

B.E Project Report  
on  
**Hybrid Deep Learning Model for Automatic  
Intrusion Detection System**

Submitted in partial fulfillment of the requirements

For the degree of  
Bachelor of Engineering  
in  
*Computer Engineering*

Submitted by  
Ashutosh Pol(19CE1046)  
Piyul Patel (19CE2030)  
Swastik Chaudhary (19CE1079)  
Vedant Pimple (19CE1032)

Guided by  
Mrs. Siddhi Kadu



**D Y PATIL**  
— RAMRAO ADIK —  
INSTITUTE OF  
**TECHNOLOGY**  
NAVI MUMBAI

Department of Computer Engineering  
Ramrao Adik Institute of Technology,  
Sector 7, Nerul , Navi Mumbai- 400 706  
(Affiliated to University of Mumbai)  
April 2023



**D Y PATIL**  
— RAMRAO ADIK —  
INSTITUTE OF  
**TECHNOLOGY**  
NAVI MUMBAI

# Ramrao Adik Institute of Technology

(Affiliated to the University of Mumbai)

Dr. D. Y. Patil Vidyanagar, Sector 7, Nerul, Navi Mumbai 400 706.

## CERTIFICATE

This is to certify that, the Project-B titled

**“Hybrid Deep Learning Model for Automatic Intrusion  
Detection System ”**

is a bonafide work done by

**Ashutosh Pol(19CE1046)**

**Piyul Patel (19CE2030)**

**Swastik Chaudhary (19CE1079)**

**Vedant Pimple (19CE1032)**

*and is submitted in the partial fulfillment of the requirement for the degree  
of*

**Bachelor of Engineering**

in

**Computer Engineering**

to the

**University of Mumbai.**



---

Supervisor

**(Mrs. Siddhi Kadu)**

---

Project Co-ordinator

**(Mrs. Bhavana Alte)**

---

Head of Department

**(Dr. Amarsingh V. Vidhate )**

---

Principal

**(Dr. Mukesh D. Patil )**

# Project Report Approval for B.E

This is to certify that the project 'A' entitled ***“Hybrid Deep Learning Model for Automatic Intrusion Detection System ”*** is a bonafide work done by ***Mr. Ashutosh Pol, Mr. Piyul Patel, and Mr. Swastik Chaudhary, Mr. Vedant Pimple*** under the supervision of ***Mrs. Siddhi Kadu***. This dissertation has been approved for the award of ***Bachelor’s Degree in Computer Engineering, University of Mumbai***.

Examiners : 1.....

2.....

Supervisors : 1.....

2.....

Principal : .....

Date : .../.../.....

Place : .....

# Declaration

We declare that this written submission represents my ideas in my own words and where other's ideas or words have been included, we have adequately cited and referenced the original sources. We also declare that I have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in my submission. We understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

**Mr. Ashutosh Pol (19CE1046)**

\_\_\_\_\_

**Mr. Piyul Patel (19CE2030)**

\_\_\_\_\_

**Mr. Swastik Chaudhary (19CE1079)**

\_\_\_\_\_

**Mr. Vedant Pimple (19CE1032)**

\_\_\_\_\_

Date : .../.../.....

# Abstract

Network assaults are the most prevalent and important issue in contemporary civilization. Network attacks can affect all networks in some way. To find these risks, intrusion detection is crucial. In order to avoid risks, deep learning and machine learning are used in a variety of fields. However, harmful threats are always growing, necessitating the use of advanced security measures. They are also conveniently accessible at the same time as new Intrusion Detection Systems are investigated and improved. Due to ongoing IP database changes, systematic updates must be made on a regular basis. This project will develop a deep learning-based hybrid ID framework that predicts and classifies harmful cyber attacks using a convolutional recurrent neural network. In order to do this, we will test the system using publicly accessible data and work to increase the accuracy of the intrusion detection. Neural networks can quickly predict the type and categorization of future data by extracting patterns and fingerprints from a raw dataset. To develop a real-time, effective network security framework, the robustness of neural network architecture may be improved.

# Contents

<b>Abstract</b>	<b>iv</b>
<b>List of Figures</b>	<b>vii</b>
<b>List of Tables</b>	<b>viii</b>
<b>1 Introduction</b>	<b>1</b>
1.1 Overview . . . . .	1
1.2 Motivation and Objectives . . . . .	1
1.3 Organisation of Report . . . . .	2
<b>2 Literature Survey</b>	<b>3</b>
2.1 Survey of Existing System . . . . .	3
2.2 Limitations of Existing System . . . . .	4
2.3 Problem Statement . . . . .	4
2.4 Scope . . . . .	4
<b>3 Project Proposal</b>	<b>6</b>
3.1 Proposed Work . . . . .	6
3.2 Proposed Methodology . . . . .	6
3.3 Details of Hardware/Software Requirement . . . . .	7
<b>4 Planning And Formulation</b>	<b>9</b>
4.1 Schedule for Project . . . . .	9
4.2 Detail Plan of Execution . . . . .	9
<b>5 Design of System</b>	<b>11</b>
5.1 Design Diagram with Explanation . . . . .	11
<b>6 Results</b>	<b>14</b>
6.1 Implementation Details . . . . .	14
6.2 Project Outcomes . . . . .	17
6.3 Overall Result . . . . .	22
<b>7 Conclusion and Future Work</b>	<b>23</b>
<b>A Weekly Progress Report Project</b>	<b>24</b>
<b>B Plagiarism Report</b>	<b>25</b>

<b>C Paper Publication</b>	<b>30</b>
<b>D Acknowledgement</b>	<b>31</b>
<b>Bibliography</b>	<b>32</b>

# List of Figures

3.1	NSL-KDD Dataset Description [1] . . . . .	7
4.1	Schedule . . . . .	9
5.1	CNN Architecture[4] . . . . .	11
5.2	Model Training Process . . . . .	13
6.1	CNN Results Table (1,2): Depicts the types of attacks with precision, recall, F1-score, and support. Table (3,4): Depicts the training and validation accuracy and loss for epochs 1 and 2 . . . . .	17
6.2	Training and Validation Accuracy for EPOCH 1 . . . . .	18
6.3	Training and Validation Accuracy for EPOCH 2 . . . . .	18
6.4	CNN bi-LSTM Results Table (1,2): Depicts the types of attacks with precision, recall, F1-score, and support. Table (3,4): Depicts the training and validation accuracy and loss for epochs 1 and 2 . . . . .	19
6.5	Training and Validation Accuracy for EPOCH 1 . . . . .	20
6.6	Training and Validation Accuracy for EPOCH 2 . . . . .	20
6.7	RNN-LSTM Results Table (1,2): Depicts the types of attacks with precision, recall, F1-score, and support. Table (3,4): Depicts the training and validation accuracy and loss for epochs 1 and 2 . . . . .	21
6.8	Training and Validation Accuracy for EPOCH 1 . . . . .	22
6.9	Training and Validation Accuracy for EPOCH 2 . . . . .	22
A.1	Weekly Progress Report . . . . .	24



# List of Tables

6.1	Performance Metrics w.r.t each Model . . . . .	22
-----	--	----

# Chapter 1

## Introduction

### 1.1 Overview

Technology is becoming increasingly omnipresent, networked, and integrated into every aspect of our life. A wide range of key infrastructure sectors, such as health care, banking, transportation, and government, are becoming more and more dependent on cyberspace to provide essential services and conduct daily operations as our society gets more networked. By 2020, more than two-thirds of the world's population will have access to the internet, predicts Cisco's Annual Internet Report. A total of 29.4 billion networked devices will be connected to IP networks, which is three times the current number of people on the planet. The potential of 5G wireless networks to offer incredibly low latency and response times has led to an increase in the speed of network connections. Existing network security designs will need to change as we continue to march toward this high-density, high-speed data trend to safeguard our private and business data. The frequency and speed of cyber security breaches have been increasing in recent years. Security is a problem that transcends technological issues in the hyperconnected era of the internet. A major concern for business and social safety is essential service interruption since it may affect the economy and negatively impact a large section of the population's well-being. More than 71 percentage of governmental and commercial entities reported at least one security breach, according to a 2019 poll by the Canadian Internet Registration Authority.

