

# **Network Intrusion Detection in Civil Aviation Based on Improved Convolutional Neural Network**

A DISSERTATION

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

**MASTER OF TECHNOLOGY**  
in  
**SOFTWARE ENGINEERING**

Submitted by:  
**SACHIN BHURE**  
2K22/SWE/16

Under the Supervision of  
**Mr. Rahul**  
**Assistant Professor**



**DEPARTMENT OF SOFTWARE ENGINEERING  
DELHI TECHNOLOGICAL UNIVERSITY  
(Formerly Delhi College of Engineering)  
Bawana Road, Delhi – 110042**

**May, 2024**

## ACKNOWLEDGEMENT

The achievement of Major Project II necessitates the assistance and support of a large number of individuals and an organisation. This project's report writing opportunity allows me to thank everyone who contributed to the project's successful completion. I would want to express my sincere gratitude to my supervisor, Mr. Rahul, for allowing me to work on my project under his supervision. Thank you very much for your support, encouragement and suggestions; without them, our work would not have been successful. His unwavering support and inspiration helped me to see that the process of learning is more important than the end result.

I want to express my sincere thanks to the faculty and personnel at the institution for providing us with a infrastructure, laboratories, library, suitable educational resources, testing facilities, and a working atmosphere that didn't interfere with our ability to complete our work.

I would also like to thank all of my friends and classmates for their unwavering support. They have assisted me in every way, providing me with fresh ideas, the knowledge I needed, and the will to finish the assignment. I want to express my gratitude to my parents for always supporting me after finishing my task.



**Sachin Bhure**  
**(2K22/SWE/16)**  
**M.Tech (Software Engineering)**  
**Delhi Technological University**

**DEPARTMENT OF SOFTWARE ENGINEERING**  
**DELHI TECHNOLOGICAL UNIVERSITY**  
(Formerly Delhi College of Engineering)  
Bawana Road, Delhi – 110042

**CANDIDATE'S DECLARATION**

I, Sachin Bhure, Roll No. 2K22/SWE/16, student of Master of Technology (Software Engineering), hereby declare that the Major Project-II Dissertation titled "**Network Intrusion Detection in Civil Aviation based on Improved Convolution Neural Network**" which is submitted by me to the Department of Software Engineering, Delhi Technological University, Delhi in partial fulfillment of requirement for the award of Degree of Master Of Technology (Software Engineering) is original under the supervision of Mr. Rahul and not copied from any source without proper citation. This work has not been previously formed the basis for the award of any Degree, Diploma Associateship, Fellowship or other similar title or recognition.



**Sachin Bhure**  
**2K22/SWE/16**

This is to certify that the student has incorporated all the corrections suggested by the examiners in the thesis and the statement made by the candidate is correct to the best of our knowledge.



**Signature of Supervisor**

**DEPARTMENT OF SOFTWARE ENGINEERING  
DELHI TECHNOLOGICAL UNIVERSITY  
(Formerly Delhi College of Engineering)  
Bawana Road, Delhi – 110042**

**CERTIFICATE**

I hereby certify that the Project Dissertation titled "**Network Intrusion Detection in Civil Aviation based on Improved Convolution Neural Network**" which is submitted by Sachin Bhure, (2K22/SWE/16) to the Department of Software Engineering, Delhi Technological University, Delhi in partial fulfillment of requirement for the award of the degree of Master of Technology, is a record of 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**

**Date:**



**Mr. Rahul**

**SPERVISOR**

**Assistant Professor**

**Department of Software Engineering**

**Delhi Technological University**

## ABSTRACT

The widespread adoption of cloud computing has introduced new security challenges, such as breaches within internal civil aviation networks and deviant behavior by users. This study seeks to tackle the issue of detecting unauthorized access or attacks, as well as analyzing deviant behavior within internal networks. To achieve this, we utilized machine learning algorithms from Weka software to analyze intrusion detection data sets. Our approach involved using the naive Bayesian algorithm to identify malicious behavior by users within civil aviation internal networks and classify both normal and abnormal behavior. The results demonstrated that the naive Bayesian algorithm effectively identifies abnormal behavior with high accuracy and efficiently analyzes user behavior within internal network data for cloud-based intrusion detection computing. The evolving characteristics of wireless network traffic attacks have posed challenges. Traditional intrusion detection technology has high false positive rates, low detection effectiveness, and limited generalisation ability. To increase security and identify hostile intrusions in wireless networks, we present an enhanced convolutional neural network (ICNN)-based technique.

First we characterized and preprocessed the network traffic data. We used ICNN to model network intrusion traffic data. CNN abstractly represented low-level intrusion traffic data as advanced features. It extracted sample features and optimised network parameters using stochastic gradient descent to converge the model. Simulation findings indicate that our suggested strategy outperformed standard models in terms of detection accuracy, true positive rates, and false positive rate. Convolutional neural networks can effectively extract characteristics from network intrusion detection data many existing methods based on them lack depth. When neural networks are deepened, problems like vanishing gradients can occur. To address these challenges This network intrusion detection solution combines an attention mechanism with DenseNet. This approach converts pre-processed network traffic data to grayscale maps and extracts features using DenseNet. This allows for deeper network architecture and prevents gradient disappearing. Experiments on the NSLKDD and UNSW-NB15 datasets show better accuracy and F1-score metrics compared to previous shallow models demonstrating their usefulness our approach.

**TABLE OF CONTENTS**

<b>Title</b>	<b>Page No.</b>
<b>Acknowledgement</b>	<b>ii</b>
<b>Candidate's Declaration</b>	<b>iii</b>
<b>Certificate</b>	<b>iv</b>
<b>Abstract</b>	<b>v</b>
<b>List of Figures</b>	<b>viii</b>
<b>List of Abbreviations and Nomenclature</b>	<b>ix</b>
<b>CHAPTER 1: INTRODUCTION</b>	<b>10</b>
<b>1.1 Overview</b>	<b>10</b>
<b>1.2 Motivation</b>	<b>11</b>
<b>1.3 Objective</b>	<b>11</b>
<b>1.4 Organization of Dissertation</b>	<b>12</b>
<b>1.5 Literature Review</b>	<b>13</b>
<b>CHAPTER 2: BACKGROUND</b>	<b>13</b>
<b>2.1 Overview</b>	<b>13</b>
<b>2.2 Machine Learning</b>	<b>13</b>
<b>2.3 CNN (Convolutional Neural Network)</b>	<b>14</b>
<b>2.3.1 Data Collection and preprocessing</b>	<b>14</b>
<b>2.3.2 CNN Architecture Design</b>	<b>15</b>
<b>2.3.3 Performance Optimisation</b>	<b>16</b>
<b>2.4 Proposed System</b>	<b>17</b>
<b>2.5 Related Work</b>	<b>18</b>
<b>2.6 Design Point</b>	<b>19</b>

<b>2.7 NIDs Model</b>	<b>19</b>
<b>2.8 Feature of Dataset</b>	<b>20</b>
<b>2.9 Prepare Algorithm Model</b>	<b>20</b>
<b>CHAPTER 3: Result and Analysis</b>	<b>21</b>
<b>3.1 Configuration of Experimental Environment and Setting</b>	<b>21</b>
<b>3.2 Evaluation Indicators</b>	<b>24</b>
<b>3.3 Experimental Design</b>	<b>27</b>
<b>3.4 Experimental Results</b>	<b>30</b>
<b>3.5 Evaluation</b>	<b>34</b>
<b>CHAPTER 4: Conclusion</b>	<b>36</b>
<b>4.1 Conclusion and Future Work</b>	<b>40</b>
<b>4.2 References</b>	<b>42</b>

**LIST OF FIGURES**

Figure 2. 1: Architecture of Proposed System .....	16
Figure 2. 2: ICNN Structure.....	20
Figure 2. 3: Flow Chart of Loss Function Calculation.....	22
Figure 2. 4: Feature Classification Table.....	23
Figure 2. 5: Sample Category Distribution Table.....	24
Figure 2. 6: IBWNIDM Architecture .....	25
Figure 2. 7: Object Description .....	26
Figure 2. 8: Relation of Iteration Number and ICNN Training Error.....	30
Figure 3. 1: Experimental Environment Configuration.....	33
Figure 3. 2: Train and Test Score of Algorithms .....	35
Figure 3. 3: F-1 Score of Algorithms .....	36
Figure 3. 4: Confusion Matrix.....	37
Figure 3. 5: Class Partition Graph.....	37



## **LIST OF ABBREVIATIONS AND NOMENCLATURE**

ICNN: Improved Convolutional Neural Network

KDD: Knowledge Discovery in Databases

DoS: Denial of Service

IDS: Intrusion Detection System

SDN: Software Defined Network

CNN: Computation Neural Network

ML: Machine Learning

ANN: Artificial Neural Network

PCA: Principal Component Analysis

IBWNIDM: ICNN Based Wireless Network Intrusion Detection Mode

RELU: Rectified Linear Unit

NSL-KDD: Network Security Laboratory Knowledge Discovery in Databases

## CHAPTER - 1

### INTRODUCTION

#### 1.1 Overview

Any detected suspicious activity is alerted at the administrator's system that monitors network traffic. Infiltrations of data may be detected by intrusion detection systems. It is a program that detects a dangerous or policy violation activity on a network or machine. Detection of information on malicious activity or policy breach occurs either at an administrator's level or with the use of SIEM systems. The SIEM detects entries into malicious and incorrect logs since there is relaying of different data sources alert filtering. A DoS attack intends to create a computers networks operation so inoperative that it can no longer be used by the users for whom it was designed. The information-as-a-result objective of the DoS attack is typically accomplished through flooding the target with traffic or crashing it. Probing attacks are invasive, which means that the physical silicon structure of the chip is explored for circumventing security measures.

In an intrusive attack, one directly accesses the critical information by breaking open a targeted device's internal cabling and connections. This article describes the limitations of the strong attachment and generalizing ability in the existing work. Efficiency and detection accuracy of this algorithm helps a new enhancement. In response to over-fitting the network in training and improving its generalization ability, and from characteristics structural of the convolutional neural network and designed ideas of cross layer aggregation this research paper present intrusion detection model based on deep learning. The substantial difference is that internal network assault occurs in civil aviation, which, however, does not vary far from network attack externally.

They exploit the inbuilt virtualization They then use normal ways to acquire the functionality of cloud computing; for instance, any legitimate user expression can be used to try and approve users. Legal exploitation of established authentication and authorization processes After the successful attainment of an authenticated proper process of authorization, the attackers can now access the internal network resources in harmful and illegal ways to reach the cloud computing system. Usually, these attacks are very hard to detect with the traditional means; these are complicated to detect. To the problem above, this paper proposes a method of applying the machine learning algorithms to identify and extract valuable information from audit behavior sequence dataset in the user internal network behavior. The model can implement the abnormal behaviors model for internal network personnel under the cloud computing model and find the most important patterns and correlation of attack behaviors. Essentially, this works in a practical way to conduct deeper research on abnormal activities within the internal networks of civil aviation operating under the cloud computing model.

