iJETRM
International Journal of Engineering Technology Research & Management
(IJETRM)
https://ijetrm.com/

# REVOLUTIONIZING IDS A NOVEL FRAMEWORK USING JAYA OPTIMIZER AND SMOTE-ENN FOR CYBERATTACKS DETECTION

**Vislavath Ramu**
Post Graduate Student, MCA, Department of Information Technology, Jawaharlal Nehru Technological University, Hyderabad, India

**G. Narasimham**
Associate Professor, Department of Information Technology, Jawaharlal Nehru Technological University, Hyderabad, India

**ABSTRACT**
Network Intrusion Detection Systems (NIDS) are pivotal in addressing the increasing cybersecurity threats by protecting computer networks from various types of cyberattacks. However, class imbalances in datasets often reduce the effectiveness of detection algorithms, leading to misclassification of minority classes. Additionally, poor feature selection can hinder performance and increase computational load. In this study, we utilize the NSL-KDD and UNSW-NB15 datasets to develop a robust IDS model. We applied Jaya Optimization for feature selection to enhance relevant data features and utilized the SMOTE-ENN method for oversampling to balance the dataset. Various algorithms were tested, including Decision Tree, Random Forest, Bagging with Decision Trees, J48, ExtraTree, and a Voting Classifier combining ExtraTree with Boosted Decision Tree.

**Keywords:**
Intrusion Detection System, Jaya Optimization, SMOTE-ENN, Machine Learning, Class Imbalance.

## I. INTRODUCTION

The rapid expansion of the Internet and associated networks has resulted in a substantial rise in data traffic, with the total number of Internet users projected to reach 5.3 billion in 2024. With the expansion of networks, the risk of network infiltration has intensified, requiring stringent technologies, protocols, and safeguards to protect systems, networks, applications, devices, and data from cyber threats and assaults. Security managers often emphasise password protection systems, encryption protocols, and access restrictions in conjunction with firewalls to safeguard the network.

**Objective:**
The objective of this study is to develop an enhanced Network Intrusion Detection System (NIDS) that effectively detects cyberattacks by addressing common challenges like class imbalance and feature selection

## II. LITERATURE SURVEY

The Internet of Things (IoT) and machine learning (ML) have various applications in different sectors of life, such as healthcare, agriculture, industries, transportation, smart cities, smart homes, etc., and their number is increasing with each passing day. The rapid development of IoT and its increasing demand in different fields of life create a serious problem of security for the IoT environment, which needs serious consideration to protect the IoT-enabled systems from external networks and cyber-attacks. Because of the open deployment environment and constrained resources, the IoT is prone to malicious assaults.

## III. METHODOLOGY

### 3.1 Proposed System

The proposed system aims to enhance Network Intrusion Detection System (NIDS) accuracy by addressing the challenges of class imbalance and feature selection, which commonly hinder effective cyberattack detection. Using NSL-KDD and UNSW-NB15 datasets, the system implements Jaya Optimization for efficient feature

iJETRM
International Journal of Engineering Technology Research & Management
(IJETRM)
https://ijetrm.com/

selection, allowing for improved model performance and reduced computational time. To address class imbalances in attack detection, the Synthetic Minority Over-sampling Technique combined with Edited Nearest Neighbor (SMOTE-ENN) is employed, creating a balanced dataset for effective classification.

**3.2 Dataset Description**

For this analysis, the NSL-KDD and UNSW-NB15 datasets are utilized, consisting of 125,972 and 82,332 instances respectively, with features related to network traffic and attack categories for intrusion detection

**3.3 System Architecture**

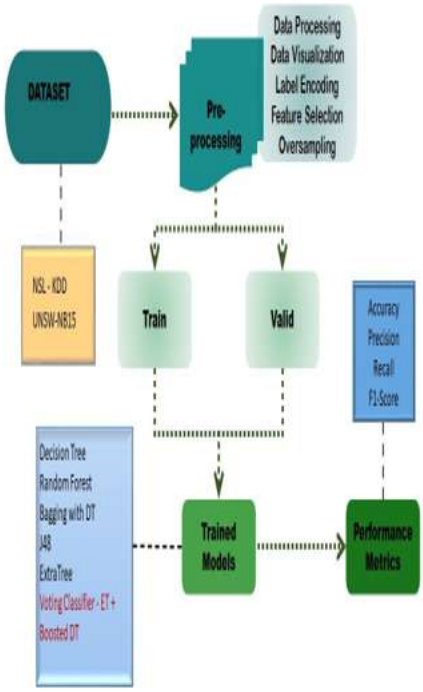



Figure.1: System Architecture Diagram.

