) [10]	They uses gated memory cells to associate long-term. sequential data dependencies.	Used to associate long-term sequential data dependencies, can handle variable-length.	Can be vulnerable to adversarial attacks, can be computationally expensive.	Detecting complex attack patterns that span over networks.

1.3.1 Long Short-Term Memory

LSTM is a type of recurrent neural-network that has shown remarkable effectiveness in the field of IDS for IoT environments. LSTMs are specifically designed to capture and analyze sequences of data, making them well-suited for detecting and analyzing patterns in network traffic and identifying malicious activities [11]. The main advantage of LSTMs lies in their ability to model longterm dependencies and preserve important contextual information over time. This is particularly beneficial for intrusion detection, as it allows the LSTM to consider the temporal nature of network traffic and capture subtle, time-dependent patterns that may indicate malicious behavior.

In the context of intrusion detection systems, LSTMs can effectively process time series data from various IoT devices and network logs, enabling accurate detection of different types of intrusions and anomalies. By learning from historical data, LSTMs can detect deviations from normal behavior and identify network activities that are indicative of potential attacks. The sequential nature of LSTMs allows them to capture the dynamics of network traffic, such as the order and timing of network events, which can be critical for accurate intrusion detection [12]. Additionally, LSTMs can handle variablelength sequences, making them adaptable to different network environments and accommodating the varying lengths of network sessions and communication patterns.

By leveraging the capabilities of LSTMs, intrusion detection systems in IoT can improve the accuracy and efficiency of threat detection, enabling real-time monitoring and response to potential security incidents. LSTMs have the potential to enhance the overall security posture of IoT environments by effectively analyzing network traffic, identifying abnormal behaviors, and mitigating potential risks. With their ability to handle temporal data and capture intricate patterns, LSTMs provide a promising approach to enhancing the effectiveness of intrusion detection systems in IoT and ensuring the integrity and security of IoT networks. In our purposed model we have used a variant of LSTM that are called as Bi-LSTM [13]. They are more effective and have a upper hand when used with recent dataset and vast amount of data.

1.4 Introduction to Intrusion Detection System

An intrusion-detection system is a software-application that is used to monitor traffic related to a network for unwanted suspicious activity and then issues alerts after such activities are tracked. It also tracks for intrusions such as any policy breach or traffic manipulation. A SIEM(Security information and event management) system is used for collecting the inputs from the system or it is also conveyed to the system administrator. The transmitted information from various sources is filtered using various malicious differentiation practices to filter the false alarms. The typical process involves examining the data and packets traversing a specific network in order to identify any indications of patterns or unusual actions. It also utilizes predetermined rules and patterns to compare against network activities, aiming to identify any attacks or unauthorized access. When it detects network behavior that aligns with these predetermined rules or patterns, an alert is sent to the SIEM system.

1.4.1 Classification of Intrusion Detection System

The classification of IDS involves categorizing them based on various criteria. One common classification is based on the detection approach used by the IDS. This approach divides IDS into two main categories: signature and anomaly-based detection. Signature based IDS rely on predefined patterns or signature-of-known attacks to identify malevolent activities. They compare traffic network against a database of signatures & raise an alert if it founds a match. Comparatively, the anomaly based intrusion framework analyze network behavior and establish a baseline of normal activities. They then monitor for deviations from this baseline, triggering an alert when unusual or suspicious behavior is detected. Another classification criterion is deployment mode, which can include network based IDS that monitor traffic network, host-based IDS that operate on discrete systems, or hybrid IDS that combine both approaches [14]. These categories are discussed below:

- Intrusion detection system based on the host: It resides inside all the devices or computers of the business and has full access to the inside network and internet of

the industry. They keep a track of both incoming and outgoing data from an industry device i.e. host and will send an alert if any maliciousness is detected within the device or network. The basic idea behind its working is that it maintains a record of the previous snapshots of the system and compares it with the newer one. An alert is sent if any system analytical files were edited or deleted. These are more reliable than the Network IDS and even catch threats missed by them.

- Intrusion detection system based on the network: It is used to monitor incoming and outgoing traffic both ways from all devices that are present inside the network. Due to this decisive reason, they are placed at critical points inside that network.
- Intrusion detection system based on the signature: It tracks each and every piece of information of the packet that is passing through the network and further compares it with a large database of known attack signatures and sends alerts according to it. The best example similar to this is antivirus software for our computer systems.
- Intrusion detection system based on the host anomaly: It tracks the network traffic and then evaluates it on the basis of a conventional standard. It technologically uses machine and deep learning models to establish standards. By using these IDS it becomes very easy to detect novel threats.

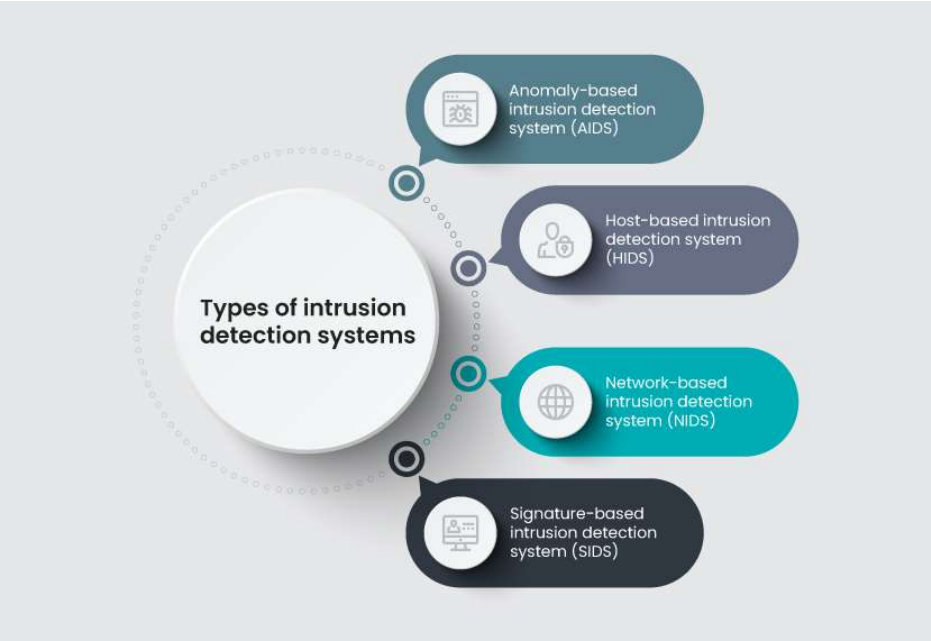

Figure 1.1: Types of IDS

1.4.2 Deployment of Intrusion Detection System

It In a network, various devices that act as a workstation are connected together to form a local area network. This LAN is connected to a switch and then it is further connected to a router. Routers are then connected to a firewall and that firewall opens up the gate for world-wide internet. The firewall also acts as a security layer that defends our workstations in LAN. In addition to this setup at the switch i.e., part of the layer 2 data-link layer Network intrusion detection is set up. This plays a vital role to monitor every frame that is passing through the switch to our LAN. So, in general we can say that Intrusion Detection System is placed at layer 2 of our OSI model.

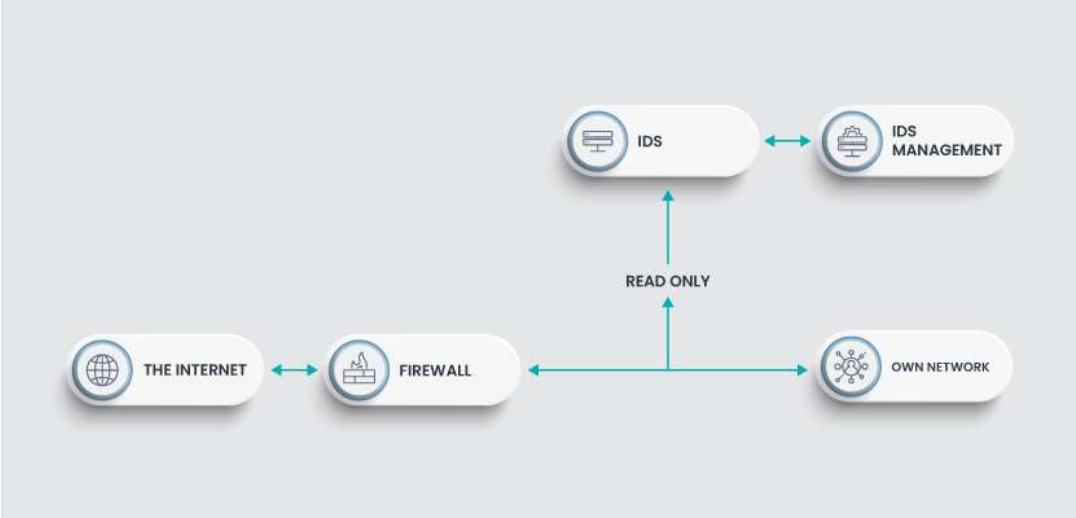

Figure 1.2: Deployment of IDS

1.4.3 Security threats on IoT devices

IoT devices create a surrounding where the internet generally has a connection with the physical world. The working environment of IoT devices is diverse and further, it requires meeting diverse goals. Due to such diversity associated with IoT devices, they are highly prone to physical and cyber-attacks. The structure of these devices is really simple and small without having any complexity. Adding to them they have low power and resources. Therefore, large physical infrastructure for the support of security is not possible. As a result of which IoT device security has become a challenging task. When we consider the main objective of IoT devices which allows devices to be accessed from

anywhere, anytime, and by-anyone we also open doors for the attackers to make devices more accessible for the attacks. An attack is a non-permissible threat that aims to destroy weaknesses related to the security of a system and further leaves an adverse impact. The below figure shows a broad classification of both active and passive attacks with types and examples associated with each.