There was a cyberattack in 2018; The World Economic Forum lists cyberattacks as one of the top ten global risks for the ensuing ten years in its Global Dangers Report 2019. According to their estimates, this risk's disruptive potential could result in up to 90 trillion dollars in net economic damage by 2030 if cybersecurity efforts do not keep up with the growing interconnection.

### 1.2 Motivation and Objectives

The twenty-first century's principal engine for economic growth is technological innovation. The world's main economies are undergoing a digital revolution, and essential elements of this process include cloud computing, big data, social media, the Internet of Things, and artificial intelligence. However, as was already mentioned, the extensive use of technology and the reliance on computer networks expose important infrastructures and institutions to security flaws and intrusion attempts by a variety of bad actors who could steal, destroy, or tamper with crucial data.[1] As technology and software development

increase, cyberattacks get increasingly complicated. In such cases, we must protect our data and privacy, and we must instill a cybersecurity culture in our current networking architecture. Given the persistence of security threats, an effective cybersecurity architecture needs a modern Network Intrusion Detection System (NIDS) to watch the stream of data traversing the network and detect intrusion attempts and malicious activity in order to stop them and their data source before they can reach and damage the core network infrastructures. With the current data traffic explosion, such systems will need to adapt and absorb current technical advancements in order to stay relevant.[2] In order for such systems to continue to be successful in protecting and safeguarding contemporary networks in the face of the current data traffic explosion, they also need to adapt, expand, and integrate with current technological developments. This thesis' major objective is to provide a new method for creating a powerful, real-time network intrusion detection system that uses cutting-edge deep learning algorithms to distinguish between and identify hostile behavior in a steady stream of network data [1].

## 1.3 Organisation of Report

The report is divided in the following chapters.

Chapter 1. Introduction This section introduces the project topic, why we are doing the project and what is being done in the project.

Chapter 2. Literature Survey This section contains all the necessary information which has been referred in the project.

Chapter 3. Proposal The proposal section contains the proposed methodology where the algorithm which is to be implemented is presented along with the complete data flow diagram of the system

Chapter 4. Planning and Formulation This section gives the iterative timeline of the project completion

Chapter 5. Design of the System This section contains the vital information about architecture of the proposed project and how it is proposed to be implemented.

Chapter 6. Expected Results This section contains the detailed explanation of what outcomes are and what future outcomes can be predicted from this project.

Chapter 7. Conclusion This section contains the final conclusive content of the report summarizing the objectives covered under the project.

Chapter 8. Future Work The section explains what all future works can be expected from the existing project.

Chapter 9. References This section consists of the literature survey and a list of research papers, journals, articles and links used for the project.

# Chapter 2

## Literature Survey

In this section, we shall review the literature that is pertinent to the formulation of this argument. The section is divided into three parts that discuss the types of algorithms used in the creation of intrusion detection systems.

### 2.1 Survey of Existing System

**The authors [1] presented that**, Throughout the area of network security, there is a constant search for cyber-attacks that could lead a network to become insecure. Additionally, with a surprising beginning and owing to growing Internet usage, harmful actions within the network are rapidly increasing. It is essential to build a strong intrusion detection system (IDS) that fights unwanted access to network resources in order to detect anomalies in the network and secure data. Recently, a number of interesting approaches have been put out as a cure-all for intrusion detection, but it is still difficult to construct a secure system since attackers frequently alter their tactics to get beyond the system's security measures. The authors of this research has used Support Vector Machine (SVM), K-nearest neighbour (KNN), and other learning (ML) classifiers.

**The authors [2] formulated that** Traditional intrusion detection systems (IDS) are unable to handle increasingly sophisticated threats as they emerge. Convolutional neural networks (CNN) have been successfully used in other industries, such as object detection, face recognition, and healthcare. This study suggests CNN-based IDS utilising the CIC-IDS-2017 dataset. A deep neural network is used to compare the model's performance (DNN). The model is supplied with the grayscale images that were created using the numerical features. Results [2] of this research paper are provided after repeating the model with various hyper parameter values. The CNN model can be regarded as helpful for transfer learning because it shortens training time and provides greater accuracy by taking feature correlation into account. Future heterogeneous transfer learning CNN model attained an accuracy of 99% as compared to DNN with 98

**The authors [3] presented that** The exchange of information has become a global phenomenon in the realm of information technology. An efficient data and network must exist for this to happen. system of protection. Security can be provided by IDS, which can also defend the network against threats and attacks and spot any security holes. The authors [3] used CIC-IDS2017 dataset to test the convolutional neural network-based intrusion detection system . Their algorithm strives to provide low false alarm rates, high

accuracy, and a high detection rate for recently made public datasets. Using CIC-IDS2017 multiclass classification, the model achieved a detection rate of 99.55 percent and a FAR of 0.12.

**The authors [4] presented that** (SVM) models with different network parameters and architecture are used in addition to 1D-CNN models. The models are performed in each experiment for up to 200 iterations on both imbalanced and balanced data, with a learning rate of 0.0001. The performance of 1D-CNN and its variant architectures has surpassed that of traditional machine learning classifiers. This is primarily because CNN has the ability to extract high-level feature representations that represent the abstract form of network traffic connection low-level feature sets.

## 2.2 Limitations of Existing System

Anomaly-based IDS has some limitations in terms of false negatives and false positive alerts, despite having the capacity to identify both known and new assaults. WSNs are also not immune to these security risks and intrusion assaults, which lower their performance and effectiveness. The most common intrusions in WSNs are denial of service (DoS) assaults, which can be launched in a variety of methods. Each of them has a unique method of entering the system. For instance, a number of various attacks aimed at the layers and protocols of WSNs may cause DoS.

## 2.3 Problem Statement

One of the biggest problems today is cyber threats, it can make or break any system. The cyber attacks are of various kinds and intruder may harm the system or get access to sensitive information. To detect these cyber attacks, we built this IDS-Intrusion Detection System. The difficulty lies in creating an IDS with a stable Convolved Neural Network (CNN) that can detect the cyber threats and give an accurate result. The proposed system aims to create a model from the dataset that can detect the cyber attacks and inform the system admin about these attacks.

## 2.4 Scope

As per the objectives defined above, this research consists of two major contributions for automatic and robust NIDS systems. The first contribution is machine learning model using Convolutional Neural Network. The second contribution is related to a hybrid deep learning model called Data Pre-Processing-Robust CNN-LSTM and Data Pre-Processing-Robust CNN-RNN. This proposed contribution focused on the first two objectives mentioned above i.e

- To design the lightweight, effective, and generalized data pre-processing algorithm using Convolutional Neural Network [1].

- To design the hybrid deep learning model for automatic intrusion detection and classification using CNN and LSTM to improve robustness and efficiency [9].
- CNN-RNN: The two primary DNN architectures, Convolutional Neural Network (CNN) and Recurrent Neural Network (RNN), are being extensively researched to improve the performance of intrusion detection systems. By expanding their signature databases, and creating a more realistic and close to real-world ground truth to test an NIDS, this approach tries to improve the ability of NIDS systems to defend against them. This model aims to overcome problems like the existing CNN method's lengthy training process and poor accuracy due to redundant traffic flows and inherent noises. Starting with data pre-processing that was solely focused on the accurate representation of network traffic flows, we will methodically develop this model. More particular, we make use of the enormous ability of recurrent neural networks (RNNs) to separate complicated patterns in a text and produce comparable ones.