## 1.2 Motivation

The major reason of implement this project was that we want to make As wireless network attacks become more sophisticated, flexible, stable, and effective intrusion detection solutions are necessary. While modern wireless network authentication and firewall technologies can handle basic security needs, their protective capabilities are limited. These protective measures are almost rendered ineffective once encountered by the malicious attacks of professional hackers. Misuse detection, representing a common intrusion detection method, is known for its shortcomings in terms of low detection accuracy and feature extraction efficiency, and it results in a high false positive rate. Recently intelligence based artificial detection methods become hot topic with IDS research with the application of artificial intelligence methods on intrusion detection systems.

## 1.3 Objectives

To realize and implement a strong Network Intrusion Detection System by using improved Convolutional Neural in Network for securing civil aviation networks in a manner that ensures operation is safe, reliable, and integral. Basically, this Networking Intrusion Detection System base on improv Convolutional Neural in Network will enhance cyber security defenses for civil aviation networks. It is designed to meet goals in terms of accuracy, adaptiveness, resource efficiency, threat coverage, regulatory adherence, usability, and continuous improvement to build a reliable and future-proof security solution for the aviation industry.

### 1.3.1 Detection Accuracy and Speed Improved:

- Objective: Design an ICNN architecture that can detect all kinds of cybersecurity threats on aviation networks, from known to unknown attacks, with high accuracy and swiftness.
- Result: High detection accuracy and real-time processing ability will be achieved to timely identify and mitigate security incidents.

### 1.3.2 Adaptability to Aviation

- Objective: Enable ICNN-based NIDS to be adapted, considering that unique characteristics are present and limitations are placed on aviation networks, onboard systems, ground control, and passenger service networks.
- Outcome: A flexible detection system designed for effectively functioning in network segments and deployed with common configuration and support to civil aviation.

### 1.3.3 Minimize False Positives and Negatives:

- Objective: The implementation of advanced data preprocessing and feature extraction techniques has been done within the scope of reducing the number of false positives (benign activity being marked as a threat) and false negatives (associated threats going undetected).

- Outcome: To improve the reliability and trustworthiness of NIDS for operational efficiency and security enhancement.

#### **1.3.4 Resource Efficiency and Scalability:**

- Objective: Develop the ICNN model such that it is optimized for computational resources to run on minuscule hardware, which is actually common in aviation systems, without performance degradation.
- Outcome: Develop a solution that is scalable across diverse platforms and scales over a range in correspondence with the augmentation of network infrastructure.

#### **1.3.5 Comprehensive Threat Coverage:**

- Objective: Generalize the scope of NIDS to cover any type of malware, DoS attack, data breach etc., by training ICNN using large and varied datasets.
- Outcome: Provide wide protection against far range in CYBER security threats to raise the security posture of aviation networks.

### **1.4 Organizational of Dissertation**

The chapter starts with introduction, setting the background and motivation for this study on how important network security is within the civil aviation sector among extant network intrusion detection systems (NIDS). Section 2 of the literature reviews chapter begins with an introduction detailing its structure. This chapter examines the existing practices and standards for aviation network security and presents case studies of past network attacks documented in the field. Chapter 3: Methodology The chapter on methodology first introduces the research strategy and design that justifies the CNN adopted for NIDS in civil aviation. It describes sources related to data collection methods which are aviation network traffic data, simulated attacks, preprocessing and labelling processes. Chapter 4: Results and Discussion The chapter start with outline of the chapter followed by an presentation of results had been obtained through a few experiments. The comparisons mentioned were between proposed system and already existing methods. The whole dissertation has been bifurcated into different chapters to highlight each of the aspects separately. The chapter contains a short brief of the structure and is followed by the representation of experimental results taken from several tests. A few comparisons between the results of the existing methods with that of the proposed system are also included. References and Appendices At the end of the dissertation, there is an inclusive list of references made to the text. Appendices are expected to contain supporting material which, though not a part of the main text, supports it, such as data sets, code fragments, and detailed experimental results.

### **1.5 Literature Review**

Lately, Software-defined Networking (SDN) has presented phenomenal promise as the successor to the Internet. SDN allows networks to be developed, integrated, managed, and adapted [10]. Apart from this, there also exist certain risks associated

with the environment, like network crashes, system failure, fraud in online e-banking, and theft. Hence, it is essential to be high performing, reliable, and have a robust infrastructure, in order to mitigate the impacts of these issues that may hurt families, businesses, and economy. In recent years, SDN has greatly evolved NIDS through the use of advanced AI algorithms. In fact, new improvement of the algorithms with the use of already acquired data and enhancement in the process of data analysis, besides other improvements, present a possibility for developing new, much better systems; more reliable and resilient; for the identification of different network attacks. The reasoning for reviewing NIDS on SDN grounds is to discern just how relevant networking intrusion detection systems are, with NIDS being this powerful securities ensuring tool by administrators of the underneath network infrastructure. NIDS captures and analyzes coming in and outgoing communication of family network devices and alerts of intrusion. An access control, these NIDSs come in different types such as SNIDS and ANIDS.

It is on the point of becoming very critical that as technology advances and networks are increasingly interconnected globally, research and development in network security become critical. The hottest issue among the worldwide security policy is the fact that a firewall may not defend all types of attack. While firewalls can allow or deny throughout the network, they are not immune to detection or attack. Implementing an access detection firewall is an additional solution for detecting network intrusions. Firewalls and intrusion detection systems (IDS) address different aspects of information technology security.

A CNN is particularly successful in extracting characteristics from network intrusion detection data. Currently, most intrusion detection algorithms based on CNNs lack deep layers, leading to difficulties such as disappearing gradients. To address these issues, we present a network intrusion detection approach that combines attention with DenseNet. This approach converts pre-processed network traffic data to grey scale maps and extracts features using DenseNet, allowing for deeper network analysis without gradient disappearing. Experiments were done utilising the NSL-KDD and UNSW-NB15 datasets to identify significant characteristics and filter out extraneous information in network traffic data. The experiment findings imply that The model exhibits higher accuracy and F1-Score metrics compared to other shallow models. The model exhibits higher accuracy and F1-Score metrics compared to other shallow models. hence proving superior performance compared to other techniques considered

## **CHAPTER - 2**

### **BACKGROUND**

#### **2.1 Overview**

When an incident is suspiciously identified, network traffic monitoring system can send alerts to the administrators. Sometimes IDS are used to recognize the presence of data breaches. This specific programme searches a network or machine for unsafe behaviors or policy violations. As a result, either the IAM administrator or SIEM system can be utilized in a bid to gather data about malicious activities and policy violations[1]. The SIEM employs alert filtering to identify dangerous and false entries from disparate sources[2]. DoS attack refers to an assault that aims at halting computers' functioning thereby making it impossible for any person intended for its use to access it. This goal is achieved by drowning target in information delivered through flooding traffic, or existence of such crashes. Probing attacks involve intruding into protected systems by examining physical silicon implementation of a chip's security features. Obtaining direct access to internal cables and connections within the targeted device during an intrusive attack may yield valuable points. Internal network attack in civil aviation is different from external network attack because they are firstly based on virtualization In order for one to have cloud computing characteristics, he/she might have ordinary ways of having genuine user identities. After successful legal authentication and authorization procedures

#### **2.2 Machine learning**

Presently, the age of machines has made artificial intelligence an important element in this era. Machine learning on other hand involves teaching a computer to do various tasks using either supervised or unsupervised methods depending on the dataset at hand. In image processing, machine learning is important as it can facilitate improvement of resolution on new images by training the machine to identify links between high and low-resolution images. However, it should be noted that clarity and quality of the ultimate high resolution image will be determined by the specific picture super-resolution technology used. With many other machine learnings techniques instead Artificial Neural Network(ANN) are now wide used and they have gained popularity over time. Also ANNs are among numerous machine learning approaches that are extensively applied because they always surpass all other models as well as give sound results across different problem domains.

#### **2.3 CNN (Convolutional Neural Network)**

Deep learning model called Convolutional Neural Network(CNN) are often utilized to analyze visual data. They are very good at this as they can identify certain patterns

of a given information, so that they are used in object identification, image classification and network intrusion detection for instance. Civil aviation networks are fundamental to the functioning of the aviation sector safely and with efficiency. These include air traffic control, onboard avionics, ground communication systems and passenger services among others. This is why a strong need for more robust cybersecurity measures to safeguard the industry has grown since it has gone digitalized day by day. My paper suggests the development a Network in Intrusion Detection System (NIDS) using Convolutional Neural Networks (CNNs) designed specifically for civil aviation industry.

The typical layers in a Convolutional Neural Network (CNN) include:

- Convolutional Layer
- Pooling
- Activation
- Dropout
- Batch Normalized Layer
- Fully Connect

Each of these layers has a different purpose in architecture of CNN.

Key Specification of Proposed System

### **2.3.1 Data Collection and Preprocessing:**

- Objective: Collect network traffic data from multiple aircraft segments, including both routine and hostile activity.
- Preprocessing: Format network traffic data for CNN processing, such as encoding packet characteristics into image-like representations or structured matrices.

### **2.3.2 CNN Architecture Design:**

- Convolutional Layers: Use convolution methods that extract features from preprocessed data.
- Pooling Layers: Reduce the dimensionality of feature maps while maintaining critical information, which aids in controlling overfitting and lowering processing needs.
- Fully Connected Layers: Combine the information acquired by convolutional layers to determine if network traffic is legitimate or malicious.
- Activation Function: Use nonlinear activation function(ReLU) bring nonlinearity into model allowing it to learn intricate patterns.
- Output of Layer: Use a soft max or sigmoid activate functions in output layer to provide probability ratings for each class (normal or other forms of assaults).

### **2.3.3 Model Training and Validation:**

- **Training:** Run the CNN on labelled network traffic data, adjusting the model parameters to reduce classification error.
- **Validation:** Test the model performance on different validation datasets to confirm that it can generalise effectively to new data.

#### **2.3.4 Deployment and Real-Time Detection:**

- **Integration:** Integrate the learned CNN model into the aircraft network architecture, allowing for real-time intrusion detection.
- **Monitoring:** Continuously monitor network traffic and feed it into the CNN for real-time analysis and threat identification.

#### **2.3.5 Performance Optimization:**

- **Resource Efficiency:** Optimise the CNN model to function efficiently within the resource restrictions common to aviation networks, such as restricted processing power and bandwidth.
- **Latency Reduction:** Implement ways to reduce detection latency, resulting in a quick reaction to recognised threats.