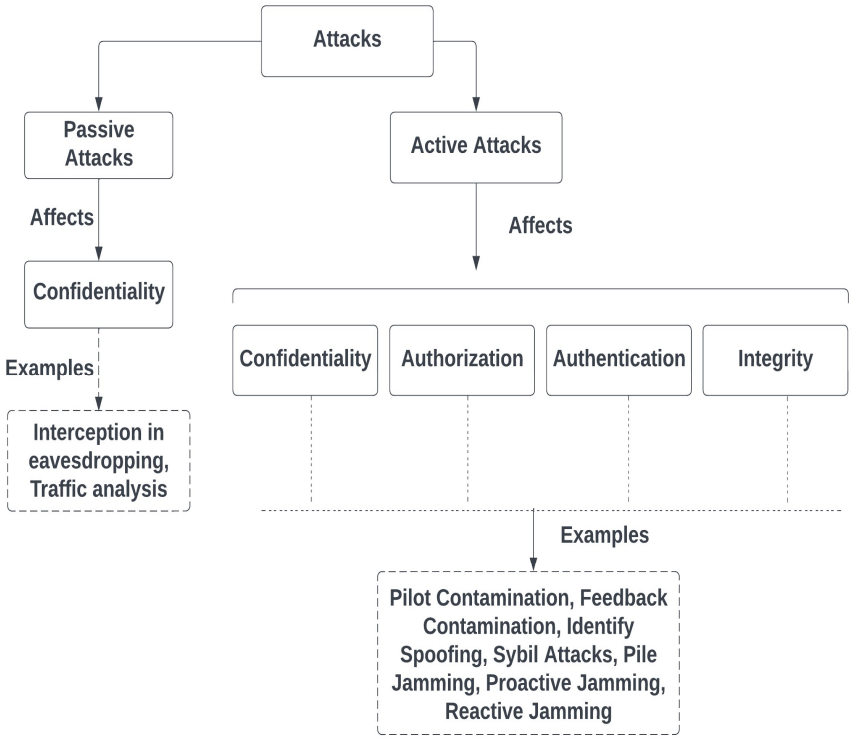

Figure 1.3: Classification of IoT attacks

IoT attacks are mostly linked with three major OSI layers that are application layer, the Network layer, and the physical layer. As discussed earlier our IDS is placed at the data-link layer to monitor the frames associated and detect any malicious activity at any of these layers [9]. Moss major attacks are mentioned below in table 1.4 related to all three Application, network, and physical layers. For multiclass classification it is important to study and analyze most of these attacks so that accuracy of the multiclass model can be increased.

Table 1.4: Attacks associated with different OSI layers

Major Attacks Associated with each layer		
Application layer	**Network Layer**	**Physical Layer**
Phishing attack	DoS	Node Capture
Malicious viruses	Spoofing	Malicious Code Injection
Social Engineering	Sinkhole	False data injection
Malicious Scripts	Wormhole	Replay
	Man-in-the-middle	Side channel
	Routing Information	Eavesdropping
	Sybil	Sleep Deprivation

CHAPTER 2

LITERATURE REVIEW

In table 2.1 we have briefly discussed about most recent studies. In this part of the study, we will be elaborating the methods used in every study, datasets, accuracies and their drawbacks.

Khan M [15] purposed a multiclass framework HCRNN-IDS in 2021 based on the CNN which can be used to detect various malicious-attacks in network. They used CSE-CICIDS2018 recent dataset which resulted an accuracy of 97.15%.

Qazi E [16] purposed a deep-learning model HDLN-IDS in 2023 based on deep-neural networks. They used most recent CSE-CICIDS2018 dataset which resulted an accuracy of 98.90%.

Soe Y [4] research employed several traditional machine learning algorithms to develop an IDS capable of detecting botnet attacks, with potential applicability to other types of attacks as well. Additionally, a hybrid model was proposed that combines both sequential and parallel approaches. They used CSE-CICIDS2017 dataset and results an accuracy of 80.07% with naïve bayes, 99.05% with j48, 99.00% with ANN and purposed hybrid model with 99.09%.

Karatas G [5] study proposed six distinct IDS models based on machine learning techniques on CSE-CICIDS2018 dataset. Moreover, to address the issue of imbalanced dataset, a synthetic-minority oversampling technique was employed. Additionally, accuracies were computed for both sampled and unsampled datasets across six different attack types (Benign, Bot, DoS, BruteForce, Infilt, Sql Inj). ADA provide the highest accuracy of 99.32%.

Kim J [17] study employed a strong spark MLib classifier to detect anomalies and a cutting-edge Conv-AE deep learning model to identify misuse attacks. They used CSE-CICIDS2018 dataset and purposed spark ML + ConvAE resulted accuracy of 98.20%.

Almomani O [18] study had three main objectives. The initial goal was to decrease the number of selected features for the IDS. Following that, bioinspired meta-heuristic

algorithms were applied. Lastly, several traditional machine learning (ML) models were utilized to evaluate the effectiveness of the approach. They used UNSW-NB15 dataset which resulted 92.79% accuracy for SVM and 92.80% for the random forest.

Hosseini S [19] research, the feature selection phase involved employing the MGA-SVM technique. Subsequently, an ANN model was utilized to detect attacks. Additionally, the training of the classifier was enhanced by incorporating PSO and HGS methods. They used NSL-KDD dataset and purposed model MGA-SVM-HGS-PSO-ANN resulted 99.30% accuracy.

Choi E [20] employed an DL approach to implement a CNN model. They used CSE-CICIDS2018 dataset with 96.77%.

Wei P [21] proposed a novel algorithm for optimizing the structure of DBNs. The algorithm utilized a combination of particle swarm and fish swarm optimizers. As a result, the detection time was decreased by a significant percentage of up to 24.69% and accuracy on kDD-Cup99 Test+ was 99.8%.

Tang M [22] research, particular attention is given to addressing the issue of class imbalance. The study proposes a model that combines CLSTM & auto-encoding techniques to effectively capture and learn highly relevant features. NSLKDD dataset resulted accuracy of 92.62% and UNSW-NB2015 dataset resulted 93.03%.

Cui J[3] study proposed a unique intrusion detection system called the GMM-WGANIDS multi-integrated module. This system comprises three distinct phases: feature extraction, imbalance processing, and classification. On NSL-KDD dataset accuracy was 84.65% and on UNSW-NB15 84.87%.

Qazi E [23] study main focus was to propose a non-symmetric deep autoencoder approach for network IDS. This model utilized stacked non-linear denoising autoencoders (NDAEs) in conjunction with support vector machines (SVM). The implementation of this approach was carried out using the TensorFlow library. KDDCup99 dataset resulted 99.65% on purposed model.

Singh G [24] employed an OnlineSequential-Extreme-Learning-Machine (OS-ELM) model for the purpose of malware detection. The study specifically emphasized

important factors such as feature selection, handling large datasets, and feature extraction. On universal dataset NSL-KDD Binary model resulted an accuracy of 98.66% and multiclass resulted in 97.67%.

Kabir [25] proposed the use of a least-squared SVM for intrusion detection. The least-square SVM was applied to the extracted samples to identify instances of intrusion. Unlike traditional SVMs that solve a quadratic-programming problem, the LS-SVM in this study solves two linear equations and accuracy of 99.64%.