# Chapter 3

## Project Proposal

### 3.1 Proposed Work

This study's main objective is to detect the Intrusion and inform the admin about it. One of the biggest problems today is cyberthreat of different kinds - DDoS, R2L, Normal and many more. These threats are major issues not just in the technology world but also in our day-to-day life. Cyber attacks gives intruders access to sensitive information, it can be anything between confidential companies information to personal information. To tackle all of these problems, we built an Intrusion detection system using CNN architecture. We stood out from our academics since we used CNN architectures in our study. Additionally, we conducted a detailed analysis, and the outcomes was better than expected .

### 3.2 Proposed Methodology

Our methodology involves usage of the cloud environment .The Google cloud stage was utilized to foster the IDS design[4]. The stage, which is given by Google, offers an assortment of administrations, the most pertinent to this theory being google process motor, which is a framework as an assistance part for provisioning dynamic figuring groups, cloud AI stage, which offers types of assistance for building and preparing AI and profound learning models, and different cloud network administrations like distributed storage, DNS the executives, and cloud API.

The UNSW15 dataset, which was developed by capturing unorganized parcels with the IXIA PerfectStorm software, will be used. This dataset served as the foundation for the development of the interruption recognition system. The Australian University of New South Wales' Australian Center for Cyber Security (ACCS) dataset has been made available as open source by the Cyber Range Lab. As seen in Figure 3.1, the dataset includes nine different forms of cyberattacks, including DOS, Reconnaissance, Generic, Fuzzers, Shellcode, Worms, and Backdoors, as well as packages that move normally. A total of 2 million organization parcel records are included in the NSL - KDD dataset[1], which is divided into four CSV documents. We'll choose a portion of this data—257,673 records—and divide the chosen parcel into 154,603 records for preparation[1]. Additionally, we will use a testing set and an approval set, both of which include 51,535 data, to carefully assess the application of the applicable deep learning model in various domains.

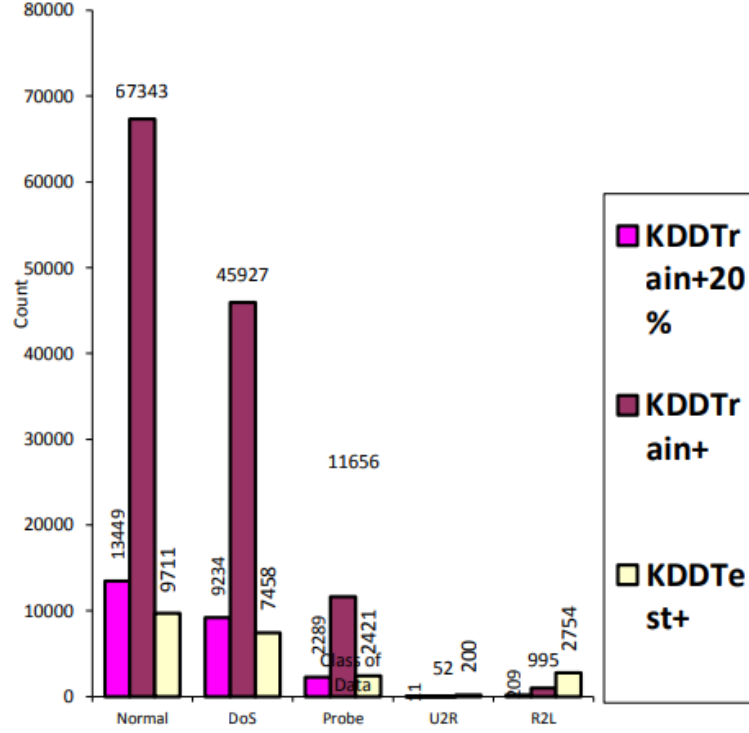


Figure 3.1: NSL-KDD Dataset Description [1]

### 3.3 Details of Hardware/Software Requirement

#### Hardware Details:

Minimum hardware requirements:

- 4-16 GB Ram
- A decent i3-15 processor

For our exploration, we set up two separate bunches in the register motor. We utilized machine type n1-standard-8, which has 8 vCPUs and 30 GB memory, to investigation and construct the model. We set up a group with machine type n1-standard-1, which has 1 vCPU and 3.75 GB memory for area explicit tests. Debian GNU/Linux10 was utilized as the boot working framework in the two bunches [1].

#### Software Details:

Python 3.7 was utilized as the essential programming language in this review, with the profound learning system TensorFlow 1.15 and Keras in the backend. Jupyter Notebook was the essential improvement climate utilized all through the examination. This successful electronic incorporated stage upholds different kinds of information handling



and measurable demonstrating and gives a unified area to all libraries utilized in a venture. Sci-Kit learn was our AI library, Pandas was utilized for information examination and control, NumPy was utilized for n-layered cluster support, and Matplotlib was utilized to produce every one of the charts for the outcomes [2].

### **Algorithms:**

#### **- CNN:**

A Convolutional Neural Network (ConvNet/CNN) is a Deep Learning technique that can take an input picture, assign different components and objects in the image importance (learnable weights and biases), and be able to differentiate between them[2]. In comparison to other classification methods, a ConvNet requires significantly less pre-processing. Contrary to earlier approaches, where filters must be hand-engineered, ConvNets are capable of learning these filters and their attributes[2].

# Chapter 4

## Planning And Formulation

### 4.1 Schedule for Project

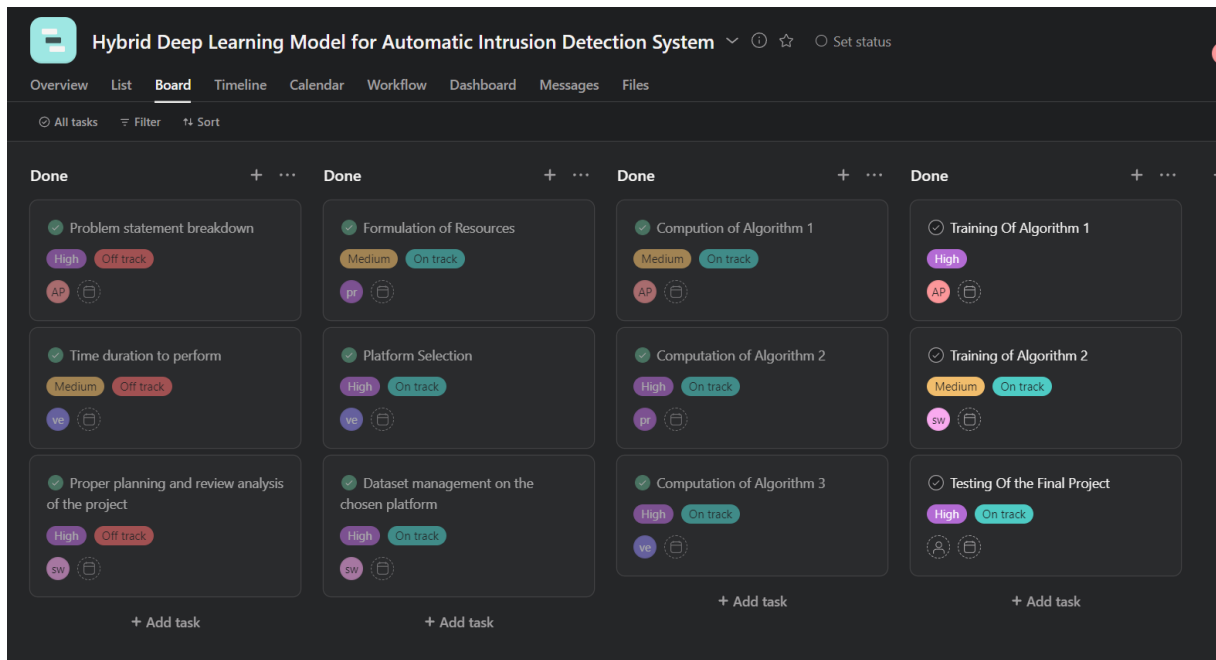


Figure 4.1: Schedule

### 4.2 Detail Plan of Execution

Week1:

Problem Statement Analysis:

- 1) Breaking down the problem statement.
- 2) Gathering information about the problem statement

Week2 :

Time duration to perform:

- 1) Estimation of the time duration required to complete the necessary steps.
- 2) Complex to simple problem formulation

Week3:

Formulation of Resources:

Estimation of resources required to compute in the project.