## **2.4 Proposed System**

IDS, or intrusion detection systems, are designed to detect suspicious activity of network traffic and alert users to potential threats. Programs are typically used to scan networks and systems for harmful activities. To prevent DoS attacks at the network infrastructure level, firewall rules and IDS implementations are commonly utilized. Once an attack is detected, IDSs can block traffic from suspicious sources. This approach is simple and effective, and cyber attackers often struggle to bypass it. Collaborative intrusion detection systems are now facing new challenges.

### **Dataset**

Retaining a record of prior events and scrutinizing the data can lead to the identification of recurring patterns.

### **KDD data:**

The KDD dataset is held in high regard within the realm of Intrusion Detection methodologies.

### **Data cleaning:**

Data cleaning is a crucial aspect of every machine learning project and involves the process of preparing data for analysis. This module encompasses data cleaning, wherein erroneous or missing data, replicated entries and improperly structured data that has been properly structured can be removed or altered to ensure its suitability for analysis. The effectiveness of a neural network is mostly dependent on its depth,



which establishes the networks abilities to extract deeper feature and improve the accuracy of feature expression. However, vanishing gradients are an issue for deep neural networks. This research uses the DenseNet network as its network infrastructure because of its ability to partially tackle the vanishing gradients problem due to its compact parameter set, overfitting resistance, and feature reuse. There will be a lot of incorrect features once network traffic feature are transformed to grayscale image. In order to address this issue, the DenseNet network adds the ECANet attention mechanism, which enhances the visibility of significant network traffic aspects reduce erroneous characteristics and raise the rate of detection. Swish activation functions is then added to DenseNet to further strengthen Densenet and increase the models capacity to extract features since it works better in the deep model.

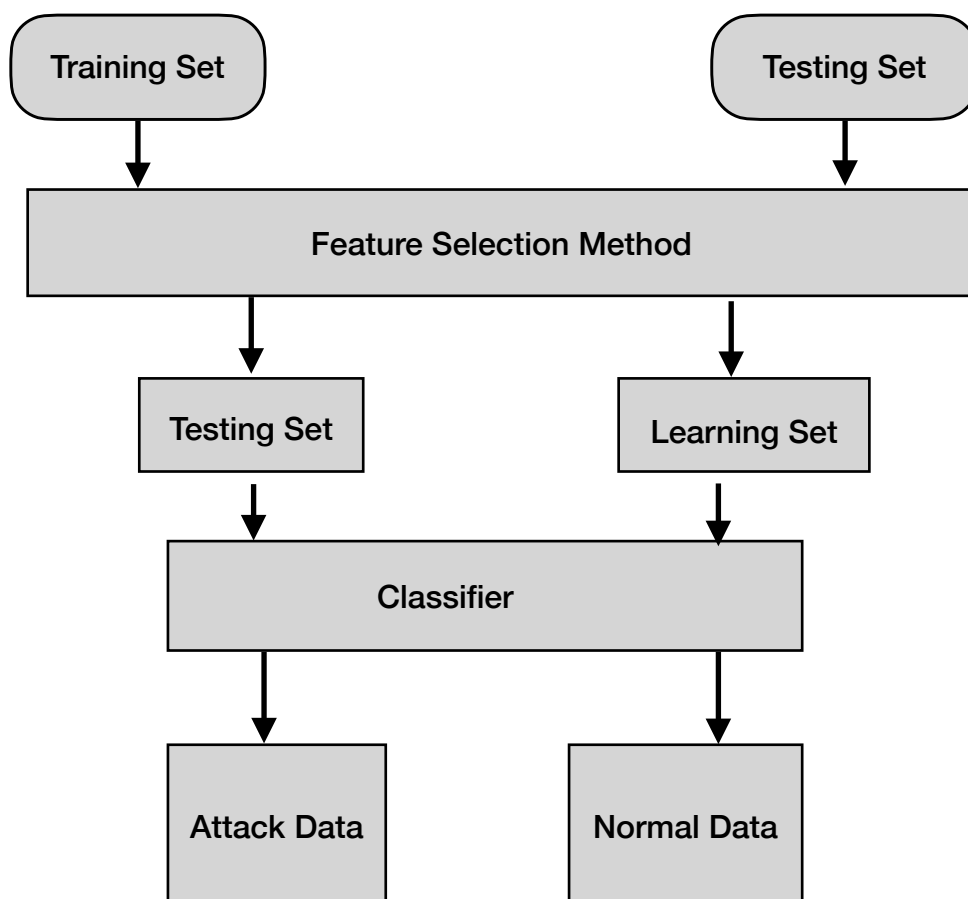


Figure 2.1: Architecture of Proposed System

Reducing the amount of characteristics in the dataset is one way to reap the benefits of faster training and better accuracy. This is achieved as part of the data cleansing process by removing or editing redundant, erroneous, or incomplete data. The ML algorithm is taught using a training model that contains both input and output data after the dataset has been cleaned. For every piece of input data, the algorithm uses

the output data to inform its predictions or categorization. Model for testing: In this specific module, the trained machine learning model's performance is assessed by applying it to the test dataset. Figure 1 depicts the flow of the DenseNet, Swish, and ECANet network intrusion detection technique. It primarily includes:

1) Data preprocess module:

The data preparation module consists of data numericalization and normalisation. Refer to Section IV.B for details on the processing process. Preprocess is used to transform original networking traffic data to a format that neural networks can handle.

2) Features compressional module:

The feature compression module compresses data feature after preprocessing. Data is compress using two completely linked layer of 64 neuron. This make the data feature more obvious and simple to classify which has proven useful in the work of literature [12].

## 2.5 Related Work

**2.5.1 Genetic Algorithm:** Genetic algorithms are grounded in natural selection and genetics principles. To begin with, GA generates an initial group of individuals with each individual representing a conceivable solution of the problems. Every individual has a unique set of chromosomes, which comprise a fixed number of genes. A fitness function measures the effectiveness of an individual's adaptation to the environment, represented numerically as a rule. At the start, a population of random individuals is generated, which evolves over time, gradually improving the quality of individuals based on their fitness values. In every generation, three fundamental genetic operators, namely selection, crossover, and mutation, are utilized.

### 2.5.2 Decision Tree Algorithm

Supervised learning techniques often involve the use of Decision Trees depict various features of a dataset, with rules serving as branches and outcomes as components Decision tree where Leaves notes create decisions based on the outcome as the decision note without any branches. Decisions or tests are performed based on the dataset's features, and the decision trees algorithm is use to classify intruder dataset and abnormal dataset before train data and computational.

The behavior analysis of internal network users in research primarily relies on host user analysis, which utilizes datasets such as RUU, PU from Purdue University, and Greenberg. However, these datasets do not encompass network action parameteric that reflection the character of cloud variable computing asof these proportion of users with specific IP addresses, network services, and protocol names. In order to develop an intrusion detection database for cloud computing systems, it is necessary to gather user action audited dataset based on knowledge as well as Dataset CIDD dataset specifically designed for cloud computing systems, consisting of 128 normal

users and capable of listening and auditing Linux and Windows host systems as well as TCP network sessions. The dataset includes real stealthy attacks and is suitable for cloud computing application environments, including virtual machine sessions.

### 2.5.3 K Nearest Neighbor (KNN) Algorithm

Network intrusion detection in civil aviation is a critical area of research and development, aiming to protect sensitive information and ensure the safety of flight operations. One promising approach in this domain is the application of machine learning technique as the K Nearest Neighbor (KNN) algorithm. Here is a detailed overview of how K-NN can be applied to network intrusion detection in civil aviation:

#### Detection Process

- **Real-time Monitoring:** Continuously monitor network traffic and extract features in real-time.
- **Classification:** For each newly data point calculate its distances to all points in training set and determine majority class among the k nearest neighbors.
- **Anomaly Detection:** Classify data points that deviate significantly from the normal pattern as potential intrusions.

#### Advantages of K-NN for NIDS in Civil Aviation

- **Simplicity:** KNN is easy to implement and understand making it suitable for real-time applications.
- **Adaptability:** It can adapt to new types of network traffic and intrusions by updating the training dataset.
- **No Training Phases:** Unlike other algorithms KNN does not require an extensive training phase allowing for quick deployment.

#### Challenges and Considerations

- **Scalability:** KNN can be computationally intensive for large datasets as it requires calculating distance to all points in the training set.
- **High Dimensionality:** Performance can degrade with high-dimensional data, necessitating dimensionality reduction techniques like Principal Component Analysis (PCA).
- **Imbalanced Data:** The algorithm might struggle with imbalanced datasets, where normal traffic far outnumbers malicious traffic. Techniques like Synthetic Minority Oversampling Techniques (SMOTE) can help address this issue.

### 2.5.4 Logistic Regression Model

Logistic regression is a supervised machine learning technique used to perform binary classification problems. It represents the chance that a given input belongs to a specific class. The technique utilizes the logistic function to map predicted values to probabilities, which are then used to classify the input as either normal or malicious.

#### Detection Process

- **Monitoring in Real Time:** Monitors network traffic at all times, extracting features from it in real time.
- **Probability Calculation:** For each new data point, the logistic regression model will calculate the probability that the traffic is coming from a bot.
- **Threshold Setting:** A probability threshold is defined beyond which network traffic can be concluded to have been classified as intrusion. This threshold can always be adjusted based on the desired sensitivity and specificity.
- **Generation of an Alert** Generate an alert for the network administrator should the probability of intrusion cross the threshold.

#### Advantages of Using Logistic Regression with NIDS in Civil Aviation

- **Simpleness and interpretability:** Logistic regression is easy to conduct and understand; hence, the results are easily interpreted. It is, therefore, very useful in critical implementations, such as aviation.
- **Efficiency:** The algorithm is good in performance computationally and can handle vast data volumes of networking traffic in real-time.
- **Scalability:** Logistic regression is easily scalable in terms of data size and, hence, can be applied to civil aviation network environments.

#### Challenges and Concerns

- **Assumption of Linearity:** Logistic regression assumes that there is a linearity relation between the inputs and log-odds of outcome, and this does not generally hold for complex network traffic data.
- **Feature Engineering:** Performance of a logistic regression model would heavily depend on the correct and efficient feature selection and extraction.
- **Class Imbalance:** In intrusion detection, the number of normal traffic instances is substantially larger in comparison with malicious ones. Some techniques that can help address this problem are oversampling the minority class or using different thresholds.
- **Evolving Threats:** Since the cyber threats are continuously changing, the model requires updates at regular intervals and retraining with new data to maintain its effectiveness.

### 2.5.5 Random Forest

Random Forest is an ensemble learning method that constructs a multitude of decision trees at training time and outputs the mode of the classes in classification problems. This has proven to work effectively in increasing the generalization ability of models, thus minimizing one's data overfitting danger.