Table 2.1: Literature survey of recent work

Ref	Description	Dataset	Model	Accuracy in %	Drawback
[15]	In this study, a intrusion-detection based deep-learning framework is purposed which is based on convolutional-recurrent neural network which can be used to classify and predict various malicious-attacks in the network.	CSE-CICIDS2018	HCRNNIDS	97.15%	Only tested on one dataset. Anomaly detection absent.
[16]	In this study, a convolutional-recurrent neural network is purposed to create a hybrid-intrusion-detection system. In it deeplayes of RNN were used to extract the features in purposed HDLNIDS.	CSE-CICIDS2018	HDLNIDS	98.90%	Zero-day attack handling capacity absent.
[4]	In this study, various classical machine learning algorithms were used to purpose a IDS which is used to detect botnet-attacks and this can further be extended to other attacks. A hybrid model was also purposed in series and parallel.	CSE-CICIDS2017	NB, J48, ANN, Hybrid	80.07%, 99.05%, 99.00%, 99.09%	It lacks normal traffic patterns on the different natures of IoT.
[5]	In this study, six different machine-learning based models of IDS were purposed. Also, imbalance of the dataset was reduced using synthetic-minority over-sampling technique. Adding to this, various accuracies were calculated using sampled and un-sampled datasets on different six-attacks (Benign, Bot, DoS, BruteForce, Infilt, Sql Inj). Average accuracies of sampled datasets are mentioned in our review.	CSE-CICIDS2018	ADA, DT, RF, KNN, GB, LDA	99.32%, 98.56%, 99.19%, 95.30%, 99.38%, 83.62%	Only tested on classical machine learning algorithms, hence lacks optimization.
[17]	In this study, a robust spark MLib classifier was used for anomaly-detection and a deeplearning state of art Conv-AE	CSE-CICIDS2018	Spark ML + Conv-	98.20%	It lacks testing on real-times

	for misuse attacks, further used for an intelligent and efficient IDS to classify & detect malicious attacks.		AE		streaming.
[18]	In this study, first objective was to reduce number of selected features for IDS. Then, secondly bioinspired meta-heuristic algorithms were imposed. Lastly, few classical ML models were deployed to access its effectiveness.	UNSW-NB15	Combination of PSO, MVO, GWO, MFO, WOA, BAT	J48-92.80%, SVM-92.79%, RF-92.80%	It lacks in overall accuracy and recent Genome Microsoft dataset which might have been used.
[19]	In this study, MGA-SVM technique was used for feature selection phase. Then an ANN model is used for the detection of attacks. Further adding to this, PSO and HGS are used.	NSL-KDD	MGA-SVM-HGS-PSOANN	99.30%	It lacks accuracy for real-time zeroday attacks
[21]	In this study, to optimize DBN's structure network a novel joint optimization-algorithm was purposed. Optimizers used were particle and fish swarm. The detection time was reduced by up to 24.69%.	KDD Cup 99	AFSA-GA-PSO-DBN	Test168.7% Test283.8% Test+99.8%	Its fitness function is not much appropriately used.
[22]	In this study, handling of class imbalance problem is more focused. A model based on CLSTM and auto-encoding is purposed which can learn high level of associated features.	NSL-KDD	ARB, CLAE	90.82%, 92.62%	Improvement in performance needed using graph relations.
		UNSW-NB2015	ARB, CLAE	91.64%, 93.03%	
[3]	In this study, a novel GMM-WGAN-IDS multi-integrated module intrusion detection system was purposed. This framework has 3 phases-feature-extraction, imbalance-processing & classification.	NSL-KDD, UNSW-NB15	GMM-WGAN-IDS	84.65%, 84.87%	Its feature extraction can be made effective.
[23]	In this study, a deep autoencoder that is nonsymmetric was purposed for the network intrusion. He mode uses stacked NDAEs and SVM along with implementation on the TensorFlow lib.	KDD Cup 99	Purposed model	99.65%	Multi attack classification absent.
[26]	In this study, implementation & design of an IDS was implemented using semi-supervised k-means algorithm.	NSL-KDD	K-means	80.19%	Very low results.
[27]	In this study, a heuristic optimization algorithm was purposed that also has time as a factor named chaos particle	NSL-KDD	TVCPSO-MCLP,	97.23%, 97.03%	Model can be applied with

	swarm optimization and it is used for the IDS framework		TVCPSO-SVM		kernel function to improve it.
[24]	In this study, a model based on OnlineSequential-Extreme-Learning-Machine is used for malware detection. It focuses on various key aspects like selecting features, the enormity of the dataset, and feature selection.	NSL-KDD	Binary OS-ELMB Multiclass OS-ELMB	98.66%, 97.67%	Alpha & beta should be tested for newer global datasets.
[28]	In this study, a support vector machine-based IDS framework was purposed with augmented-features. Along with this logarithm-marginal-density ratio transformation was done to option better quality new features.	NSL-KDD	SVM1, SVM2	99.18%, 99.15%	Different attack types can be included.
[29]	In this study, a classic ensemble classifier random forest was purposed. Discretization is used a pre-processing technique and performance of J48 and RF was calculated.	NSL-KDD	Random Forest, J48 Tree	99.67%, 99.28%	Imbalances in older dataset present.
[25]	In this study, a least-square SVM was purposed. To detect intrusion, at extracted samples it was applied. LS-SVM instead of solving a quadratic-programming.	KDD Cup99	LS-SVM	99.64%	Lack Zeroday attacks detection.
[30]	In this study, a voting-machine algo was used which is further made specific to wormhole prediction.	Kyoto 2006+	OPFC, SA-IDS	97.53%, 96.02%	Real-time deployment and testing absent.
[31]	In this study various machine learning models were deployed and tested with highly preproccess data.	NSL-KDD, DARPA	KNN, J48, CANN	99.2%, 99.30%, 99.50%	Lacks results of newer dataset with recent attack types.

CHAPTER 3

METHODOLOGY

In this section of methodology of machine and deep learning for intrusion detection in IoT devices we will majorly focus on implementation part. Initially we discuss about the available global datasets and compare them on the basis of features, records, data source and description. Moving further we discuss about exploratory data analysis and pre-processing of data records and selecting CSE-CICIDS 2018 one of the most recent dataset. Then we will discuss about the architecture of purposed Bi-LSTM model along with major hyper parameters.

3.1 Standard Global Datasets

Global intrusion detection datasets are essential in the field of cybersecurity because they give academics and professionals useful tools for researching and comprehending different kinds of network attacks. These datasets provide information about actual network traffic that has been gathered from a variety of sources, including business networks, educational institutions, and research initiatives. They record a variety of attack situations, such as Distributed Denial of Service (DDoS), port scanning, malware infections, and unauthorised access attempts, among others. These databases are used by researchers to create and assess intrusion detection systems, study attack patterns, and improve network security measures . In this part we have provided a description of 10 major datasets with the help of a short summary. We have provided a brief of each dataset along with its year, total attacks, attack definition, and deficiency associated with each dataset.

- **KDD Cup-99:** This dataset is one of the first intrusion detection datasets which was prepared in 1998 by the MIT labs [32]. It was based on the military environment. It was made public and used in during the 1999 contest due to which it holds its name. It consists of a huge 5 million training records and

around 2 million testing records. Every record is associated with 41 attributes that are labelled as normal or attack. Attacks are further narrowed into 4 types- DoS (Denial of service) for an example- syn-flood, R2L(Remote to local) for an example- password guessing, U2R(User to Root).