Week4:

Platform Selection:

Compatibility tests are done to generate results regarding chosen platform.

Week5:

Computation of resources:

Preprocessing of the data in-order to make it clean for the further steps

Week6:

Data Management:

We have considered the size of the dataset and solved this big data problem by proper management steps involved.

Week7:

Training Data:

In this week we have completed our training steps so that we can use our smart data for further processes

Week8:

Testing Data and Final code implementation:

Taking into account the algorithm required, we have generated our machine learning model.

# Chapter 5

## Design of System

### 5.1 Design Diagram with Explanation

The proposed network order and 100 percent interruption identification's framework design will be talked about in this part. The framework engineering of the start to finish profound learning pipeline applied to arrange grouping errands utilizing different subcomponents is displayed in Figure 5.1.

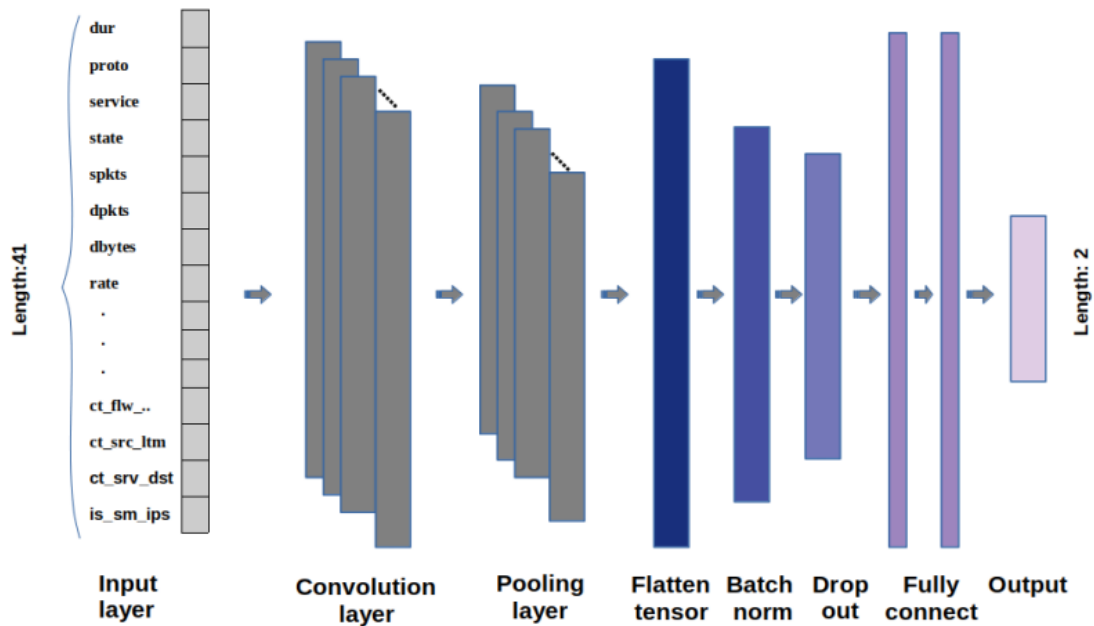


Figure 5.1: CNN Architecture[4]

The framework design of the pipeline might be separated into seven key stages, which are summed up underneath.

- 1.Data Capture:- The pipeline begins by social occasion information from the source space as well as organization streams from the objective area. The IXIA Perfect Storm device was utilized to gather genuine organization traffic and reproduced present day digital assaults as parcel information in the NSL - KDD dataset, which

is likewise investigated more meticulously. The TCPdump program is additionally used to make Pcap records, which are additionally portioned from the 100 GB of gathered information into 1000 MB sections utilizing the TCPdump utility.

- 2.Data Cleaning:- The unprocessed Pcap data are synthesized into reliable features using the Argus and Bro-IDS toolsets. The Argus program analyzes the Pcap data and produces the network flow characteristics, as illustrated. The open-source Bro-IDS tool analyzes network traffic using raw Pcap files and provides connection data like HTTP and FTP requests and responses. A feature set that comprises both flow-based and packet-based features is created by combining the output of these tools.
- 3.Feature Engineering:- We utilize different information pre-handling methods, for example, highlight determination, include scaling, and component standardization, to work on the adequacy and crude elements of our information. These procedures are additionally examined exhaustively in the accompanying segments. Whenever information is used with a model to create ends, the significant objective of component designing is to accomplish the most ideal speed and precision. Highlight designing produces the most dependable portrayal of the information stream’s hidden examples.
- 4.Model Training:- During this stage, we train and construct insightful models fit for learning semantic connections in the information utilizing profound gaining systems and the pre-arranged information from the past advances. The model can foresee the sort of recently seen information by learning the information’s fundamental design, which can be utilized for different order errands. In this situation, we’re utilizing network stream information that incorporates both typical and vindictive parcels to prepare our model so it can perceive and order new organization stream information in light of those rules. This is an iterative interaction wherein we utilize marked information to further develop our model’s characterization capacities until it can make right expectations.
- 5.Model Evaluation:- We assess the model utilizing a subset of inconspicuous information that was not used during its preparation after we are content with the insightful model outcomes from the preparation stage. To gauge the exhibition and viability of the pre-owned model, we apply preset assessment measures. The assessment measures for the interruption identification framework portrayed in this proposition are laid forward in Section 4.6. In light of the assessment findings, we may calibrate the applied model’s numerous hyper parameters to retrain the model, increasing our outcomes with each focus.
- 6. Transfer Learning Methodology:- We will save the model and its loads in the HDF5 design, which is intended to store colossal and confounded information progressively, when the model produces acceptable outcomes in view of the given appraisal measures. From that point forward, the model is communicated to our objective space by means of a totally new organization stream with similar designed properties. Assuming that the result marks in the objective space should be transformed, we will thaw the model’s last layers and retrain them. We utilize the got data from the source area by utilizing a pre-prepared model with unblemished

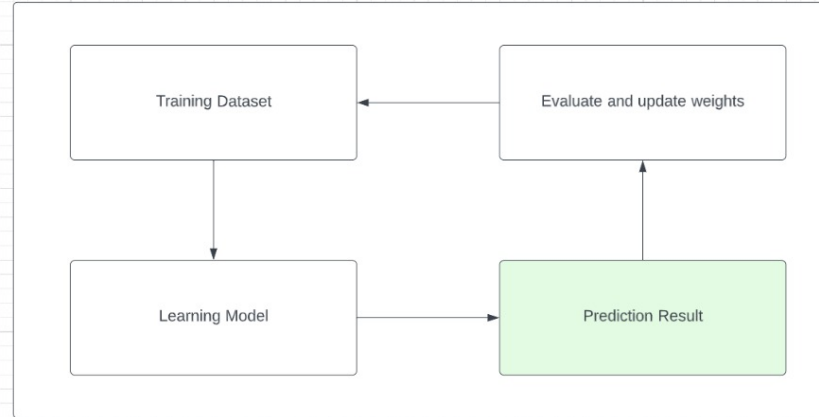


Figure 5.2: Model Training Process

weight boundaries, which diminishes preparing time and PC resources. The source space model would sum up suitably in the objective space assuming it was huge and strong, and it had been prepared with countless preparation tests. We would show our exploratory results from the exchange learning philosophy in our respective result methodology.