#### Detection Process

- **Real-time monitoring:** Observe the network continuously for traffic by extracting feature functionalities as on-the-fly.
- It makes the prediction whether the new data point belongs to normal or malicious traffic through majority voting by the decision trees.

- Anomaly detection: Classify data points as a potential intrusion if considered to be malicious by the model.

Advantages of using Random Forest in Application for Civil Aviation's NIDS

- More Accuracy: Random Forest models happen to be high-accuracy models in classification tasks, which finds very many applications.
- Robustness: The ensemble design of Random Forest makes it robust to overfitting for the handling of diverse and complex network traffic data.
- Interpretability: It supports the generation of importance scores for features, i.e., the outcome of such models in defining which features contribute more to
- Scalability: Random Forest is capable of processing very large datasets and can be scaled to meet the requirements of different aviation network environments.

Challenges and Issues

- Computational Resources: The training and deploying of Random Forest models may become very heavy computationally, especially for the largest data sets that entail high numbers of trees.
- Feature Engineering: Good selection and engineering features into any model is what gives rise to a good performance. Poor features choice might be related to low performances.
- Class Imbalance: Intrusion in detection dataset often suffering from class imbalance where normal traffic instances significant out number maliciously ones. Techniques unlike oversampling undersamply or using weighted loss function can help address this issue.
- Evolving Threats: Cyber threats continuously evolve, necessitating regular updates to the model and retraining with new data to maintain effectiveness.

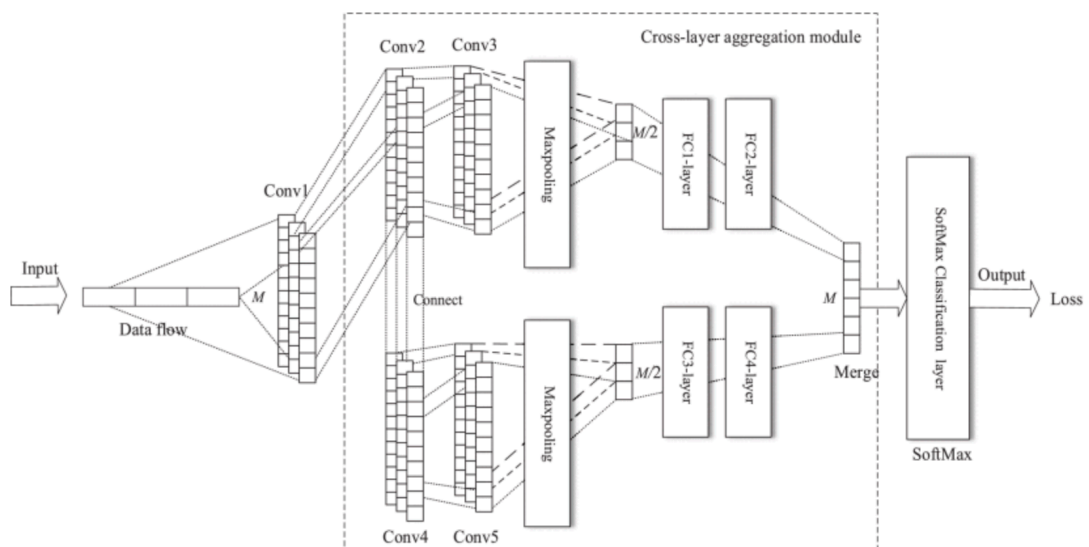


Figure 2.2: ICNN Structure

## 2.6 Design point

The traditionally classify detect also requires a considerable no of manual settings for crucial parameters during the training process. This can lead to a loss of keys features informations and difficulty in parameters tuning. This study proposes the use of an endtoend semisupervised network train classifier based on Convolutionals Neural Networks (CNNs) to overcome these limitations. By utilizing the multi-layer features of CNNs, the proposed ICNN is able to learn sampling data featured and identify data rule during the train preprocess. This simplifies the implement preprocess and offers an improved approach for network detection.

1)The intrusion detection model is designed using the cross-layer aggregation method, which begins

from seconds convolution operations and stores convolved outcome. This approach enables the execution of convolution pooling and fully connection operation separately.

2) Once the outputs of the 3rd convolatory operating undergoes the same process, the tensored flow concat() function are employed into merge the output dataset value to measure steps.

3) Then SoftMax classification result are used into compute the loss value, which is then utilized for backpropagation. The network weights and biases undergo iterative training to obtain a favorable convergence effect.

### 2.6.1 Advanced Convolutory NN Structures

These architecture of the Advanced Convolutory NN Structures CNN is illustrated CNN comprises 4 convolutional steps. The relu activation function is applied to Conv1, Conv2, and Conv3 to enhance network sparsity. To prevent overfitting during training, the dropout regularity function is implemented into the 2 full complete connection layered of the crosslayered aggregation module, namely FC1\_layer and FC3\_layer.

### 2.6.2 Module Train dataset CNN

— Propagation Forward

These train process of ICNN in the forward propagation phase follows a specific structure. Firstly batch pass training is implemented within networks where each train random select a fixed size blocked from processe training dataset as input. Then dimensions of the data parametersduring training are set as four-dimensional parameters: (batch\_size, H, W, channel). During each raining sessions a block of size N is randomly select through dataset, with the input data set's heighted and wide set to 2 and 123 respective and the channels is set to a unit value

$$y_j^l = \sigma \left( \sum_{i \in M_j} y_i^{l-1} w_{ij}^l + b_i^l \right) \quad (1)$$

$$Relu(y) = \begin{cases} y & (y > 0) \\ 0 & (y \leq 0) \end{cases} \quad (2)$$

Equation 2.1: Relu Function

The formula for calculate the output results of ICNNs convolutional layer is expressed a follow:  $y_j^l$  denotes the result obtained by processing  $j$  convolution kernels with  $l$  convolutional layers, where  $\sigma$  represents the activating functionality  $x$  denotes the weighted and  $c$  are these unbias. Afterward, then cross-layer aggregation module processes the output results of Conv2 and Conv3. In Conv4, each feature map outputted by a convolution kernel is downsampled by the Max\_pooling1 pooling layer to reduce dimensionality and extract the maximum regional features of the data. The output results are then inputted to the FC1layer and FC2layer, resulting in an  $M/2$ - dimensional dataset. The pooling layer is calculated using the following formula: Backtrack Properties

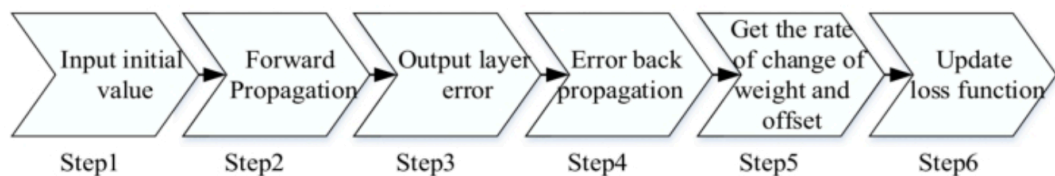


Figure 2.3: Flow Chart of Loss Function

The back propagation processed in ICNN are arch follow The Soft max layers is used to classifies the samples in the train set and overall error parameter value loss is calculated. Then, based on this loss value, back properties are performed to check the measure and biases of every type Networking set iteratively. Aim of these back propagation process is to minimize these duplicate between the actual and then output expected by through out put updating the networking parameters. This process continues unless a good convergences effects are achieve. The objective of back propagation in ICNN is to achieve convergence by iterative adjusting weight and bias these expected valued. To facilitate the search for enable the network outputs

function  $x$  to fit all are defined is employed into minimum these losses  $fun(y)$  and accelerate the optimisation process. The lossy program is expressed as it. Due to the availability of open datasets, the academic community in network security has initiated research and analysis related to them. For instance, in the authors analyze security strategies for storage managements and application of big data in smart cities. They introduce a multi-label-based securities access controls model for big dataset which enables data owners to assign security labels based on their. In which involves constructing a multidimensional analysis users behavioral and features descriptions to facilitate comprehensive analysis and modelling of user behaviors.

No.	Features	Types	No.	Features	Types
1	duration	continuous	22	is_guest_login	discrete
2	protocol_type	symbolic	23	count	continuous
3	service	symbolic	24	srv_count	continuous
4	flag	symbolic	25	serror_rate	continuous
5	src_bytes	continuous	26	srv_serror_rate	continuous
6	dst_bytes	continuous	27	rerror_rate	continuous
7	land	discrete	28	srv_rerror_rate	continuous
8	wrong_fragment	continuous	29	same_srv_rate	continuous
9	urgent	continuous	30	diff_srv_rate	continuous
10	hot	continuous	31	srv_diff_host_rate	continuous
11	num_failed_logins	continuous	32	dst_host_count	continuous
12	root_shell	discrete	33	dst_host_srv_count	continuous



13	num_compromised	continuous	34	dst_host_same_srv_rate	continuous
14	root_shell	discrete	35	dst_host_diff_srv_rate	continuous
15	su_attempted	discrete	36	dst_host_same_src_port_rate	continuous
16	num_root	continuous	37	dst_host_srv_diff_port_rate	continuous
17	num_file_creations	continuous	38	dst_host_serror_rate	continuous
18	num_shells	continuous	39	dst_host_srv_serror_rate	continuous
19	num_access_files	continuous	40	dst_host_rerror_rate	continuous
20	num_outbound_cmds	continuous	41	dst_host_srv_rerror_rate	continuous
21	is_host_login	discrete			

Figure 2.4: Feature classification table

## 2.7 NIDs Model

### 2.7.1 NIDs design

The report presents the CNN Base Wireless NID system Models IBW which are developed using the ICNN framework. IBWNIDM consists of three main components, namely, the datasets processing model CNN module train data set and classify detect models. The design of IBW are illustrated in as follows

### 2.7.2 NIDS Model design

1. Dataset processing the modules encompasses 2 operational task namely numeric process and normaliz with then objective is presenting a uniform in datasets of networking train. These processing sequence into modules are structured as. Initially One-hot technique is employed to process these continuity and discretisation symbol date present into datasets numeric value. Next, the numerical data obtained from this process is subjected to linear mapping, which standardizes the input data set for the network training purpose.
2. The CNN train modules involves 2 key process the process of extracting features through forward propagation and iteratively adjusting weights using back propagation. optimization. During the process preprocess train datasets are input asit the initial dataset and CNN autonomously extracts features from the data. In the back

Category	KDDTrain	KDDTest+	KDDTest <sup>-21</sup>
Probe	45927	2421	4342
DOS	11656	7458	2402
R2L	995	2754	2754
U2R	52	200	200
Normal	67343	9711	2152
Total	125973	22544	11850

Figure 2.5: Sample Category Distribution Table

propagation iterative optimization process, the error is propagated back according to these losses valued obtained during CNN train and then parameters are optimized and replicated until these models achieve a desirable level of convergence.