Table 3.1: Global Standard Datasets

Dataset	Year	Total Records	Feature	Attack Types	Source	Description	Disadvantage
KDD Cup 99	1999	4,900,000	41	4	DARPA	KDD Cup 99 was the first benchmark dataset used for intrusion detection research, containing a large number of network connection records.	It lacks diversity in attack types and may not accurately represent modern attacks.
Kyoto 2006+	2006	2,085,529	14	8	Kyoto University	Kyoto 2006+ is a dataset that includes both normal and attack traffic in a university network environment.	It only includes traffic data from a single university network. It also suffers from class imbalance.
NSL KDD	2009	1,251,507	42	4	UNB, KDD99	It is a refined version of KDD Cup 99. It is modified to remove redundancy & inconsistencies. It has more diverse attack types, making it a more realistic benchmark dataset.	It lacks diversity in the types of attacks and the source of data. The dataset also suffers from class imbalance problem.
AWID	2009	177,858	63	1	University of Calgary	It is a wireless intrusion detection dataset that contains both normal and abnormal traffic data in a wireless environment. It includes a large number of features such as RSSI and packet inter-arrival time.	It suffers from a high degree of class imbalance, where the number of attack instances is significantly lower than the number of normal instances.
Drebin	2014	123,453	215	1	North eastern University	It is a dataset that contains real-world Android malware samples. A large	It only includes Android malware samples, which

						number of features such as permissions and API calls.	may not be representative of other types of malwares.
UNSW-NB15	2015	2,540,044	49	10	University of South New Wales	It is a recent dataset that includes more diverse attacks than previous datasets. It also includes normal traffic data, making it a more representative for real-world.	It suffers from a high degree of redundancy and irrelevant features, which can negatively impact models.
CICIDS 2017	2017	1,780,657	79	8	Canadian Institute	It is a dataset that includes a significant number of features, including packet header and payload features. It contains both normal traffic and attack data.	It suffers from class imbalance, where the number of attack instances is significantly lower than the number of normal instances.
CSECICIDS 2018	2018	1,780,657	79	15	University of New Brunswick	It is a newer dataset that includes newer attacks such as Web-based attacks and IoT-based attacks including newer attacks such as Ransomware and DDoS attacks	It suffers from a high degree of redundancy and irrelevant features
NAB	2018	58,675	53	13	University of California	It is a dataset that contains a variety of time-series. It includes both normal and anomalous data and a wide range of anomalies such as spikes, dips, and changes in trend.	It includes a limited number of anomalies, which can make it difficult to train machine learning models.
Genome	2019	5,000,000	1,273	4	Microsoft	It contains real-world adversarial web content. It includes malicious and benign URLs and a large number of features.	It is a very huge dataset making it complex to analyse for any preprocessing.

- **Kyoto-2006+:** This dataset was obtained from a range starting from 2006- 2009 by establishing email servers, honey-pots, sensors on the darknet, and web-crawlers

over the range of three years by the kyoto university [33]. It consists of in total of 24 attributes. Among these 24 attributes, 14 major features were endured from KDD Cup-99, and rest 10 features were added by the kyoto university. These added features support a more enhanced platform to evaluate an intrusion detection system.

- **NSL-KDD:** It is the enhanced and improved version of KDDCup-99 dataset. In this dataset, the number of records is kept proportionate with respect to the record percentages in the KDDCup-99 which results in the efficient classification over wide ranges of machine and deep learning methods [32]. It also reduces the need to choose randomly a portion small of the dataset for experiments by making it affordable to evaluate. Further improvements also involve removing redundant datarecords from the training set which eradicates the biased nature of a classifier towards frequent records.
- **UNSW-NB-15:** This dataset was purposed by the Range Cyber Lab based of Australia by the cyber-centre by using the tool IXIA PerfectStrom. Data of around 100GB was captured using a more advanced algorithm tcpdump tool. It has forty-nine different types of attributes which are encountered using 12 advanced algorithms. For each record, it consists of 9 different types of attacks which are briefly described as Worms, Reconnaissance, Shellcode, Port scans, Generic, Fuzzers, Exploits, DoS, and lastly Backdoors.
- **CIC-IDS-2017:** This dataset is one the most recent and advanced which involves the latest intrusion attacks faced by IoT devices. It was developed during the year 2017 by the Institute-Cyber-Security(CIC) based out in Canada [34]. It was assembled by collecting huge data during the intense working hour of 9:00 am to 5:00 pm from Monday to Friday. This is the most common working schedule which makes this dataset more realistic and reliable to research. Parameters that were used during the analysis of CIC-IDS-2017 were the IP addresses of the destination & source and other stamps of time. It consists of 7 different types of attacks associated with each attribute which are BruteForce, Botnet, Heart-Bleed, DoS, DDoS.

3.2 Exploratory Data Analysis of CSE-CICIDS 2018

CSE-CICIDS 2018 CSE-CICIDS 2018 is a comprehensive and widely used dataset in the field of intrusion detection system (IDS) research. It is specifically designed for evaluating and benchmarking the performance of IDS algorithms and techniques. The dataset is derived from real-world network traffic data captured in a controlled environment, simulating various types of attacks and normal network traffic. The CSE-CICIDS 2018 dataset consists of a diverse range of network traffic features, including packet-level and flow-level attributes. It contains a total of 80 features, encompassing both numerical and categorical variables. The dataset covers multiple attack categories such as Denial-of-Service (DoS), Distributed Denial-of-Service (DDoS), Port Scan, Brute Force, and Web Attacks, among others [35]. It also includes benign network traffic to represent normal behavior.

One of the notable features of the CSE-CICIDS 2018 dataset is its large size. It contains millions of instances, making it suitable for training and evaluating machine learning algorithms. The dataset is highly imbalanced, with a significantly larger number of benign instances compared to attack instances. This characteristic poses a challenge for developing effective IDS models, as they need to accurately detect and classify rare attack patterns while avoiding false positives.

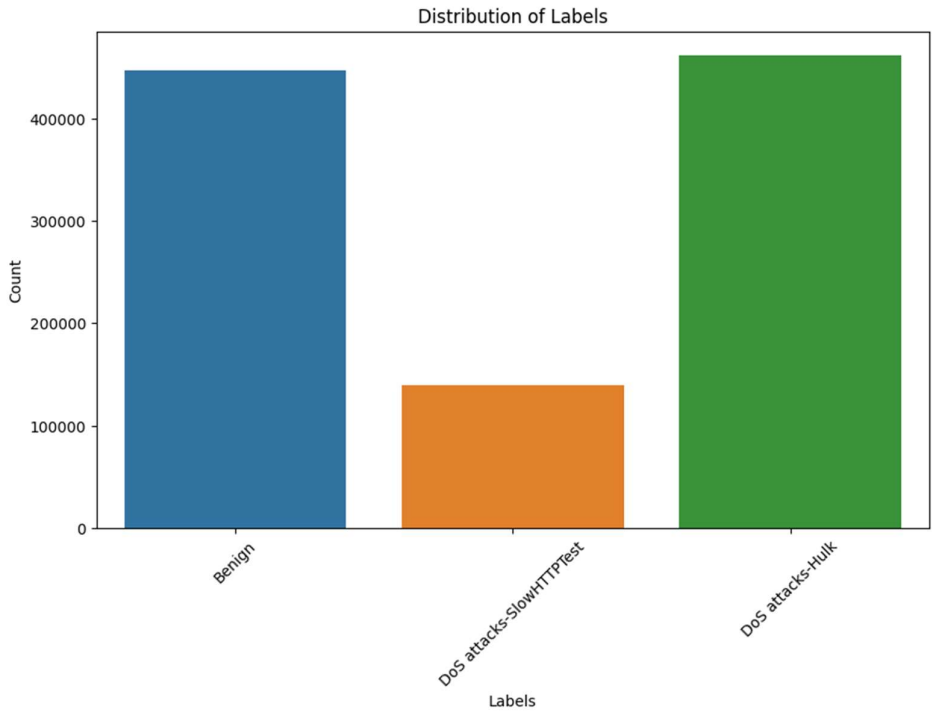

Figure 3.1: Distribution of labels in CSE-CICIDS 2018

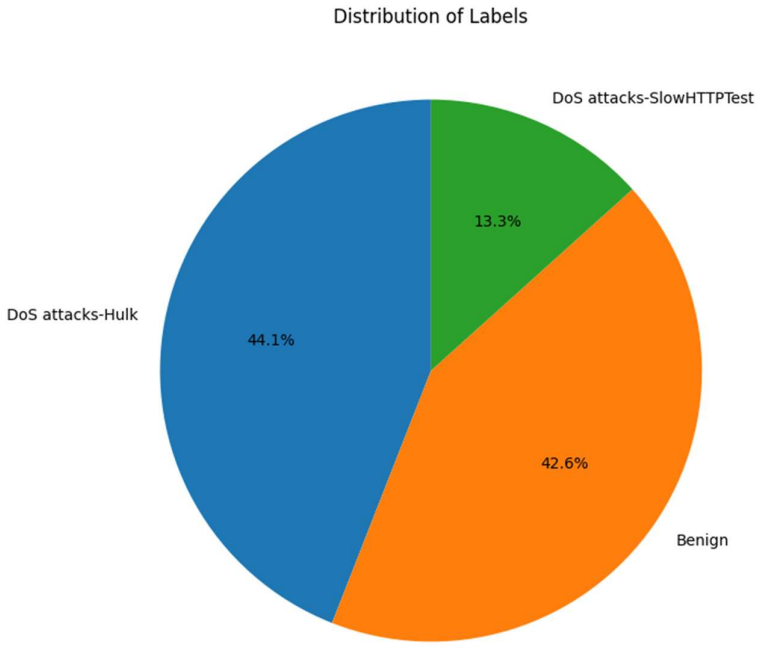

Figure 3.2: Pie chart showing distribution in percentage

Few other important data analyses were performed like filling missing values with 0, dropping infinite and null values.

```
# Drop Infinate and Null

def dropInfinateNull(df):
    print (df.shape)

    # replace infinity value as null value
    df = df.replace(["Infinity", "infinity"], np.inf)
    df = df.replace([np.inf, -np.inf], np.nan)

    # drop all null values
    df.dropna(inplace=True)

    print (df.shape)

    return df
```

Figure 3.3: Function used for dropping infinite and null values

3.3 Transforming labels to Binary

CSE-CICIDS 2018 dataset were transformed in Binary labels for single class classification.