- **7. Deployment:-** Based on our predefined assessment measures and measurements, the architect-ed model has gone through a few emphases in both the source and target areas. We can convey the framework in genuine creation setting after we are satisfied with the order results. The advantage of utilizing move learning techniques is that we can now cycle and develop our model, improving its exhibition capacities by noticing and figuring out how to order another subset of organization action. This fosters the model over the long run, permitting it to perceive a wide scope of bundle information in network traffic while working in a real world setting, which is difficult to accomplish utilizing conventional profound learning strategies. We iteratively train the brought together model to arrange network bundles, as shown. The model is then coordinated into the Intrusion Detection System, which gathers network traffic and applies different information pre-handling strategies to further develop characterization precision. Utilizing the exchange learning approach, this planned design is then moved to an alternate area with less information and calculation assets, where it adjusts to the objective space to keep up with its exhibition on an inconspicuous information stream while essentially further developing its general order speed. Regardless of the information and computational limits, this approach might be utilized to introduce enormous and strong profound learning put together interruption recognition frameworks with respect to asset inadequate edge gadgets to save their security.

# Chapter 6

## Results

### 6.1 Implementation Details

- CNN

Step 1:

Firstly, we get integrated with our google drive account We mount our Google Drive into your Google Collab notebook so that we can navigate through folders in our Google Drive in case our project requires multiple files from different locations.

Step 2: In this step we import various libraries that are necessary in the project so that we are able to perform the essential CNN analysis for our model.

Step 3: We train our data using the pandas data frame object on the initial specified columns of the dataset in order to prepare our data

Step 4: This usually takes care of the major pre-processing steps involved that is basically cleaning the data before we taking care of the outliers present in our data.

Step 5: We use k-fold cross validation technique to re sample our data which eventually improves the models prediction.

Step 6: We add convolutional neural network layers to our sequential model. These layers are combined in a stacked manner where, each layer depicts some certain importance contributing to our performance.

Step 7: Training of our processed data takes place and hence this trained dataset gets fitted into our model

Step 8: The confusion matrix is being calculated on our predicted variable and the test data in order to provide a suitable accuracy measure

Step 9: We plot our training accuracy as well as the validation accuracy which in turn also displays the calculated precision, recall, f1score and support values of the different IDS attacks detected by our system. Hence the result gets displayed here.

## ● CNN bi-LSTM

Step 1:

Firstly, we get integrated with our google drive account We mount our Google Drive into your Google Collab notebook so that we can navigate through folders in our Google Drive in case our project requires multiple files from different locations.

Step 2: In this step we import various libraries that are necessary in the project so that we are able to perform the essential CNN bi-LSTM analysis for our model.

Step 3: We train our data using the pandas data frame object on the initial specified columns of the dataset in order to prepare our data

Step 4: This usually takes care of the major pre-processing steps involved that is basically cleaning the data before we take care of the outliers present in our data.

Step 5: We have used MaxPooling1D layer twice in the model. We have then added LSTM layer to a Convolutional Network Layer Network as a hybrid model.

Step 6: Training of our processed data takes place and hence this trained dataset gets fitted into our model

Step 7: The confusion matrix is being calculated on our predicted variable and the test data in order to provide a suitable accuracy measure

Step 8: We plot our training accuracy as well as the validation accuracy which in turn also displays the calculated precision, recall, f1score and support values of the different IDS attacks detected by our system. Hence the result gets displayed here.

## ● RNN LSTM

Step 1:

Firstly, we get integrated with our google drive account We mount our Google Drive into your Google Collab notebook so that we can navigate through folders in our Google Drive in case our project requires multiple files from different locations.

Step 2: In this step we import various libraries that are necessary in the project so that we are able to perform the essential RNN LSTM analysis for our model.

Step 3: We train our data using the pandas data frame object on the initial specified columns of the dataset in order to prepare our data

Step 4: This usually takes care of the major pre-processing steps involved that is basically cleaning the data before we take care of the outliers present in our data.

Step 5: We have added recurrent activation- hard sigmoid which gives a hybrid nature to our existing LSTM model.



Step 6: Training of our processed data takes place and hence this trained dataset gets fitted into our model

Step 7: The confusion matrix is being calculated on our predicted variable and the test data in order to provide a suitable accuracy measure

Step 8: We plot our training accuracy as well as the validation accuracy which in turn also displays the calculated precision, recall, f1score and support values of the different IDS attacks detected by our system. Hence the result gets displayed here.

## 6.2 Project Outcomes

### • CNN

Importing the various libraries into the project is the most crucial task because they will be used to do the crucial CNN analysis for our model. In order to prepare our data, we train it using pandas data frames on the dataset's initial stated columns. Pandas can analyze datasets larger than those that fit in memory because it is used for in-memory analytics. The CNN model analysis is carried out after the libraries have been imported. In order to prepare our data, the dataset is then trained using pandas data frame objects on the first provided columns. Typically, this completes the main pre-processing stages, which entail cleansing the data before dealing with any outliers that may be present. We resample our data using k-fold cross-validation approaches, which enhances the model's prediction. Our processed data is trained, and this trained dataset is then incorporated into our model.

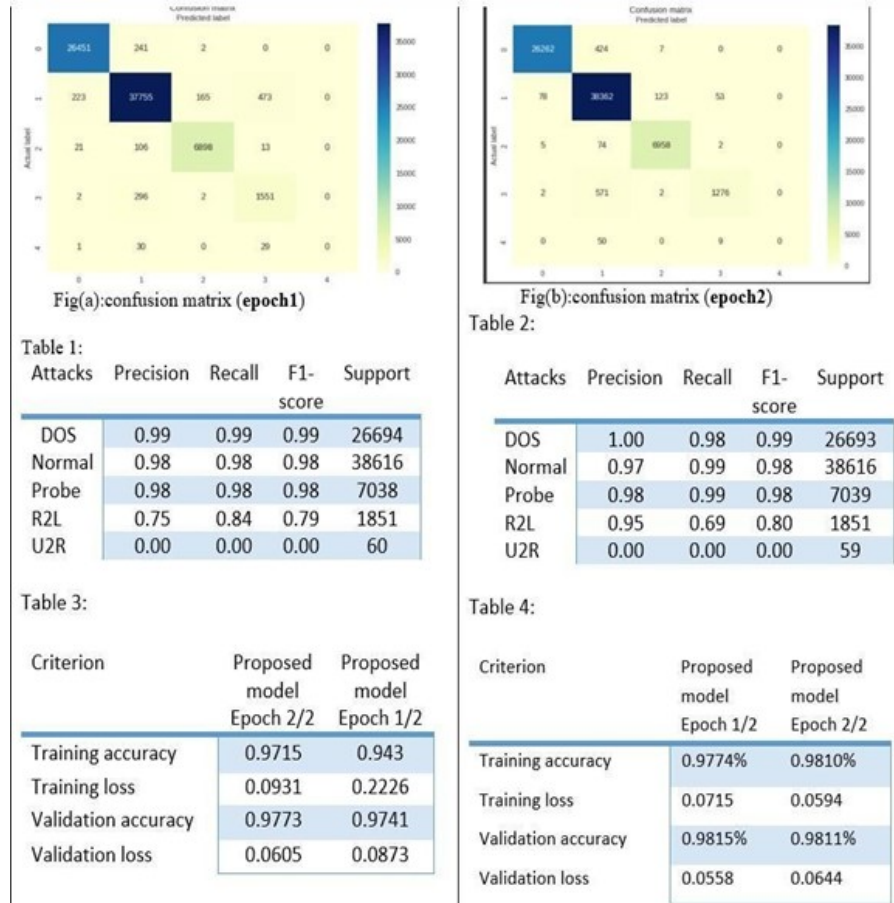


Figure 6.1: CNN Results

Table (1,2): Depicts the types of attacks with precision, recall, F1-score, and support.

Table (3,4): Depicts the training and validation accuracy and loss for epochs 1 and 2

In the above-given matrix, a total of 74258 trained samples are used and all the 74258 samples are tested and executed. The DoS attack consists of the 26451 tuples where each of the tuples shows the true positive nature which shows that the predictions are completely true for our result. The Normal attack which includes all the intrinsic attacks shows the tuple of 38616 which supports true positive plus False Positive which means

that these results are true in our results. The probe attack of tuple 7038 supports the true positive plus false positive nature i.e these predictions are again considered as true in our results. Similarly, the R2L and U2R consist of the tuple of 1851 and 60. R2L supports the True positive of nature i.e. the predictions are explained in our result whereas U2R supports the true Positive plus false positive in nature which these predictions depict in our result. The number of epochs, which is a hyperparameter, controls how many times the learning algorithm will run through the full training dataset. One epoch indicates that the internal model parameters have had a chance to be updated for each sample in the training dataset. Only when your dataset contains a large amount of data does increasing epochs make sense. However, your model will ultimately reach a point when adding more epochs will not enhance accuracy.