3. The classification detection module trains classifiers to classify the data into five categories: Probe, DOS, UR, RL, and abnormal. Then preprocessed testing datasets are fed into these training classifiers as input for testing. Then classification performance is detected on the test sample set and produces a confusion matrix with five dimensions, which are considered the detection outcome.

## 2.8 Features Of Dataset

### 2.8.1 Data Selection and Extraction

The utilized the KDD dataset as the primary dataset for model development and evaluation. Compared to the KDD dataset, the KDD dataset has significantly reduced the amount of redundant data and has a more balanced and reasonable sample data distribution, which makes it more suitable for verifying the experimental requirements of this study's test model. The sample data of both datasets have similar characteristics, with each intrusion record containing 43 dimension features that is broken down into 37 dimension numeric features, 3 dimension symbolic features, and labels. The NSLKDD dataset consists mainly of normal data (Normals) and four major attack type data (Probe, DOSs, u2r, r2l), which can be further subdivided into 39 subclasses. NSLKDD dataset includes three sub-datasets: the training set (KDD Train), the test set (KDD Test) and the test set (KDDTest21). The sample categories, distribution, and feature classifications can be seen for respective.

$$AC = \frac{TP + TN}{TP + TN + FP + FN}$$

Equation 2.2 : Accuracy Rate

### 2.8.2 Processing Dataset

The initial step of data preprocessing comprises two stages: numeric process and normaliz process.

1. The numeric process stage is crucial in preparing testing dataset for ICNN as the input for ICNN must be in the form of a digital matrix. To address the presence of symbol feature in test dataset a onehot encod methods use into mapping dataset into a digital feature vector. This process is designed to address the following three characteristics:

- The protocol of type features comprises with three attribute types: TCP, UDPs and ICMPs which are encod binary vector (0, 1, 1), (1,0, 1), and (1, 1, 0) respectively.
- The 72 symbol attributes within services type of features are transformed into a 74dimensional binary features vectors through process for encode.
- flag type feature contains 11 types of symbol attributes, which are converted into an 11 dimension binary2 features vectors through encode. Through numeric process the aforementioned 3 type of symbolic feature are transformed into 85 dimension 2binary in features vector. Combined withinthen 37 dimension digitals feature in dataset, each records into dataset have 41 dimension feature which is ultimately transformed in of to 123 dimension binarys features vectors.

2. Dataset contains continuou features dataset with significantly different value ranges. For instance, the num\_root feature has a valued rangeof [1, 7468], while the off num shells features has a valued rangeof [1, 5]. This indicates a large difference between the min and max value of two features. To enable efficient arithmetic process and eliminate dimension, a normalization method is employed, which uniformly linearly maps the value range of each feature in the interval [0, 1]. The normalization formula is as follows: where Xmax and Xmin denote then min and max feature measure value, respectively.

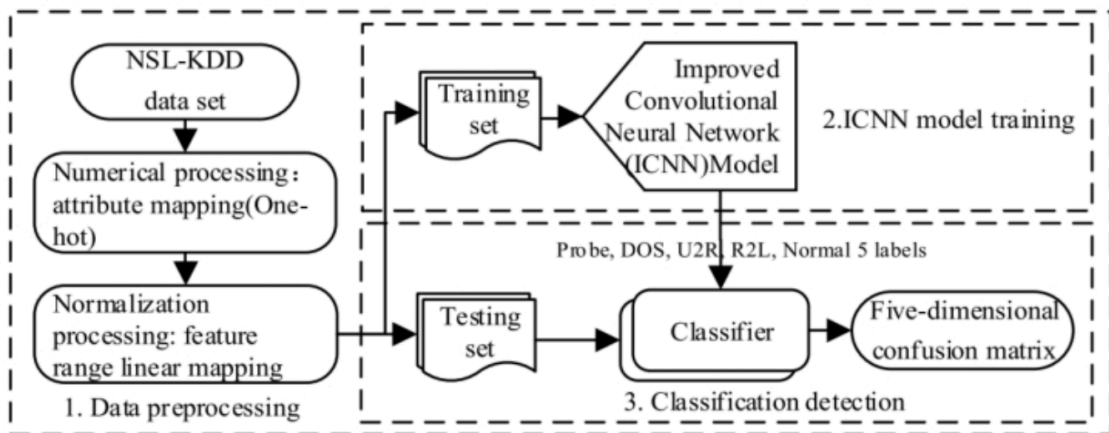


Figure 2.6 : IBWNIDM Architecture

### 2.8.3 Wireless Networking Intrusion Detection Model Architecture

#### 1. Dataset Preprocessing

The module performs two operation: numerical process and normalisation. The module objective is to offer a standardised set of input data for networks training. The modules processing design is as follow First the continuous and discrete symbolic data in the dataset is numerically process using the Onehot approach. The numerical data is then linear translated to the features range yielding a standardised networks input dataset.

#### 2. ICNN Training model

The module has two processe: forward propagation features extraction and reverse propagation iterative optimisation.

Forward propagation feature extraction: The pre-processed training data set is sent as Input Data, and the feature extraction is handled by ICNN's autonomous learning capacity.

Backpropagation Iterative Optimisation: The error back propagation procedure is carried out in accordance with the loss value received from ICNN training, and the parameters are continually optimised until the model achieves a satisfactory convergence effect.

#### 3. Classification Detection

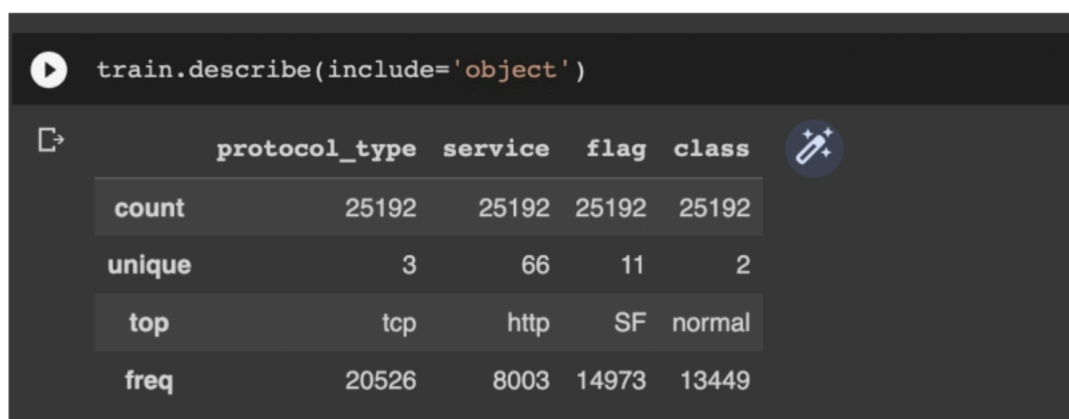
Classifiers for classification tags are trained using five sample categories: Probe, DOS, U2R, R2L, and Normal, with the pre-processed test data set serving as test data. The classifier performs classification detection on the identified samples and returns a five-dimensional confusion matrix as the detection result.

## 2.9 Data Features Analyse

### 5.2.1 Dataset Selections and Characterize

This work used the NSLKDD CUP dataset[17] as the baseline for model training and performance assessment. Compared to the KDD dataset the NSLKDD CUP data set removes a large amount of redundant data, making the proportion distribution of the sample data more balanced, reasonable and usable allowing it to better meet the verification experiment requirements of the test model in this study. The sample data characteristics of the above two data sets are identical, and each intrusion record in the data set comprises 42-dimensional features, which are further divided into 38-dimensional digital features, 3-dimensional symbol features and one assault type label. The data types in the NSLKDD CUP dataset primarily include Normal data (Normal) and four big assault types data (Probe, DOS, U2R, and r2l). The data for the four assault categories may be separated into 39 subclasses.

The NSLKDD data collection consists of three subdatasets KDDTrain, KDDTest+, and KDD Tables 1 and 2 demonstrate the sample category distribution and feature categorization, respectively.



	protocol_type	service	flag	class
count	25192	25192	25192	25192
unique	3	66	11	2
top	tcp	http	SF	normal
freq	20526	8003	14973	13449

Figure 2.7 :Object Description

Data preprocess include two steps: numerical processing and normalis process.

#### 1. Numerical of Process

Because the ICNN input is a digital matrix, the test data set's symbolic features are converted to a digital feature vector via a one-hot encoding method. This approach is primarily designed for the following three characteristics:

TCP, UDP, and ICMP are all represented as binary vectors (1, 0, 0), (0, 1, 0), and (0, 0, 1).

Encoding the 70 symbol properties in the service type feature yields a 70-dimensional binary feature vector.

The flag type feature's 11 discrete symbol attributes are encoded into an 11-dimensional binary feature vector.

Following numerical processing, the aforementioned three types of symbolic characteristics produce 84-dimensional binary feature vectors, and when paired with the data set's 38-dimensional digital features, the 42-dimensional characteristics of each record Create dimensional binary feature vectors.

## 2. Normalized Processing

The dataset the range of values for continuous feature data varies greatly. For example, the value range of the num\_root type feature but the value range of the numshells type feature is [0, 5]. It is clear that the range of the least and greatest values of the two is vastly different. To simplify arithmetic processing and eliminate dimension, a normalised processing approach is used, and each feature's range of values is uniformly linearly mapped in the interval [0, 1]. The normalised formula is:

$$X_{norm} = \frac{X - X_{min}}{X_{max} - X_{min}}$$

Equation 2.3: Normalisation

where Xmax and Xmin represents maximums and minimums values of features.

## 2.9 Prepare Algorithm Model

Deep learning is widely used in several disciplines, including image processing, data processing, and network security. Deep learning and machine learning vary from the following points: Deep learning can automatically learn and detect various data characteristics. Deep learning takes longer to train than machine learning, but it has clear advantages in terms of faster network threat detection and greater classification accuracy during testing. The technique improves model prediction and classification accuracy by deepening and strengthening feature learning at each layer.

Tuning network parameters such that the trained model may simplify the data to be processed from complicated, resulting in a relatively simple input-output connection. To address the issue of poor identification of tiny proportions of attack data in the network attack data set, in the experimental section, the hybrid CNN AdaBoost algorithm and sparse encoder method are employed to test the network attack data set. The three operational phases are data preparation, feature extraction, and data categorization.

Network intrusion data covers worm attacks, Trojan horse attacks, access denial attacks, remote local attacks, and user root attacks. There are too many redundant and noisy items in KDD99, NSLKDD99, UNSWNB15, and other authoritative original intrusion network datasets, increasing data training time and affecting classification

precision. Therefore it is highly crucial to preprocess the network attack data before the experiment.