The labels were termed as Benign if they are non-malicious and Not-Benign as malicious. This will modify dataset into single class classification.

```
#Transform Target Label into Binary Class
%%time
# encode the target feature
df['Label'] = df['Label'].apply(lambda x: "Benign" if x == 'Benign' else "Malicious")
print(df['Label'].unique())
```

```
['Benign' 'Malicious']
CPU times: user 672 ms, sys: 551 ms, total: 1.22 s
Wall time: 1.19 s
```

```
df['Label'].value_counts()
```

```
Malicious    601802
Benign       446772
Name: Label, dtype: int64
```

Figure 3.4: Binary label transformation of CSE-CICIDS 2018

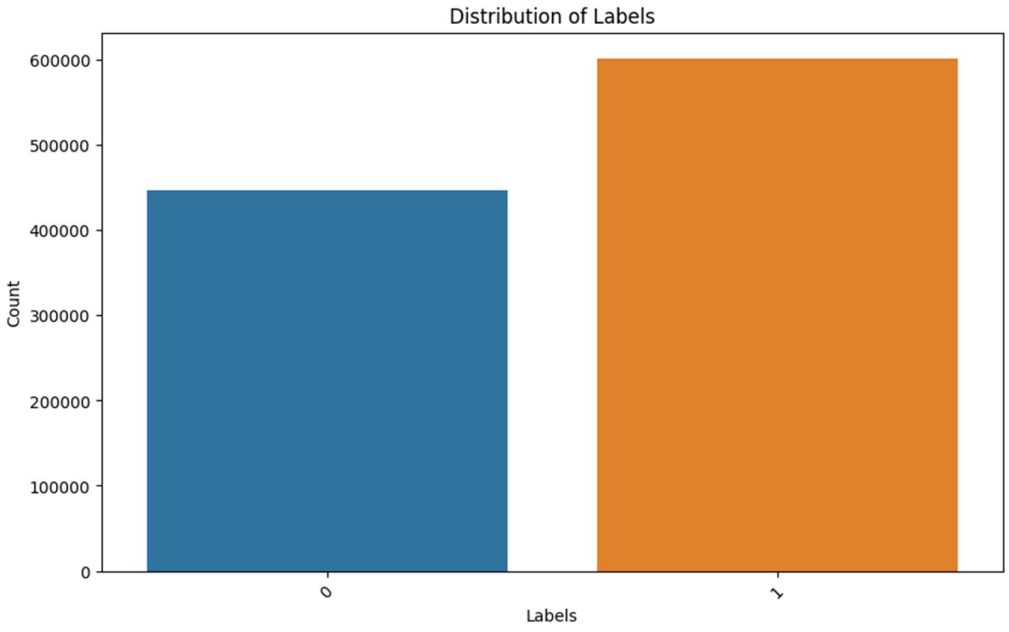

Figure 3.5: Bar Graph of CSE-CICIDS 2018 after Binary labels

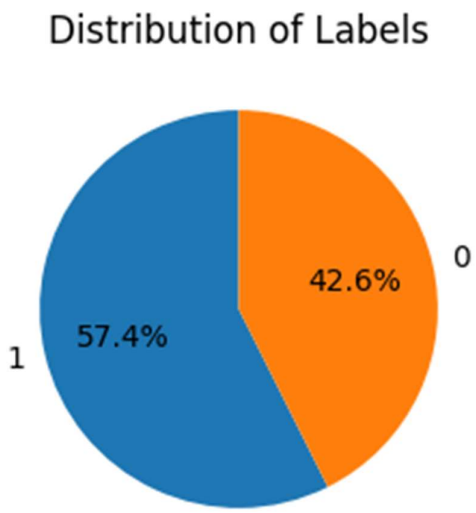

Figure 3.6: Pie chart of CSE-CICIDS 2018 after Binary labels

3.4 Purposed Bi-LSTM model

Bi-LSTM (Bidirectional Long Short-Term Memory) is a variant of the RNN-architecture that has gained significant attention in the field of IDS, specifically for analyzing the CSE-CICIDS 2018 dataset. The CSE-CICIDS 2018 dataset contains a large number of network traffic records, consisting of both benign and malicious activities.

Bi-LSTM offers a powerful solution for capturing longrange dependencies and sequential-patterns in temporal data, making it well-suited for analyzing network traffic and detecting intrusions. Unlike traditional LSTM models, Bi-LSTM processes the input sequence in both forward-backward directions simultaneously, allowing it to capture information from past-future time steps. This bidirectional nature enables the model to effectively understand the context and dependencies within the network traffic data.

In the context of ID, Bi-LSTM can effectively learn complex patterns and behaviors associated with different types of attacks, including Denial of Service (DoS), Distributed DoS (DDoS), probing attacks, and more. By analyzing the temporal dynamics of network traffic, Bi-LSTM can detect anomalous patterns that deviate from normal behavior and accurately classify them as malicious activities.

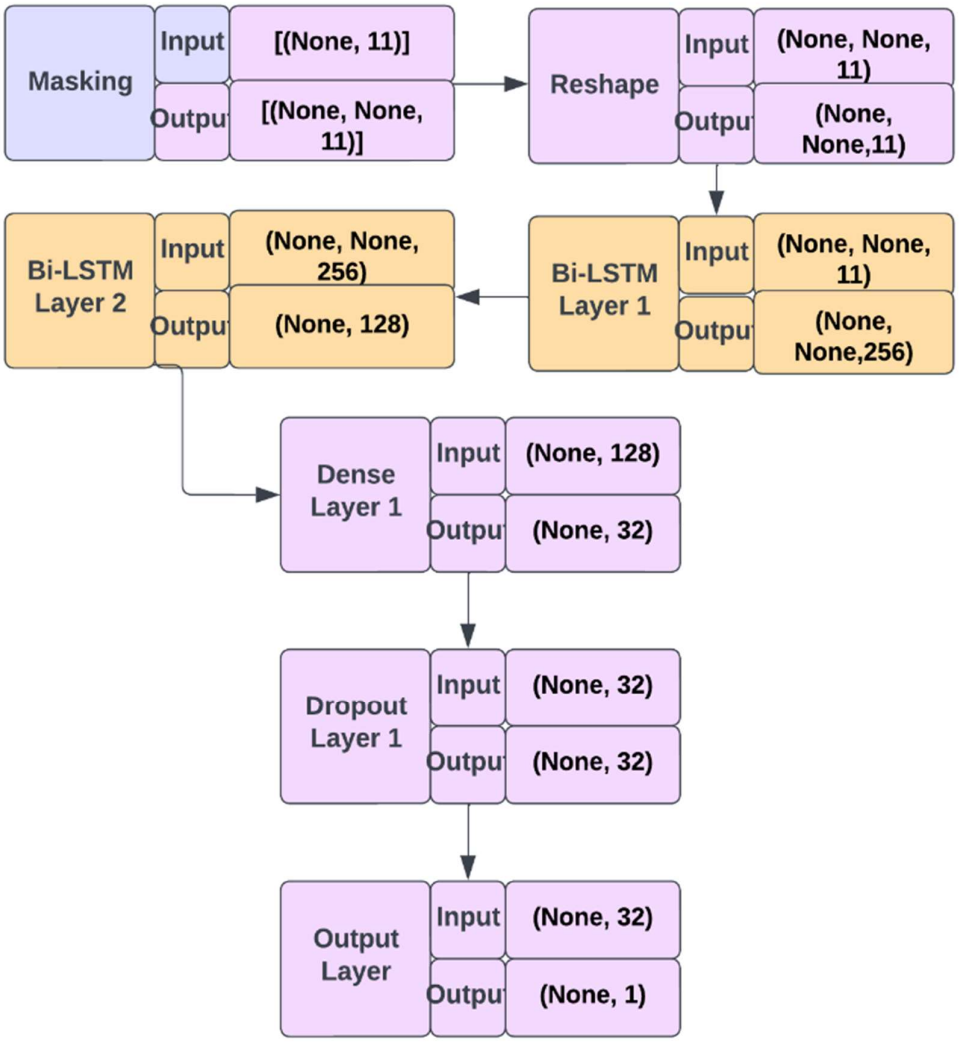

Figure 3.7: Architecture of purposed Bi-LSTM model

The advantages of using Bi-LSTM for ID is its ability to automatically learn relevant features from the raw input data. This eliminates the need for manual feature engineering and allows the model to capture high-level representations of the network traffic. The sequential nature of Bi-LSTM also enables it to handle variable-length input sequences, making it suitable for analyzing network traffic data of different sizes and durations. The proposed model shown in Fig is a Bi-directional-LSTM a type of neural network, which is a variant of the LSTM that can process input sequences in both onward and retrograde directions. The model architecture is defined using the keras- Sequential-API. The model has a Masking layer, Reshape Layer, two Bi-LST layer, dropout layer and two dense layers.