Figure 6.2: Training and Validation Accuracy for EPOCH 1



Figure 6.3: Training and Validation Accuracy for EPOCH 2

### • CNN bi-LSTM

In this model, integration of both the CNN and the LSTM takes place. It is a also sequential model where it repeat process for cnn and adds another set of layers as an attempt to make a hybrid model for better results.

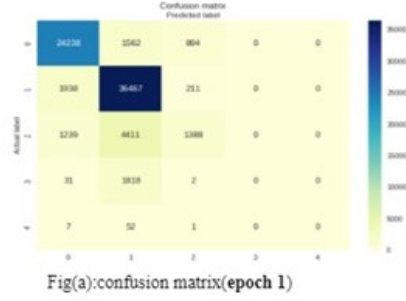


Table 1:

Attacks	Precision	Recall	F1-score	Support
DOS	0.88	0.91	0.90	26694
Normal	0.82	0.94	0.88	38616
Probe	0.56	0.20	0.29	7038
R2L	0.00	0.00	0.00	1851
U2R	0.00	0.00	0.00	60

Table 3:

Criterion	Proposed model Epoch 2/2	Proposed model Epoch 1/2
Training accuracy	0.8826	0.8262
Training loss	0.3954	0.5331
Validation accuracy	0.8362	0.8264
Validation loss	0.5197	0.4538



Table 2:

Attacks	Precision	Recall	F1-score	Support
DOS	0.93	0.91	0.92	26693
Normal	0.91	0.97	0.94	38616
Probe	0.74	0.67	0.71	7039
R2L	0.26	0.05	0.08	1851
U2R	0.00	0.00	0.00	59

Table 4:

Criterion	Proposed model Epoch 2/2	Proposed model Epoch 1/2
Training accuracy	0.9150	0.8970
Training loss	0.2706	0.3288
Validation accuracy	0.8973	0.8940
Validation loss	0.3431	0.3565

Figure 6.4: CNN bi-LSTM Results

Table (1,2): Depicts the types of attacks with precision, recall, F1-score, and support. Table (3,4): Depicts the training and validation accuracy and loss for epochs 1 and 2

In the above-given matrix, a total of 74258 trained samples are used and all the 74258 samples are tested and executed. The DoS attack consists of the 26193 tuples where each of the tuples shows the true positive nature which shows that the predictions are completely true for our result. The Normal attack which includes all the intrinsic attacks shows the tuple of 38001 which supports true positive plus False Positive which means that these results are true in our results. The probe attack of tuple 6979 supports the true positive plus false positive nature i.e these predictions are again considered as true in our results. Similarly, the R2L and U2R consist of the tuple of 1400 and 60. R2L supports the True positive of nature i.e. the predictions are explained in our result whereas U2R supports the true Positive plus false positive in nature which these predictions depict in our result.



Figure 6.5: Training and Validation Accuracy for EPOCH 1

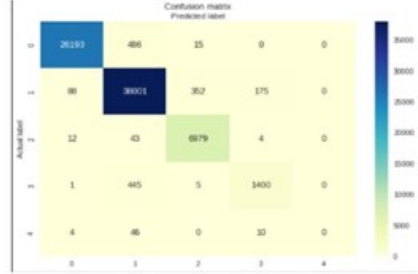


Figure 6.6: Training and Validation Accuracy for EPOCH 2

The training vs. validation accuracy for epoch 1 is shown in the above figure, and it shows that our validation dataset performs better than our training dataset. As a result, when fresh data records are taken into account, the cleaned dataset trained in our model improves itself linearly. With such a case, we may move to epoch 2 and determine whether or not our model is overfitted.

## • RNN-LSTM

Here the LSTM (Long -Short Term Memory) LSTM assists RNN in remembering the critical inputs needed to generate the necessary output. By integrating the hard-sigmoid recurrent activation method we create a hybrid model which depicts the following results:



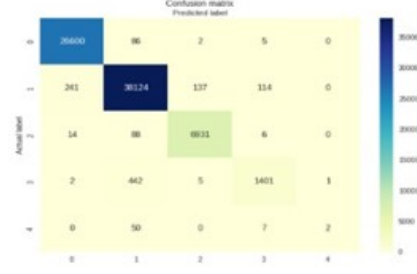
Fig(a):confusion matrix(epoch 1)

Table 1:

Attacks	Precision	Recall	F1-score	Support
DOS	1.00	0.98	0.99	26694
Normal	0.97	0.98	0.98	38616
Probe	0.95	0.99	0.97	7038
R2L	0.88	0.76	0.81	1851
U2R	0.00	0.00	0.00	60

Table 3:

Criterion	Proposed model Epoch 2/2	Proposed model Epoch 1/2
Training accuracy	0.9721	0.9582
Training loss	0.0844	0.1305
Validation accuracy	0.9773	0.9225
Validation loss	0.0654	0.2237



Fig(b):confusion matrix(epoch 2)

Table 2:

Attacks	Precision	Recall	F1-score	Support
DOS	0.99	1.00	0.99	26693
Normal	0.98	0.99	0.99	38616
Probe	0.98	0.98	0.98	7039
R2L	0.91	0.76	0.83	1851
U2R	0.67	0.03	0.06	59

Table 4:

Criterion	Proposed model Epoch 2/2	Proposed model Epoch 1/2
Training accuracy	0.9721	0.9582
Training loss	0.0569	0.0666
Validation accuracy	0.9805	0.9776
Validation loss	0.9838	0.9780

Figure 6.7: RNN-LSTM Results

Table (1,2): Depicts the types of attacks with precision, recall, F1-score, and support.

Table (3,4): Depicts the training and validation accuracy and loss for epochs 1 and 2

The DoS attack consists of the 24238 tuples where each of the tuples shows the true positive nature which shows that the predictions are completely true for our result. The Normal attack which includes all the intrinsic attacks shows the tuple of 36467 which supports true positive plus False Positive which means that these results are true in our results. The probe attack of tuple 1388 supports the true positive i.e these predictions are again considered as true in our results. Similarly, the R2L and U2R consist of the tuple of 1820 and 60. R2L supports the True positive of nature i.e. the predictions are explained in our result whereas U2R supports the true Positive plus false positive in nature which these predictions depict in our result.



Figure 6.8: Training and Validation Accuracy for EPOCH 1



Figure 6.9: Training and Validation Accuracy for EPOCH 2

### 6.3 Overall Result

Criterion	CNN	CNN bi-LSTM	RNN-LSTM
Training Accuracy	0.9582	0.9721	0.8826
Training Loss	0.0715	0.0569	0.2706
Validation Accuracy	0.9667	0.9805	0.8973
Validation Loss	0.0644	0.3431	0.0654

Table 6.1: Performance Metrics w.r.t each Model

## Chapter 7


# Conclusion and Future Work

In conclusion, the hybrid models created for detecting cyberattacks have demonstrated outstanding levels of accuracy, with CNN bi-LSTM and RNN-LSTM generalisation accuracy scores of 0.97 and 0.88, respectively. Hence, from the above results, we can conclude that the CNN bi-LSTM model is the most efficient predictive model for IDS. These findings show how well the model works at spotting cyber-attacks and how it might help with cybersecurity. Furthermore, a notable advancement in cybersecurity is the models' unaided capacity to recognise crucial components for cyber-attack detection. It lessens the need for manual intervention, conserves time and money, and guarantees that the models continue to be useful in recognising fresh cyberthreats as they appear. One of the most important steps towards enhancing and improving the efficiency of these instruments is the consideration devoted to evaluating the general technical proficiency of various technologies, including the hybrid model. Researchers and cybersecurity specialists can find areas for development, refine their methods, and create more reliable and accurate cybersecurity solutions through regular reviews. Overall, if we want to make sure that our online activities are safe and secure from cyber threats, we need to employ hybrid models for cyber-attack detection and continuously assess their technical competence. We must constantly build and improve our cybersecurity solutions to keep ahead of the threat landscape as technology develops and cyberattacks get more sophisticated.