Network attack data sets typically comprise two sorts of data: character and numerical types. Character data consists of protocol type, flag, and service attributes. During the data preprocessing stage, character data is often converted into numerical data to help the model handle the input more effectively. In this part, we apply the unique hot coding method [11] to map character data to numerical data. The mapped character codes are [1, 0, 0], [0, 1, 0], and [0, 0, 1]. Following standardised processing, the feature data has 122 dimensions after deleting all zero vectors, with the remaining 121 dimensions. To reduce dimension differences between feature data after standardisation, the experiment employs maximum and minimum normalisation approaches [12].

The first stage in training a CNN model is to set the starting parameters, which are subsequently adjusted continually based on the training outcomes of each round until the model detection effect reaches its optimal state. In the hybrid algorithm model's CNN structure, the convolution layer is located between the second and fourth layers. This part uses a convolution kernel of  $1 \times 3$  and  $1 \times$ . The number of convolution kernels is 16 and 32, respectively. The key rationale for establishing tiny convolution kernel is because numerous research have proven that utilising small convolution kernel as filter may enable CNN better recognise and classify data characteristics, so as to boost the precision of classification and detection for small-scale attacks. The pooling layer is the third and fifth layers. The goal of adjusting the step size to 2 is to minimise model parameters and total computational complexity, hence increasing calculation speed, reducing error caused by classification imbalance, and improving overall model detection performance.

### **2.9.1 Algorithm:**

One possible implementation of an intelligent attack classification networks in intrusion detection systems using ADTree be described as follows: Ones of primary goals of network intrusion detection in system to categorizes various types of network attacks using input data derived from the KDD dataset.. First, the data is loaded into the system. Preprocessing is then applied to fill in any missing values in the data. The datasets are then clustered into four types based on the type of attack DOS, probe, r2land U2R. Each cluster is then partitioned into training and test sets. ADTree algorithm is applied on each of these sets for training. The testing datasets is then field to AOD for attacks classification. Accuracies detected rate DR and falsely alarm rates FAR are recorded to evaluation the performance of systems.

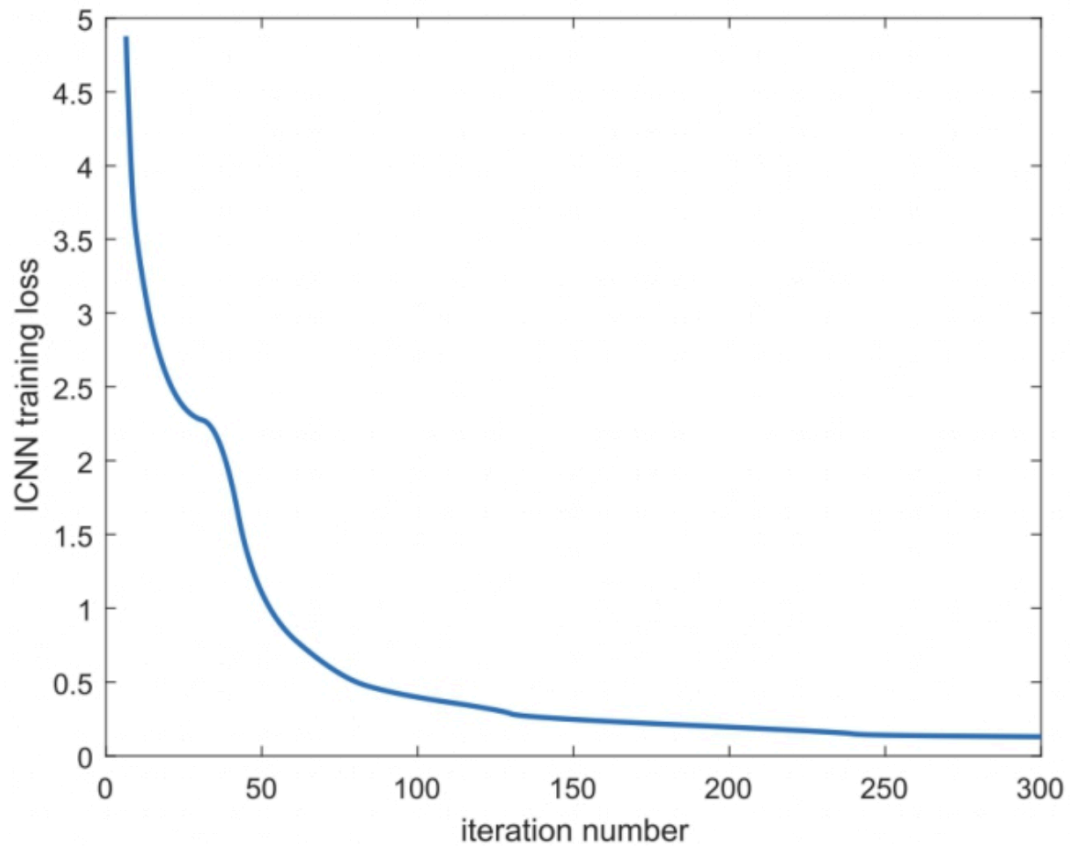


Figure 2.8:Relation of Iterate Numbers and ICNN training error

### 2.9.2 Replace missing values:

The Weka software was utilized to replaced any miss value in the NSLKDD dataset. Specifically, the "replace missing values" filter was applied, which replacing skip feature measures of the mean and mode values obtained through the train data. The KDD dataset was chosen for this study as its an advanced version for KDD he whole dataset also comprises of 42 attributes, with the last attribute indicating whether to check a particular thing normal instance attack. To obtain the completed dataset, please refers to the appropriate source. The performances of classify was evaluate using the following measures, and 10-fold cross validation was employed for classification purposes.



## CHAPTER - 3

### Result and Analysis

#### 3.1 Configuration of Experimental Environment and Hyperparameter Settings

In this report, The experimentation, validation, and comparative analysis of the proposed models were conducted on the Linux operating system. Python's deep learning library Tensor flow was utilized to implement CNN. To enhance computational efficiency and reduce training time, Tensor flow GPU was employed for parallel computing acceleration. The experimental arrangement utilised optimised software and hardware. The test model is verified and compared using the Linux operating system. This article demonstrates how to implement ICNN and IBWNIDM using Tensorflow, a Python deep learning package. TensorflowGPU accelerates computation and reduces training time. Table 5 shows the experimental software and hardware setup environment.

Hardware Configuration	Software Environment
Intel Core i7-7700 CPU @ 2.80GHz	Ubuntu16.04
NVidia GeForce GTX 1050	Python3.5.2
16.0GB RAM	Pycharm2019

Figure 3.1: Experiment Environmental Configurations

Following repeated and smallscale parameter combinat train based to the train and testing results decide the superparameter setting of ICNN of training as follows: The network learning rates is 0.2. The weighted deactivation rate of the regularisation technique Drop out is set at 0.5 with a total of 300 experiments iterations training cycles. Training setdata is used to choose each iterations of the experimental procedure. The dataset consists of 1000 items with 50 iteration for each batch size.

#### 3.2 EVALUATION INDICATORS

This report utilize three primary indicators, namely Accuracy AC, truly Positive Rate TPR and Falsely Positive Rate FPR to asses the performance of intrusion detection. First indicator is the proportionate of accurate classified samples to the total numbers of sample test. Indicator 1 TPR: The classify accuracy is defined as the proportion of accurate classify sample to the totals number of sample that are tested, and it is important metrics for evaluate performance of a classification models.

$$TPR = \frac{TP}{TP + FN}$$

Equation 3.4 : Total PositiveRate

Intrusion in Network Detection System (IDS) typical comprises three component namely:

1. Dataset Source: This component is also known as an event generator, as it generates events that are monitored by the IDS.
2. Analysis Engine: The Analysis Engine receives inputs from dataset sources and verifies whether to check any symptom for attacking are present.
3. Response Manager: The Response Manager is responsible for taking appropriate actions when potential attacks are detected on the system and notifying relevant parties.

Indicators 3 FPR: Classifier classify negatives samples wrong the ratio of the numbers of positive sample to the number of all negatives sample.

$$FPR = \frac{FP}{FP + TN}$$

Equation 3.5: False Positive Rate

The precise functioning of an IDS is essential in creating a reliable Network Intrusion Detection System (NIDS). NIDS with either low detect rate(DR) or high falsely positive rates(FPR) are often inoperable. To enhance attack detection performance, our proposal introduces an intelligent IDS that leverages the ADTree algorithm.

Below, we provide a description of the algorithm we have proposed.

The True Positive (TP) class indicates the numbers of sample accurately categorised positive. The true Negatives(TN) class represents the numbers of sample that were accurately categorised as negative. The false positive class (FP) represents the number of negative sample misclassified as positive whereas the false negative class(FN) represent the numbers of positive samples misclassified as negative.

### 3.3 Experiment Design

Three test trials were run utilising the KDDTest and NSLKDD CUP data sets. The experimental design is as follows.

Experiment 1: The test set (KDDTest) contains five data labels: Normal, Probe, DOS, u2rand R2L, which are categorised into five categories and detected. The accuracy, true positive rates and false positive rates indicators of IBWNIDM for four type of assaults are calculated using the five-dimensional confusion matrix (detection result) produced by categorization detection.

Experiment 2 involves applying the NSLKDD data set to three common neural networks used in intrusion detection: Lenet5 [18] DBN [19] and RNN [20]. Conduct train and test and compare the outcome to detection findings of IBWNIDM.

Experiment 3: Using the NSLKDD dataset two CNN-based intrusion detection models, IDABCNN and NIDMBCNN [15] [16] are trained and evaluated. The detection effect is compared to IBWNIDM.

### 3.4 Experimental Result

#### Classifications of Test and Result

The investigation includes two phases: train and test. Figure 5 depicts the link between training error and IBWNIDM iteration count. Figure 5 shows that the numbers of iteration increase the training errors gradually decrease. At 50 iterations error values is the smallest indicating that superparameter settings of the ICNN models structure designed and model training is reasonable and meets the detection requirement.