3.5 Layers of purposed Bi-LSTM model

Purposed Bi-LSTM model has a Masking layer, Reshape Layer, two Bi-LST layer, dropout layer and two dense layers. Every layer is defined below with its proper working.

- Masking: This receives the pre-processed input data, which consists of various features like the source-destination IP addresses, ports, and protocols. In intrusion detection our data varies on the basis of duration for which it is being monitored leading to variable length sequences. As, Bi-LSTM can only process 3D tensor we have used masking to handle variable-length sequences. It basically converts the coming 2D tensor into 3D so that it can we further processed by the Bi-LSTM layer.
- Reshape: The reshape layer is used to change the arbitrary shape of the input data into the shape specified in the model based on the model requirements. It does not have any parameters that needs to be trained and it does not change the data, it simply changes the dimensions of the data.
- Bi-LSTM layer 1: This layer consists a subnetwork of two LSTM units with 128 hidden units each. Bi-LSTM layer processes the input in both directions The input sequence in this layer is processed in both forward and backward directions to the input sequence in parallel, which results into the capturing of more information

Table 3.2: Purposed Model layers and Params

Layer	**Parameter Name**	**Value**
Masking	Mask Value	0
Reshape	Target Shape	(-1, 11)
Bi-LSTM Layer 1	Units	128
	Return sequences	True
Bi-LSTM Layer 2	Units	64
Dense-Layer 1	Units	32
	Activation Function	ReLU
Dropout Layer 1	Dropout Value	20%
Output Layer/ Dense-Layer 2	Units	1
	Activation function	Sigmoid

compared to a single LSTM layer. This increased information further results in better training of model. The output of the first Bi--LSTM layer is a set of high-level-features that capture the temporal dependencies within the input-sequence. The return sequence is kept as true keeping input provided to Bi-LSTM layer 2 in 3D.

- Bi-LSTM layer 2: This layer consists of two LSTM units with 64 hidden units each. Like the previous layer, the input sequence in this layer is processed in forward and back both directions and helps the model to find more complex-patterns and allows model to capture higher temporal dependencies in provided input data.
- Dense layer 1: This dense-layers is a classifier or also called a fully connected layer. The output of the second BiLSTM layer is typically a high-dimensional representation of the input sequence that captures the temporal dependencies within the data. This layer has ReLU activation that is then applied to this high-dimensional depiction to produce a lower-dimensional-representation that can be used for classification. This layer applies a linear-transformation to the input features and produces a set of activations, which are then passed through the ReLU activation function. A rectification operation is applied by the ReLU function, that means it sets all -ive values to zero and passes +ive values unchanged. This non-linear activation function helps to introduce non-linearity into the model. The output of this layer after ReLU activation is a lower-dimensional-representation of the input sequence that has been mapped to the output classes.
- Dropout layer: This dropout layer has a drop out value of 20%. They are used to tackle the over-fitting of the model during training. Drop-out is the regularization method that arbitrarily drops out some units in the network during training. Overall, this layers in the model allows it to effectively process and classify the input data, while also preventing overfitting during training.
- Output Layer / Dense layer 2: This dense-layer with the 'sigmoid' activation

function further used for the final classification of provided data into 1 of 2 categories that is “Benign” or “Not-Benign”. The sigmoid function maps the output of the model to a probability distribution, and simply marks a threshold value of 0.5. It predicts as “Benign” if value is above 0.5 and “Not-benign” if value is below 0.5.

Table 3.3: Hyperparameters of purposed model

Hyperparameter Name	**Parameter**
Train test split ratio	80:20
Loss function	Binary Cross entropy
Learning rate	1e-5
Optimizer	ADAM Optimizer Beta values: (0.9, 0.999) Epsilon value: 1e-7 Weight Decay: None
Metrics	Accuracy
Epochs	10
Features Used	'Timestamp', 'Fwd Pkt Len_Std', 'Fwd Pkt Len_Mean', 'Fwd Pkt Len_Max', 'Fwd Seg Size Avg', 'Pkt Len Std', 'Flow IAT Std', 'Bwd Pkt Len_Std', 'Bwd Seg Size Avg', 'Pkt Size_Avg', 'Subflow Fwd Byts'

CHAPTER 4

RESULTS AND DISCUSSION

4.1 Experimental Setup

The purposed model is implemented on google colab on Google Compute Engine backend (GPU). The system was having a system ram of 12.7GB with disk space of 78.2GB. Hardware acceleration of GPU was provided for faster execution and training of model.

4.2 Evaluation Metrics

In deep learning, evaluation metrics are used to determine a model's performance. To determine the performance of our models, we utilize classification metrics like F1 score, recall, precision, and accuracy. We can measure the performance using a confusion matrix. It is a matrix of 2 * 2 table, for binary classification.

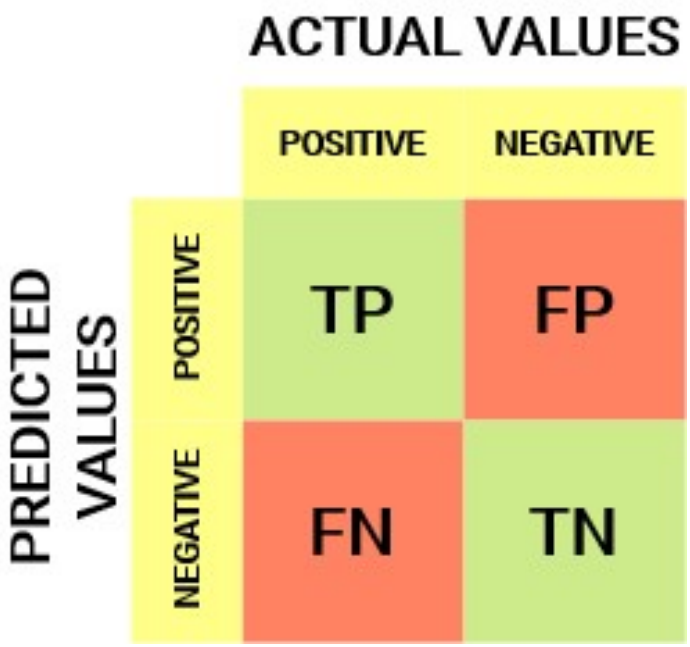

Figure 4.1: Binary confusion Matrix

- True Positive (TP): Our model predicted class ‘malicious’ and the actual class is ‘malicious’.
- True Negative (TN): Our model predicted class ‘non-malicious’ and the actual class is ‘non-malicious’.
- False Positive (FP): Our model predicted class ‘malicious’ but the actual class is ‘non-malicious’.
- False Negative (FN): Our model predicted class ‘non-malicious’, but the actual class is ‘malicious’.

4.3 Result Analysis

The proposed Bi-LSTM based IDS using dataset CSECIC-IDS-2018 have achieved an accuracy of 99.554%, precision of 0.992, recall of 1.0, and F1-score of 0.9961 as shown in Fig. These results demonstrate the effectiveness of Bi-LSTMs in detecting network intrusions when compared with other relevant models in table 4.1.