# Appendix A

## Weekly Progress Report Project


**Ramrao Adik Institute of Technology**  
**Department of Computer Engineering**  
**Weekly Project Performance Report for BE, Computer Engg. 2022-2023 Even Sem**

Group No: - G-15  
 Project Title: - Hybrid Deep Learning Model for Intrusion Detection system

		Name of Students 1: Vedant Pimple		Name of Students 2: Piyu Patel		Name of Students 3: Ashutosh Pol		Name of Student 4: Swastik Chaudhary		
Week No.	Expected Topics to be Covered	Progress Status	Student 1 Sign	Progress Status	Student 2 Sign	Progress Status	Student 3 Sign	Progress Status	Student 4 Sign	Suggestions if any
1.	Design of proposed system	A	<i>Vedant Pimple</i>	A	<i>Piyu Patel</i>	A	<i>Ashutosh Pol</i>	A	<i>Swastik Chaudhary</i>	Good
2.	Implementation Details Module wise	A	<i>Vedant Pimple</i>	A	<i>Piyu Patel</i>	A	<i>Ashutosh Pol</i>	A	<i>Swastik Chaudhary</i>	
3.	Implementation Details Module wise	A	<i>Vedant Pimple</i>	A	<i>Piyu Patel</i>	A	<i>Ashutosh Pol</i>	A	<i>Swastik Chaudhary</i>	
4.	Full Implementation Details	A	<i>Vedant Pimple</i>	A	<i>Piyu Patel</i>	A	<i>Ashutosh Pol</i>	A	<i>Swastik Chaudhary</i>	
5.	Cost and benefit analysis	A	<i>Vedant Pimple</i>	A	<i>Piyu Patel</i>	A	<i>Ashutosh Pol</i>	A	<i>Swastik Chaudhary</i>	
6.	Testing	A	<i>Vedant Pimple</i>	A	<i>Piyu Patel</i>	A	<i>Ashutosh Pol</i>	A	<i>Swastik Chaudhary</i>	
7.	Result Analysis	A	<i>Vedant Pimple</i>	A	<i>Piyu Patel</i>	A	<i>Ashutosh Pol</i>	A	<i>Swastik Chaudhary</i>	Do apply on open repo
8.	Report Writing	A	<i>Vedant Pimple</i>	A	<i>Piyu Patel</i>	A	<i>Ashutosh Pol</i>	A	<i>Swastik Chaudhary</i>	Good
9.	Report Writing	A	<i>Vedant Pimple</i>	A	<i>Piyu Patel</i>	A	<i>Ashutosh Pol</i>	A	<i>Swastik Chaudhary</i>	
10.	Conclusion / Future Work	A	<i>Vedant Pimple</i>	A	<i>Piyu Patel</i>	A	<i>Ashutosh Pol</i>	A	<i>Swastik Chaudhary</i>	
11.	Research Paper	A	<i>Vedant Pimple</i>	A	<i>Piyu Patel</i>	A	<i>Ashutosh Pol</i>	A	<i>Swastik Chaudhary</i>	Good

A: Satisfactory      B: Average      C: Needs Improvement

*Siddhi Faden*  
 Project Guide Name and Sign

Figure A.1: Weekly Progress Report

# Appendix B

## Plagiarism Report

Hypbrid IDS Project Report G15 (1).pdf

ORIGINALITY REPORT

21%  
SIMILARITY INDEX

PRIMARY SOURCES

1	<a href="http://ir.lib.uwo.ca">ir.lib.uwo.ca</a> Internet	373 words — 6%
2	<a href="http://akshaypakhle10.github.io">akshaypakhle10.github.io</a> Internet	197 words — 3%
3	<a href="http://www.coursehero.com">www.coursehero.com</a> Internet	147 words — 2%
4	<a href="http://www.itm-conferences.org">www.itm-conferences.org</a> Internet	127 words — 2%
5	<a href="http://assets.researchsquare.com">assets.researchsquare.com</a> Internet	54 words — 1%
6	Venkata Ramani Varanasi, Shaik Razia. "CNN Implementation for IDS", 2021 3rd International Conference on Advances in Computing, Communication Control and Networking (ICAC3N), 2021 Crossref	50 words — 1%
7	<a href="http://blog.paperspace.com">blog.paperspace.com</a> Internet	44 words — 1%
8	Takeshi Nakazawa, Deepak V. Kulkarni. "Wafer Map Defect Pattern Classification and Image Retrieval using Convolutional Neural Network", IEEE Transactions on Semiconductor Manufacturing, 2018	40 words — 1%

- 
- 9 Soroush M. Sohi, Jean-Pierre Seifert, Fatemeh Ganji. "RNNIDS: Enhancing Network Intrusion Detection Systems through Deep Learning", *Computers & Security*, 2020  
Crossref 28 words — < 1%
- 
- 10 [gecgudlavalleru.ac.in](http://gecgudlavalleru.ac.in)  
Internet 24 words — < 1%
- 
- 11 Abhiramy Ajith, Abin Oommen Philip, Sreela Sreedhar, M U Sreeja. "Road Accident Detection from CCTV Footages using Deep Learning", 2022 3rd International Conference on Smart Electronics and Communication (ICOSEC), 2022  
Crossref 22 words — < 1%
- 
- 12 Shuang Cai, Ahmet Palazoglu, Laibin Zhang, Jinqiu Hu. "Process alarm prediction using deep learning and word embedding methods", *ISA Transactions*, 2018  
Crossref 21 words — < 1%
- 
- 13 Sepp Hochreiter, Jürgen Schmidhuber. "Long Short-Term Memory", *Neural Computation*, 1997  
Crossref 20 words — < 1%
- 
- 14 [mospace.umsystem.edu](http://mospace.umsystem.edu)  
Internet 20 words — < 1%
- 
- 15 Meliboev Azizjon, Alikhanov Jumabek, Wooseong Kim. "1D CNN based network intrusion detection with normalization on imbalanced data", 2020 International Conference on Artificial Intelligence in Information and Communication (ICAIIIC), 2020  
Crossref 18 words — < 1%
-

16	<a href="http://www.researchgate.net">www.researchgate.net</a> Internet	18 words — < 1%
17	<a href="http://ijsrset.com">ijsrset.com</a> Internet	17 words — < 1%
18	<a href="http://www.termpaperwarehouse.com">www.termpaperwarehouse.com</a> Internet	15 words — < 1%
19	Riya Gupta, Yogesh Deshpande, Manasi Kulkarni. "Handwritten Mathematical Equation Recognition and Solver", 2022 3rd International Conference on Issues and Challenges in Intelligent Computing Techniques (ICICT), 2022 Crossref	13 words — < 1%
20	<a href="http://pdffox.com">pdffox.com</a> Internet	13 words — < 1%
21	<a href="http://www.turing.com">www.turing.com</a> Internet	13 words — < 1%
22	"International Conference on Communication, Computing and Electronics Systems", Springer Science and Business Media LLC, 2020 Crossref	11 words — < 1%
23	Zhen Yang, Xiaodong Liu, Tong Li, Di Wu, Jinjiang Wang, Yunwei Zhao, Han Han. "A systematic literature review of methods and datasets for anomaly-based network intrusion detection", Computers & Security, 2022 Crossref	11 words — < 1%
24	<a href="http://www.epp.eu">www.epp.eu</a> Internet	11 words — < 1%
25	<a href="http://irep.iium.edu.my">irep.iium.edu.my</a> Internet	10 words — < 1%