### 3.5 Evaluation and Results

Result of Comparison of ICNN, KNN, Logistic regression and decision tree algorithm

Model	Train Score	Test Score
KNN	0.982817	0.98227
Logistic Regression	0.928774	0.923128
ICNN	0.99983	0.996031

Figure 3.2: Train and Test Score of Algorithms

Comparing the performances of different machines learning algorithm on data involves

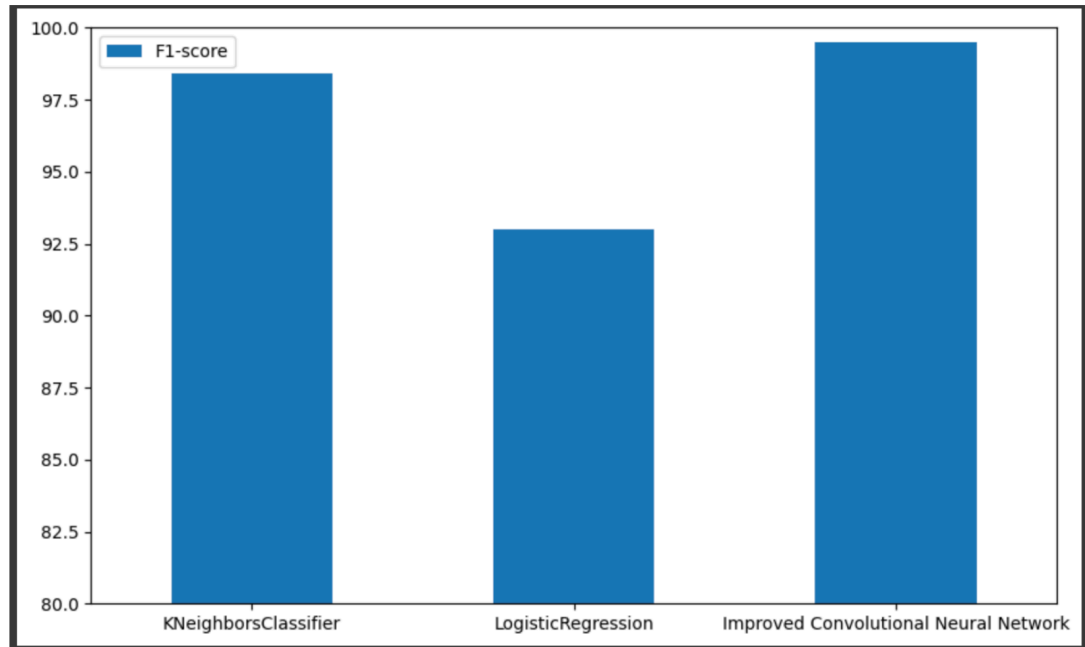


Figure 3.3: F1-Score of algorithms

evaluating their train and test scores, which are measures of how well the models fit the train data and how well they general to seen dataset respectively. Here, we'll consider a hypothetical comparison of four algorithms: Improved Convolutional Neural Network (ICNN), KNearest Neighbor (KNN), Logistic Regressions and Decisions Tree.

Evaluating the performance of networks in intrusion detection system (NIDS) using metric like precisions recalls and F1 score is essential understand how well the system distinguishes between normal and malicious network traffic. Here's a breakdown of these metrics and how they might look when applied to different machine learn algorithms in context of networks in intrusion detection.

```

***** KNeighborsClassifier Model Testing *****
[[3435  63]
 [  65 3995]]
-----
              precision    recall  f1-score   support

   normal         0.98         0.98         0.98         3498
  anomaly         0.98         0.98         0.98         4060

 accuracy         0.98         0.98         0.98         7558
 macro avg         0.98         0.98         0.98         7558
 weighted avg         0.98         0.98         0.98         7558

***** LogisticRegression Model Testing *****
[[3127  371]
 [ 210 3850]]
-----
              precision    recall  f1-score   support

   normal         0.94         0.89         0.91         3498
  anomaly         0.91         0.95         0.93         4060

 accuracy         0.92         0.92         0.92         7558
 macro avg         0.92         0.92         0.92         7558
 weighted avg         0.92         0.92         0.92         7558

***** Improved Convolutional Neural Network Model Testing *****
[[3484  14]
 [  27 4033]]
-----
              precision    recall  f1-score   support

   normal         0.99         1.00         0.99         3498
  anomaly         1.00         0.99         0.99         4060

 accuracy         0.99         0.99         0.99         7558
 macro avg         0.99         0.99         0.99         7558
 weighted avg         0.99         0.99         0.99         7558

```

Figure 3.4: Confusion Matrix

- F1 Score: The harmonic mean of accuracy and recall. It offers a single statistic to balance both issues.
- Precision: The percentage of real positive detections (actual incursions successfully recognised) among all positive detections (including true and false positive). It evaluates the accuracy of optimistic predictions.
- Recall (Sensitivity): The fraction of genuine positive detections compared to total invasions (including false negative). It assesses the model's ability to identify all relevant instances in the dataset.

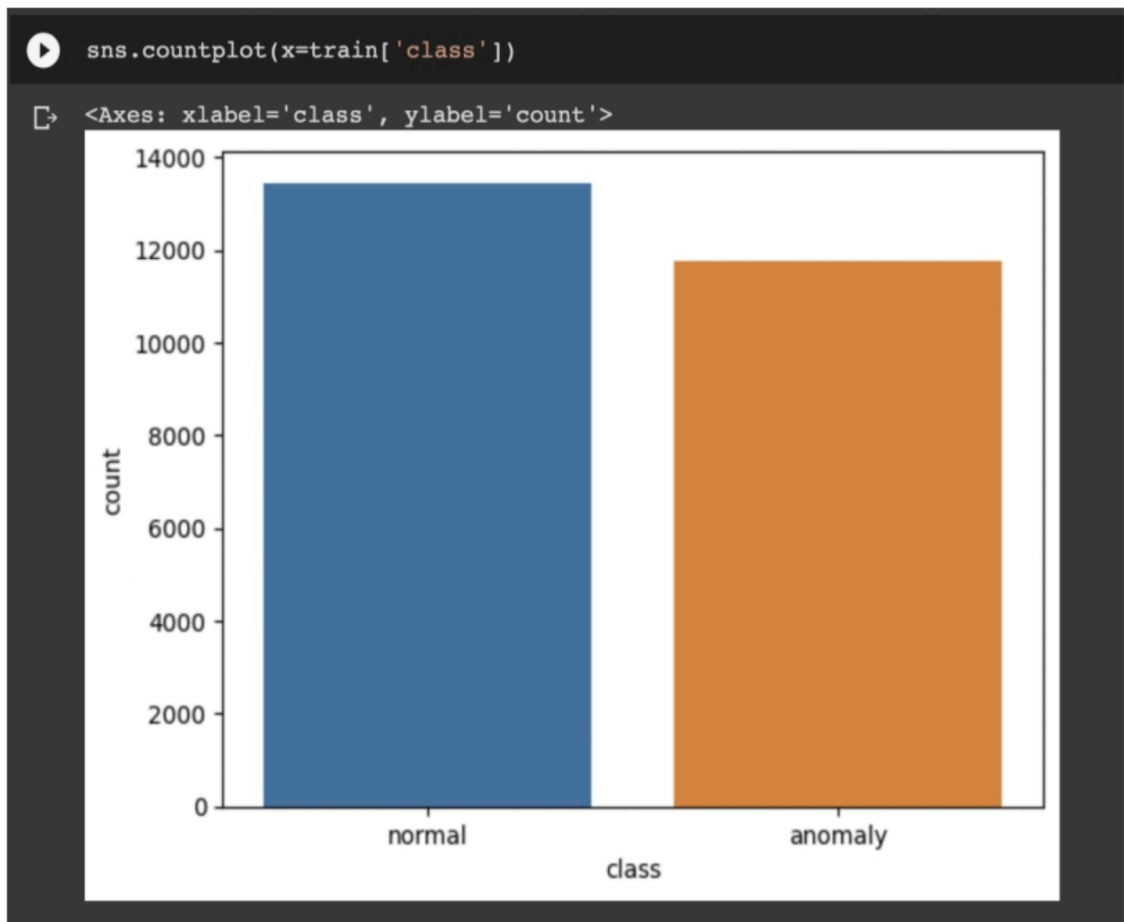


Figure 3.5: Class Partition Graph

## CHAPTER - 4

### 4.1 Conclusion and Future Work

The concept of detours refers to the ethical behavior standards during street accidents and everyday activities. To manage the IDS access road, the tree-based approach is most suitable and integrated into the genetic algorithm of prevention. In contrast, this technology effectively avoids excessive regulations, such as firewalls. This report proposes a unique way to classifying intrusion attempts using intrusion data using alternating decision tree (ADT). Our simulation results were based on the dataset, where the attack types were classified into three categories. Our proposed system effectively classifies various types of attacks, and we measured the Our proposed methodology entails partitioning the dataset into four distinct clusters, followed by training and testing the ADTree algorithm on each cluster and subsequently deploying the algorithm for attack classification. Additionally, we evaluate the efficacy of approached based on multiple performance metric such as detection rate accuracy and false alarm rate. Our experimental results affirm that our intelligent NIDS built using the ADTree algorithm is an effective mechanism for detecting networks intrusions, thus enhance overall securities of an organization network. In conclusion, our proposed method can serve as an extra layer of protection against cyber attacks and proactively mitigate any potential damage to an organization's network.

To next enhance the application of ICNN in networks intrusion detection for civil aviation, several avenues for future work are recommended:

Develop more extensive datasets that encompass a wider range of normal and anomalous network traffic specific to aviation environments. Employ semi-supervised learning and active learning to improve the efficiencies and accuracies of data labeling reducing the dependency on large labeled datasets. Investigate techniques such as model prune quantize and the developed of more efficient neural networking architecture to decrease the computational load without sacrificing accuracy. Explore distributed computing frameworks and edge computing solutions to decentralize processing, reduce latency, and enhance real-time detection capabilities. Create visualization tools that can illustrate the decision-making process of ICNNs, aiding in the validation and trust-building of AI-based detection systems. By addressing these areas, the effectiveness and reliabilities of ICNN-based networks in intrusion detection systems in civil aviation can be significantly improved. This will contribute to the overall cybersecurity posture of aviation networks, enhancing their resilience against far range of cyber threat.

## 4.2 References

- [1] R. M. A. Ujjan, Z. Pervez, K. Dahal, A. K. Bashir, R. Mumtaz, and J. González, "Towards sFlow and adaptive polling sampling for deep learning based DDoS detection in SDN," *Future Generation Computer Systems*, vol. 111, pp. 763-779, 2020.
- [2] M. Nobakht, V. Sivaraman, and R. Boreli, "A Host -Based Intrusion Detection and Mitigation Framework for Smart Home IoT Using OpenFlow," presented at the 2016 11th International Conference on Availability, Reliability and Security (ARES), 2016.
- [3] P. Ghosh, C. Debnath, D. Metia, and Dr. R. Dutta, An Efficient Hybrid Multilevel Intrusion Detection System in Cloud Environment, *IOSR Journal of Computer Engineering (IOSRJCE)* e- ISSN: 2278-0661, p-ISSN: 2278-8727, Volume 16, Issue 4, Ver. VII (Jul –Aug. 2014), PP 16-26.
- [4] IDS.RUUdataset[EB/OL].<http://ids.cs.columbia.edu/content/ruu.html>, 2017-11-20. LANE T, BRODLEY C E. An Application of Machine Learning to Anomaly Detection,2017-11-20.
- [5] GREENBERU S.Using UNIX: Collected Traces of 16R Users[EB/OL].<https://dspace.ucalgary.ca/handle/1880/45929>,2017- 11-20.
- [6] K HOLIDY H A, BAIARDI F. CIDD: A Cloud Intrusion Detection Dataset for Cloud Computing and Masquerade Attacks[C]//IEEE. Ninth International Conference on Information Technology: New Generations, April 16-18,2012,Las Vegas, NV,USA. NJ:IEEE, 2012:397-402
- [7] CHEN Hong-song, HAN Zhi, DENG Shu-ning. Analysis and Research on Big Data Security in Smart City[J]*Netinfo Security*,2015(7):1-6
- [8] Qian Mao, Student Member, IEEE, Fei Hu, Member, IEEE, and Qi Hao, Member, IEEE, "Deep Learning for Intelligent Wireless Networks: A Comprehensive Survey" *JOURNAL OF LATEX CLASS FILES*, VOL. 14, NO. 8, JANUARY 2018.
- [9] C. Lu, "Research on the technical application of artificial intelligence in network Intrusion detection system," 2022 International Conference on Electronics and Devices, Computational Science (ICEDCS), Marseille, France, 2022, pp. 109-112, doi: 10.1109/ICEDCS57360.2022.00031.