	Accuracy	Precision	Recall	F1 Score
Bi-LSTM	0.995542	0.992278	1.0	0.996124

Figure 4.2: Classification report

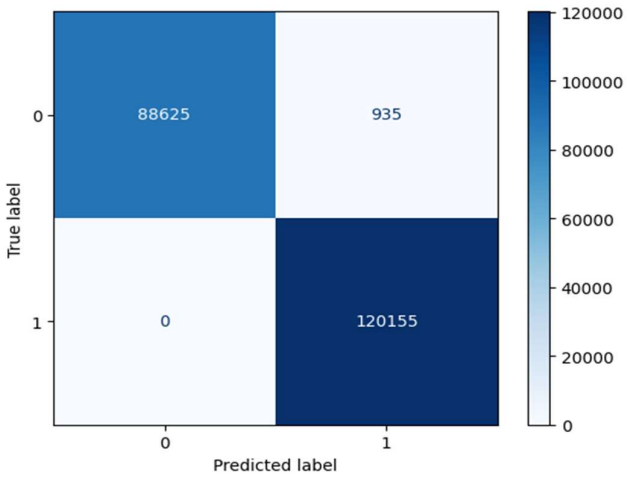

Figure 4.3: Confusion Matrix

Table 4.1: Comparing model accuracy with recent works

References	Related Model/s	Dataset	Accuracy
[36]	LSTM, LSTM-PCS, LSTM-MI	CSECICIDS-2018	98.88% 99.29% 96.24%
[37]	DL-CNN-LSTM	NSL-KDD UNSWGNB-15	98.5% 98.9%
[38]	H-C-RNN	CSECICIDS-2018	97.75%
[39]	I-Siam IDS	NSL-KDD CSECICIDS-2018	95.00% 93.10%
[40]	Decision-Tree Random-Forest KNN ADA	CSECICIDS-2018	98.56% 99.19% 95.30% 99.20%
[41]	PCA-Naïve Bayes	NSL-KDD	85.5%
[42]	G-IDS(GAN)	NSL-KDD99	96.88%
[43]	LMDRT-SVM LMDRT-SVM2 Single-SVM	KDD CUP 99	99.13% 99.28% 97.35%
Proposed	**Proposed Bi-LSTM**	**CSECICIDS-2018**	**99.554%**

Overall, the model trained on the CSE-CICIDS 2018 dataset was able to achieve high accuracy in detecting whether a packet on receiving server is “benign” (Not-Malicious) or “Not benign” (Malicious), indicating that the model has the potential to be used for realworld applications in cyber security to fulfil the ongoing demand of intrusion detection system methods.

CHAPTER 5

CONCLUSION AND FUTURE SCOPE

Clickbait Intrusion detection System is a critical aspect of ensuring the security of computer networks and IoT devices. In this study, we have compared more than 20 recent literatures based on deep & machine learning models, bio-inspired meta-heuristic models, hybrid and optimization models w.r.t IDS based on their architecture, datasets used and accuracies. Adding to this drawback associated with every literature review model is also mentioned. We have also compared 10 global datasets based on their launched year, total records, features, attack types, sources and disadvantages. Moving forward we have used most recent CSECIC-IDS2018 data set which have 15 different number of features and 36 recent different attack type. We have also proposed a model based deep learning technique that is Bi-LSTM having 7 different layers that can be used to make prediction of "Benign" (Not-malicious) and "Not benign" (malicious) and have achieved an accuracy of 99.554%, precision of 0.992, recall of 1 and F1 score of 0.996. This accuracy suggest that Bi-LSTM can be used to make a very efficient Intrusion detection system and have real world applications. Although significant progress has been made in the field of intrusion detection, there are several areas that offer opportunities for further research and improvement. Future research could focus on implementing a multiclass model in which we can predict different types of malicious attacks such as DoS-attack, Botnet-attack, Web and Infiltration attack and others associated attacks with CSECIC-IDS2018 dataset. Adding to this in future, we can also focus on predicting severity of associated attack on a fixed scale. Severity of attack can be decided on the basis of features or on the basis of associated attacks giving weights to each thus making it more suitable for realtime applications. Further research could focus that IDS should be designed to adapt and evolve with the changing landscape of cyber threats using more newer datasets like Genome-2020 with 5,000,000+ records and 1,273 features designed by Microsoft. Handling unknown/Zero-

day attacks should also be a point of focus while developing anomaly detection techniques and unsupervised learning approaches to effectively identify unknown attacks. Adding to above future research can explore real-time anomaly detection using techniques such as stream processing, online learning, and adaptive models to enable real-time anomaly detection and response. This would reduce response time and enhance the system's ability to detect and mitigate attacks promptly. Lastly studies in IDS on IoT also lacks a real-world implementation. Without real-world deployment or testing, it is challenging to determine the performance and practical applicability of the techniques in a production and commercial environment.

REFERENCES

[1] S. V. N. Santhosh Kumar, M. Selvi, and A. Kannan, "A Comprehensive Survey on Machine Learning-Based Intrusion Detection Systems for Secure Communication in Internet of Things," *Comput Intell Neurosci*, vol. 2023, pp. 1–24, Jan. 2023, doi: 10.1155/2023/8981988.

[2] G. Singh and N. Khare, "A survey of intrusion detection from the perspective of intrusion datasets and machine learning techniques," *International Journal of Computers and Applications*, vol. 44, no. 7, pp. 659–669, 2022, doi: 10.1080/1206212X.2021.1885150.

[3] J. Cui, L. Zong, J. Xie, and M. Tang, "A novel multi-module integrated intrusion detection system for high-dimensional imbalanced data," *Applied Intelligence*, vol. 53, no. 1, pp. 272–288, Jan. 2023, doi: 10.1007/s10489-022-03361-2.

[4] Y. N. Soe, Y. Feng, P. I. Santosa, R. Hartanto, and K. Sakurai, "Machine learning-based IoT-botnet attack detection with sequential architecture," *Sensors (Switzerland)*, vol. 20, no. 16, pp. 1–15, Aug. 2020, doi: 10.3390/s20164372.

[5] G. Karatas, O. Demir, and O. K. Sahingoz, "Increasing the Performance of Machine Learning-Based IDSs on an Imbalanced and Up-to-Date Dataset," *IEEE Access*, vol. 8, pp. 32150–32162, 2020, doi: 10.1109/ACCESS.2020.2973219.

[6] B. I. Farhan and A. D. Jasim, "Survey of Intrusion Detection Using Deep Learning in the Internet of Things," *Iraqi Journal for Computer Science and Mathematics*, vol. 3, no. 1, pp. 83–93, 2022, doi: 10.52866/ijcsm.2022.01.01.009.

[7] S. Tsimenidis, T. Lagkas, and K. Rantos, "Deep Learning in IoT Intrusion Detection," *Journal of Network and Systems Management*, vol. 30, no. 1, Jan. 2022, doi: 10.1007/s10922-021-09621-9.

[8] M. H. Shahriar, N. I. Haque, M. Rahman, and M. Alonso Jr, *G-IDS: Generative Adversarial Networks Assisted Intrusion Detection System*. 2020.

[9] H. Hindy, R. Atkinson, C. Tachtatzis, J. N. Colin, E. Bayne, and X. Bellekens, "Utilising deep learning techniques for effective zero-day attack detection," *Electronics (Switzerland)*, vol. 9, no. 10, pp. 1–16, Oct. 2020, doi: 10.3390/electronics9101684.

[10] A. Boukhalfa, A. Abdellaoui, N. Hmina, and H. Chaoui, "LSTM deep learning method for network intrusion detection system," *International Journal of Electrical and Computer Engineering*, vol. 10, no. 3, pp. 3315–3322, 2020, doi: 10.11591/ijece.v10i3.pp3315-3322.

[11] E. Mushtaq, A. Zameer, M. Umer, and A. A. Abbasi, "A two-stage intrusion detection system with auto-encoder and LSTMs," *Appl Soft Comput*, vol. 121, p. 108768, 2022, doi: https://doi.org/10.1016/j.asoc.2022.108768.

[12] A. Alferaidi *et al.*, "Distributed Deep CNN-LSTM Model for Intrusion Detection Method in IoT-Based Vehicles," *Math Probl Eng*, vol. 2022, 2022, doi: 10.1155/2022/3424819.

[13] K. O. A. Alimi, K. Ouahada, A. M. Abu-Mahfouz, S. Rimer, and O. A. Alimi, "Refined LSTM Based Intrusion Detection for Denial-of-Service Attack in Internet of Things," *Journal of Sensor and Actuator Networks*, vol. 11, no. 3, Sep. 2022, doi: 10.3390/jsan11030032.

[14] "Intrusion Detection Weblink".

[15] M. A. Khan, "HCRNNIDS: Hybrid convolutional recurrent neural network-based network intrusion detection system," *Processes*, vol. 9, no. 5, 2021, doi: 10.3390/pr9050834.

[16] E. U. H. Qazi, M. H. Faheem, and T. Zia, "HDLNIDS: Hybrid Deep-Learning-Based Network Intrusion Detection System," *Applied Sciences*, vol. 13, no. 8, p. 4921, Apr. 2023, doi: 10.3390/app13084921.