26	<a href="http://www.ijnc.org">www.ijnc.org</a> Internet	10 words — < 1%
27	<a href="http://www.mdpi.com">www.mdpi.com</a> Internet	10 words — < 1%
28	"Quality, Reliability, Security and Robustness in Heterogeneous Systems", Springer Science and Business Media LLC, 2021 Crossref	9 words — < 1%
29	<a href="http://openaccess.altinbas.edu.tr">openaccess.altinbas.edu.tr</a> Internet	9 words — < 1%
30	<a href="http://papers.academic-conferences.org">papers.academic-conferences.org</a> Internet	9 words — < 1%
31	<a href="http://repository.tudelft.nl">repository.tudelft.nl</a> Internet	9 words — < 1%
32	Crow, John Charles. "Neural Networks based Surrogate Models for Quantification of Gap and Overlap Defects in Tow Steered Composites", San Diego State University, 2022 ProQuest	8 words — < 1%
33	Dylan Chou, Meng Jiang. "A Survey on Data-driven Network Intrusion Detection", ACM Computing Surveys, 2022 Crossref	8 words — < 1%
34	<a href="http://ceur-ws.org">ceur-ws.org</a> Internet	8 words — < 1%
35	<a href="http://dergipark.org.tr">dergipark.org.tr</a> Internet	8 words — < 1%

---

36 [mdpi-res.com](https://mdpi-res.com) 8 words — < 1%  
Internet

---

37 [www.hindawi.com](https://www.hindawi.com) 8 words — < 1%  
Internet

---

38 Sharvari Adiga, DV Vaishnavi, Suchitra Saxena, Shikha Tripathi. "Multimodal Emotion Recognition for Human Robot Interaction", 2020 7th International Conference on Soft Computing & Machine Intelligence (ISCMI), 2020 7 words — < 1%  
Crossref

---

39 "Applications of Artificial Intelligence and Machine Learning", Springer Science and Business Media LLC, 2021 6 words — < 1%  
Crossref

---

40 Akshay Shringarpure, Ronak Shetty, Ajinkya Surve, Amarsinh Vidhate. "Sports Injury Prediction System using Random Forest Classifier", ITM Web of Conferences, 2022 6 words — < 1%  
Crossref

---

EXCLUDE QUOTES OFF  
EXCLUDE BIBLIOGRAPHY ON

EXCLUDE SOURCES OFF  
EXCLUDE MATCHES OFF

# Appendix C

## Paper Publication

Network attacks constitute the most ubiquitous and pivotal problem confronting modern society. All networks are susceptible to network threats to some degree. Intrusion Detection is imperative to identify these threats. Although deep learning and machine learning are applied in a variety of sectors to prevent attacks, malicious threats continue to be on the ascent, necessitating the implementation of advanced security solutions. The software's purpose is to create an extensive and reliable profound learning Intrusion Detection System in a supply region with a significant allotment of information as well as figuring assets. Because of frequent changes in IP, databases must be modified systematically on a regular basis. The proposed solution advocates the employment of Convolutional Neural Networks to develop an Intrusion Detection framework whose foundation is deep learning., which will predict and categorize hostile cyber-attacks. A raw dataset can be used by neural networks to extract signatures and patterns to anticipate the characteristic and classification of future data at a faster rate. The Automatic Intrusion Detection System (IDS) will be developed using the CNN, CNN BI LSTM and RNN-LSTM architecture, for latent feature abstraction, memory retention, and categorization skills. Utilizing the exchange learning procedure, this proposition exhibits that powerful profound learning-based IDS frameworks might be executed on certifiable gadgets with less assets while safeguarding economy and speed.

As such we have chosen to submit our paper to the **International Conference on Data Science and Network Security, organized by Kalpataru Institute of Technology, Titpur, Karnataka**, as we believe it is an appropriate platform to showcase our research and engage with a wider scientific community.

# Appendix D

## Acknowledgement

We take this opportunity to express my profound gratitude and deep regards to our guide **Mrs.Siddhi Kadu** for his/her exemplary guidance, monitoring and constant encouragement throughout the completion of this report. We truly grateful to his/her efforts to improve our understanding towards various concepts and technical skills required in our project. The blessing, help and guidance given by him/her time to time shall carry us a long way in the journey of life on which we are about to embark.

We take this privilege to express our sincere thanks to **Dr. Mukesh D. Patil**, Principal, RAIT for providing the much necessary facilities. We are also thankful to **Dr. Amarsinh V. Vidhate** , Head of Department of Computer Engineering, Project Co-ordinator **Mrs. Bhavana Atle**,Department of Computer Engineering, RAIT, Nerul Navi Mumbai for their generous support.

Last but not the least we would also like to thank all those who have directly or indirectly helped us in completion of this thesis.

**Mr. Ashutosh Pol**  
**Mr. Piyul Patel**  
**Mr. Swastik Chaudharyl**  
**Mr. Vedant Pimple**



# Bibliography

- [1] P. Illavarason and B. Kamachi Sundaram, "A Study of Intrusion Detection System using Machine Learning Classification Algorithm based on different feature selection approach," 2019 Third International conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC), 2019, pp. 295-299, doi: 10.1109/I-SMAC47947.2019.9032499.
- [2] V. R. Varanasi and S. Razia, "CNN Implementation for IDS," 2021 3rd International Conference on Advances in Computing, Communication Control and Networking (ICAC3N), 2021, pp. 970-975, doi: 10.1109/ICAC3N53548.2021.9725426.
- [3] A. H. Halbouni, T. S. Gunawan, M. Halbouni, F. A. A. Assaig, M. R. Effendi and N. Ismail, "CNN-IDS: Convolutional Neural Network for Network Intrusion Detection System," 2022 8th International Conference on Wireless and Telematics (ICWT), 2022, pp. 1-4, doi: 10.1109/ICWT55831.2022.9935478.
- [4] M. Azizjon, A. Jumabek and W. Kim, "1D CNN based network intrusion detection with normalization on imbalanced data," 2020 International Conference on Artificial Intelligence in Information and Communication (ICAIIIC), 2020, pp. 218-224, doi: 10.1109/ICAIIIC48513.2020.9064976.
- [5] DL-IDS: Extracting Features Using CNN-LSTM Hybrid Network for Intrusion Detection System
- [6] Moustafa, N., Turnbull, B., & Choo, K. K. R. (2018). An ensemble intrusion detection technique based on proposed statistical flow features for protecting network traffic of internet of things. *IEEE Internet of Things Journal*, 6(3), 4815-4830.
- [7] Alqahtani, Mnahi, et al. "A genetic-based extreme gradient boosting model for detecting intrusions in wireless sensor networks." *Sensors* 19.20 (2019): 4383.
- [8] Xiao, Yihan, et al. "An intrusion detection model based on feature reduction and convolutional neural networks." *IEEE Access* 7 (2019): 42210-42219.
- [9] Sun, Pengfei, et al. "DL-IDS: Extracting features using CNN-LSTM hybrid network for intrusion detection system." *Security and communication networks* 2020 (2020).
- [10] Sohi, Soroush M., Jean-Pierre Seifert, and Fatemeh Ganji. "RNNIDS: Enhancing network intrusion detection systems through deep learning." *Computers Security* 102 (2021): 102151.

## List of Abbreviations

Abbreviation : Definition

IDS : Intrusion Detection System  
IP : Internet Protocol  
CNN : Convolutional Neural Network  
LSTM : Long Short Term Memory  
RNN : Recurrent Neural Network  
NIDS : Network Intrusion Detection System  
IDES : Intrusion Detection Expert System  
HIDS : Host-based Intrusion Detection System  
OS : Operating System  
TCP : Transfer Control Protocol  
UDP : User Datagram Protocol  
API : Application Programming Interface  
AI : Artificial Intelligence  
SVM : Support Vector Machine  
AUC : Area Under Curve  
ROC : Receiver Operating Characteristic