- [10] Nivedhidha, M., Ramkumar, M.P. and GSR, E.S., 2023, July. CopulaGAN Boosted Random Forest based Network Intrusion Detection System for Hospital Network Infrastructure. In *2023 14th International Conference on Computing Communication and Networking Technologies (ICCCNT)* (pp. 1-7). IEEE.
- [11] S. Zheng, "Network Intrusion Detection Model Based on Convolutional Neural Network," *2021 IEEE 5th Advanced Information Technology, Electronic and Automation Control Conference (IAEAC)*, Chongqing, China, 2021, pp. 634-637, doi: 10.1109/IAEAC50856.2021.9390930.
- [12] D. Liao, S. Huang, Y. Tan and G. Bai, "Network Intrusion Detection Method Based on GAN Model," *2020 International Conference on Computer Communication and Network Security (CCNS)*, Xi'an, China, 2020, pp. 153-156, doi: 10.1109/CCNS50731.2020.00041.
- [13] X.Li, "Research and Design of Network Intrusion Detection System," *2022 IEEE 2nd International Conference on Power, Electronics and Computer Applications (ICPECA)*, Shenyang, China, 2022, pp. 1069-1072, doi: 10.1109/ICPECA53709.2022.9718920.
- [14] X. Zhan, H. Yuan and X. Wang, "Research on Block Chain Network Intrusion Detection System," *2019 International Conference on Computer Network, Electronic and Automation (ICCNEA)*, Xi'an, China, 2019, pp. 191-196, doi: 10.1109/ICCNEA.2019.00045.
- [15] B. -S. Lee, J. -W. Kim and M. -J. Choi, "Federated Learning Based Network Intrusion Detection Model," *2023 24th Asia-Pacific Network Operations and Management Symposium (APNOMS)*, Sejong, Korea, Republic of, 2023, pp. 330-333.
- [16] J. Li, "Network Intrusion Detection Algorithm and Simulation of Complex System in Internet Environment," *2022 4th International Conference on Inventive Research in Computing Applications (ICIRCA)*, Coimbatore, India, 2022, pp. 520-523, doi: 10.1109/ICIRCA54612.2022.9985720.
- [17] J. Chen, Y. Guo, K. Shi and M. Yang, "Network Intrusion Detection Method of Power Monitoring System Based on Data Mining," *2022 2nd International Conference on Algorithms, High Performance Computing and Artificial Intelligence (AHPCAI)*, Guangzhou, China, 2022, pp. 255-259, doi: 10.1109/AHPCAI57455.2022.10087405.

- [18] B. Xiang, C. Zhang, J. Wang and B. Wang, "Network Intrusion Detection Method for Secondary System of Intelligent Substation based on Semantic Enhancement," *2022 4th International Conference on Electrical Engineering and Control Technologies (CEEECT)*, Shanghai, China, 2022, pp. 796-800, doi: 10.1109/CEEECT55960.2022.10030264.
- [19] C. Chen, X. Xu, G. Wang and L. Yang, "Network intrusion detection model based on neural network feature extraction and PSO-SVM," *2022 7th International Conference on Intelligent Computing and Signal Processing (ICSP)*, Xi'an, China, 2022, pp. 1462-1465, doi: 10.1109/ICSP54964.2022.9778404.
- [20] L. Zhang, H. Yan and Q. Zhu, "An Improved LSTM Network Intrusion Detection Method," *2020 IEEE 6th International Conference on Computer and Communications (ICCC)*, Chengdu, China, 2020, pp. 1765-1769, doi: 10.1109/ICCC51575.2020.9344911.
- [21] P. Zhang, G. Tian and H. Dong, "Research on network intrusion detection based on Whitening PCA and CNN," *2023 7th International Conference on Smart Grid and Smart Cities (ICSGSC)*, Lanzhou, China, 2023, pp. 232-237, doi: 10.1109/ICSGSC59580.2023.10319169.
- [22] A. Shah, S. Clachar, M. Minimair and D. Cook, "Building Multiclass Classification Baselines for Anomaly-based Network Intrusion Detection Systems," *2020 IEEE 7th International Conference on Data Science and Advanced Analytics (DSAA)*, Sydney, NSW, Australia, 2020, pp. 759-760, doi: 10.1109/DSAA49011.2020.00102.



# DELHI TECHNOLOGICAL UNIVERSITY

(Formerly Delhi College of Engineering)

Shahbad Daulatpur, Main Bawana Road, Delhi-42

## PLAGIARISM VERIFICATION

Title of the Thesis Network Intrusion Detection in Civil Aviation  
Based on Improved Convolutional Neural Network

Total Pages 42 Name of the Scholar Sachin Bhuwal

Supervisor (s)

(1) Mr. Rahul

(2) \_\_\_\_\_

(3) \_\_\_\_\_

Department Software Engineering

This is to report that the above thesis was scanned for similarity detection. Process and outcome is given below:

Software used: Turnitin Similarity Index: 12 %, Total Word Count: 10590

Date: 27/05/24

Candidate's Signature

Signature of Supervisor(s)



**DELHI TECHNOLOGICAL UNIVERSITY**  
(Formerly Delhi College of Engineering)  
Shahbad Daultpur, Main Bawana Road, Delhi-110042, India

**CERTIFICATE OF FINAL THESIS SUBMISSION**

(To be submitted in duplicate)

1. Name: Sachin Bhure

2. Roll No: 2k22/SWF/116

3. Thesis title: Network Intrusion Detection in Civil Aviation  
Based on Improved Convolutional Neural Network

4. Degree for which the thesis is submitted: M. TECH

5. Faculty (of the University to which the thesis is submitted)  
Mr. Rahul

6. Thesis Preparation Guide was referred to for preparing the thesis. YES  NO

7. Specifications regarding thesis format have been closely followed. YES  NO

8. The contents of the thesis have been organized based on the guidelines. YES  NO

9. The thesis has been prepared without resorting to plagiarism. YES  NO

10. All sources used have been cited appropriately. YES  NO

11. The thesis has not been submitted elsewhere for a degree. YES  NO

12. All the correction has been incorporated. YES  NO

13. Submitted 2 hard bound copies plus one CD. YES  NO

(Signature(s) of the Supervisor(s))

Name(s): Mr. Rahul

(Signature of Candidate)

Name: Sachin Bhure

Roll No: 2k22/SWF/116

PAPER NAME

Sachin Bhure Thesis.pdf

WORD COUNT

10590 Words

CHARACTER COUNT

61295 Characters

PAGE COUNT

42 Pages

FILE SIZE

3.5MB

SUBMISSION DATE

May 27, 2024 2:46 PM GMT+5:30

REPORT DATE

May 27, 2024 2:47 PM GMT+5:30

● **12% Overall Similarity**

*Sachin Bhure*  
27/5/24

The combined total of all matches, including overlapping sources, for each database.

- 5% Internet database
- 8% Publications database
- Crossref database
- Crossref Posted Content database
- 6% Submitted Works database

● **Excluded from Similarity Report**

- Bibliographic material
- Quoted material
- Small Matches (Less than 10 words)

## ● 12% Overall Similarity

Top sources found in the following databases:

- 5% Internet database
- 8% Publications database
- Crossref database
- Crossref Posted Content database
- 6% Submitted Works database

### TOP SOURCES

The sources with the highest number of matches within the submission. Overlapping sources will not be displayed.

1	<b>Hongyu Yang, Fengyan Wang. "Wireless Network Intrusion Detection B...</b>	4%
	Crossref	
2	<b>Zhijun Wu, Cheng Liang, Yuqi Li. "Intrusion Detection Method based on ...</b>	2%
	Crossref	
3	<b>dspace.dtu.ac.in:8080</b>	2%
	Internet	
4	<b>Delhi Technological University on 2018-12-04</b>	<1%
	Submitted works	
5	<b>Delhi Technological University on 2024-05-23</b>	<1%
	Submitted works	
6	<b>Glyndwr University on 2024-03-03</b>	<1%
	Submitted works	
7	<b>dspace.dtu.ac.in:8080</b>	<1%
	Internet	
8	<b>Xin Guan, Xinyu Cao. "Network Intrusion Detection Method Based on A...</b>	<1%
	Crossref	

- 9 Priya Singh, Abhineet Prakash, S.K. Parida. "Neural network based patt... <1%  
Crossref

---
- 10 Kookmin University on 2020-06-01 <1%  
Submitted works

---
- 11 digital.slub-dresden.de <1%  
Internet

---
- 12 Delhi Technological University on 2020-06-30 <1%  
Submitted works

---
- 13 Delhi Technological University on 2024-05-22 <1%  
Submitted works

---
- 14 era.library.ualberta.ca <1%  
Internet

---
- 15 bookdown.org <1%  
Internet

---
- 16 Imtiaz Ullah, Qusay H. Mahmoud. "A Deep Learning Based Framework ... <1%  
Crossref

---
- 17 Khalid M. Mosalam, Yuqing Gao. "Artificial Intelligence in Vision-Based... <1%  
Crossref

---
- 18 The University of Wolverhampton on 2022-05-05 <1%  
Submitted works

---
- 19 Tilburg University on 2024-05-20 <1%  
Submitted works

---
- 20 Xuanrui Xiong, Yufan Zhang, Huijun Zhang, Yi Chen, Hailing Fang, Wen ... <1%  
Crossref

21	<b>amsdottorato.unibo.it</b> Internet	<1%
22	<b>downloads.hindawi.com</b> Internet	<1%
23	<b>fastercapital.com</b> Internet	<1%
24	<b>technodocbox.com</b> Internet	<1%
25	<b>ai.rug.nl</b> Internet	<1%