[17] M. A. Khan and J. Kim, "Toward developing efficient Conv-AE-based intrusion detection system using heterogeneous dataset," *Electronics (Switzerland)*, vol. 9, no. 11, pp. 1–17, Nov. 2020, doi: 10.3390/electronics9111771.

[18] O. Almomani, "A Hybrid Model Using Bio-Inspired Metaheuristic Algorithms for Network Intrusion Detection System," *Computers, Materials and Continua*, vol. 68, no. 1, pp. 409–

429, Mar. 2021, doi: 10.32604/cmc.2021.016113.

[19] S. Hosseini and B. M. H. Zade, “New hybrid method for attack detection using combination of evolutionary algorithms, SVM, and ANN,” *Computer Networks*, vol. 173, May 2020, doi: 10.1016/j.comnet.2020.107168.

[20] J. Kim, Y. Shin, and E. Choi, “An Intrusion Detection Model based on a Convolutional Neural Network,” *Journal of Multimedia Information System*, vol. 6, no. 4, pp. 165–172, Dec. 2019, doi: 10.33851/jmis.2019.6.4.165.

[21] P. Wei, Y. Li, Z. Zhang, T. Hu, Z. Li, and D. Liu, “An optimization method for intrusion detection classification model based on deep belief network,” *IEEE Access*, vol. 7, pp. 87593–87605, 2019, doi: 10.1109/ACCESS.2019.2925828.

[22] M. Tang, J. Cui, and Z. Jiang, “An intrusion detection algorithm based on convolutional long-short-term-memory and auto-encoding,” 2023, doi: 10.21203/rs.3.rs-2789937/v1.

[23] E.-H. Qazi, M. Imran, N. Haider, M. Shoaib, and I. Razzak, “An intelligent and efficient network intrusion detection system using deep learning,” *Computers and Electrical Engineering*, vol. 99, p. 107764, 2022, doi: https://doi.org/10.1016/j.compeleceng.2022.107764.

[24] R. Singh, H. Kumar, and R. K. Singla, “An intrusion detection system using network traffic profiling and online sequential extreme learning machine,” *Expert Syst Appl*, vol. 42, no. 22, pp. 8609–8624, Dec. 2015, doi: 10.1016/j.eswa.2015.07.015.

[25] E. Kabir, J. Hu, H. Wang, and G. Zhuo, “A novel statistical technique for intrusion detection systems,” *Future Generation Computer Systems*, vol. 79, pp. 303–318, Feb. 2018, doi: 10.1016/j.future.2017.01.029.

[26] Bülent Ecevit Üniversitesi. Department of Electrical and Electronics Engineering, Bülent Ecevit Üniversitesi. Department of Biomedical Engineering, Bülent Ecevit Üniversitesi. Department of Computer Engineering, and Institute of Electrical and Electronics Engineers, *2016 24th Signal Processing and Communication Application Conference (SIU) = 2016 24.*

Sinyal İşleme Ve İletişim Uygulamaları Kurultayı (SİU) : proceedings : 16-19 May 2016, Zonguldak, Turkey.

[27] S. M. Hosseini Bamakan, H. Wang, T. Yingjie, and Y. Shi, "An effective intrusion detection framework based on MCLP/SVM optimized by time-varying chaos particle swarm optimization," *Neurocomputing*, vol. 199, pp. 90–102, Jul. 2016, doi: 10.1016/j.neucom.2016.03.031.

[28] H. Wang, J. Gu, and S. Wang, "An effective intrusion detection framework based on SVM with feature augmentation," *Knowl Based Syst*, vol. 136, pp. 130–139, Nov. 2017, doi: 10.1016/j.knosys.2017.09.014.

[29] N. Farnaaz and M. A. Jabbar, "Random Forest Modeling for Network Intrusion Detection System," in *Procedia Computer Science*, Elsevier B.V., 2016, pp. 213–217. doi: 10.1016/j.procs.2016.06.047.

[30] H. Bostani and M. Sheikhan, "Hybrid of anomaly-based and specification-based IDS for Internet of Things using unsupervised OPF based on MapReduce approach," *Comput Commun*, vol. 98, pp. 52–71, Jan. 2017, doi: 10.1016/j.comcom.2016.12.001.

[31] R. K. Gunupudi, M. Nimmala, N. Gugulothu, and S. R. Gali, "CLAPP: A self constructing feature clustering approach for anomaly detection," *Future Generation Computer Systems*, vol. 74, pp. 417–429, Sep. 2017, doi: 10.1016/j.future.2016.12.040.

[32] L. Dhanabal and S. P. Shantharajah, "A Study on NSL-KDD Dataset for Intrusion Detection System Based on Classification Algorithms," *International Journal of Advanced Research in Computer and Communication Engineering*, vol. 4, 2015, doi: 10.17148/IJARCCE.2015.4696.

[33] "A Detailed Study on A Benchmark Intrusion Dataset - Kyoto 2006+," *International Journal of Emerging Trends in Engineering Research*, vol. 8, no. 10, pp. 7228–7231, Oct. 2020, doi: 10.30534/ijeter/2020/958102020.

[34] R. Panigrahi and S. Borah, "A detailed analysis of CICIDS2017 dataset for designing

Intrusion Detection Systems Analysis of Selected Clustering Algorithms Used in Intrusion Detection Systems View project Intrusion detection system View project A detailed analysis of CICIDS2017 dataset for designing Intrusion Detection Systems," 2018. [Online]. Available: https://www.researchgate.net/publication/329045441

[35] L. Liu, G. Engelen, T. Lynar, D. Essam, and W. Joosen, "Error Prevalence in NIDS datasets: A Case Study on CIC-IDS-2017 and CSE-CIC-IDS-2018," in *2022 IEEE Conference on Communications and Network Security (CNS)*, 2022, pp. 254–262. doi: 10.1109/CNS56114.2022.9947235.

[36] F. E. Laghrissi, S. Douzi, K. Douzi, and B. Hssina, "Intrusion detection systems using long short-term memory (LSTM)," *J Big Data*, vol. 8, no. 1, Dec. 2021, doi: 10.1186/s40537-021-00448-4.

[37] A. Alferaidi *et al.*, "Distributed Deep CNN-LSTM Model for Intrusion Detection Method in IoT-Based Vehicles," *Math Probl Eng*, vol. 2022, 2022, doi: 10.1155/2022/3424819.

[38] M. A. Khan, "HCRNNIDS: Hybrid convolutional recurrent neural network-based network intrusion detection system," *Processes*, vol. 9, no. 5, 2021, doi: 10.3390/pr9050834.

[39] P. Bedi, N. Gupta, and V. Jindal, "I-SiamIDS: An Improved Siam-IDS for handling class imbalance in Network-based Intrusion Detection Systems."

[40] G. Karatas, O. Demir, and O. K. Sahingoz, "Increasing the Performance of Machine Learning-Based IDSs on an Imbalanced and Up-to-Date Dataset," *IEEE Access*, vol. 8, pp. 32150–32162, 2020, doi: 10.1109/ACCESS.2020.2973219.

[41] B. S. Sharmila and R. Nagapadma, "Intrusion detection system using naive bayes algorithm," in *2019 5th IEEE International WIE Conference on Electrical and Computer Engineering, WIECON-ECE 2019 - Proceedings*, Institute of Electrical and Electronics Engineers Inc., Nov. 2019. doi: 10.1109/WIECON-ECE48653.2019.9019921.

[42] M. H. Shahriar, N. I. Haque, M. Rahman, and M. Alonso Jr, *G-IDS: Generative Adversarial Networks Assisted Intrusion Detection System*. 2020.

[43] H. Wang, J. Gu, and S. Wang, “An effective intrusion detection framework based on SVM with feature augmentation,” *Knowl Based Syst*, vol. 136, pp. 130–139, Nov. 2017, doi: 10.1016/j.knosys.2017.09.014.

List of Publications

[1] Vineet Tomar and Pawan Singh Mehra, **"Deep Learning Bi-LSTM Model for Intrusion Detection in IoT"**, communicated and accepted at 5th International Conference on Advances in Computing, Communication Control and Networking- ICAC3N, IEEE Conference Record No.60023 15th - 16th December 2023, Greater Noida, India

[2] Vineet Tomar and Pawan Singh Mehra, **"Machine and Deep Learning models for IoT Network Intrusion Detection: A Survey"**, communicated and accepted at 5th International Conference on Advances in Computing, Communication Control and Networking- ICAC3N, IEEE Conference Record No.60023 15th - 16th December 2023, Greater Noida, India