This image visually represents the high-level components and their interactions within a system. It likely shows how different parts of the system—such as the user interface, web server, application logic, and data storage—are connected.

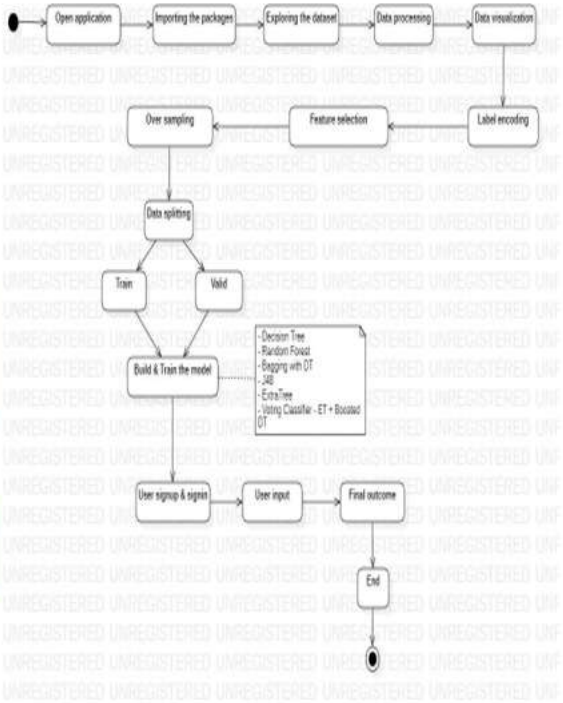



Figure.2: Activity Diagram.

An activity diagram depicts the flow of actions and decisions involved in the NIDS. Actions include importing packages, processing data, and visualizing results, while decisions guide the control flow based on conditions such as user input or data validation.

iJETRM
International Journal of Engineering Technology Research & Management
(IJETRM)
https://ijetrm.com/

### 3.4 Methodology (ML Pipeline)

The machine learning pipeline was executed as follows:

1. Data Preprocessing: Data processing involves cleaning and preparing the datasets for machine learning tasks. This step includes handling missing values, removing duplicates, and standardizing numerical values to ensure consistency across the dataset.
2. Train-Test Split: The dataset was partitioned into an 80% training set and a 20% testing set to allow for unbiased model evaluation.
3. Data Visualization**:** Data visualization aims to provide insights into the dataset through graphical representations.
4. Label Encoding: Label encoding is a technique used to convert categorical variables into numeric form, making them compatible with machine learning algorithms that require numerical input. In this step, each unique category in the dataset is assigned an integer value

## IV. Experiment and Analysis

The experimental phase focused on training and validating the chosen machine learning model and then integrating it into a functional web application with a rich feature set.

### 4.1 Key Features

The final implemented system boasts several key features designed to provide a comprehensive and user-friendly experience:

- Accuracy**:** The accuracy of a test is its ability to differentiate the patient and healthy cases correctly. To estimate the accuracy of a test, we should calculate the proportion of true positive and true negative in all evaluated cases
- ExtraTree**:** ExtraTree accelerates training by randomly selecting thresholds for each feature, enhancing model diversity. This approach captures intricate data patterns, ensuring high accuracy and robustness in classifying network traffic, including novel intrusion attempts.
- J48: J48 generates Decision Trees based on information gain ratio, classifying network traffic efficiently by handling both categorical and continuous data while reducing overfitting through pruning techniques, enabling the identification of complex relationships within the data.

### 4.2 Experiment Analysis and Results

***Table.1 Performance Evaluation Table – NSL-KDD – With SMOTEENN***

| ML Model | Accuracy | Precision | Recall | F1_score | MCC |
|---|---|---|---|---|---|
| Decision Tree | 0.981 | 0.981 | 0.981 | 0.981 | 0.976 |
| Random Forest | 0.996 | 0.996 | 0.996 | 0.996 | 0.995 |
| Bagging DT | 0.983 | 0.983 | 0.983 | 0.983 | 0.979 |
| J48 | 0.983 | 0.983 | 0.983 | 0.983 | 0.979 |
| ExtraTree | 0.993 | 0.993 | 0.993 | 0.993 | 0.979 |
| **Extension** | **1.000** | **1.000** | **1.000** | **1.000** | **1.000** |

***Table.2 Performance Evaluation Table – NSL-KDD – Without SMOTEENN***

| ML Model | Accuracy | Precision | Recall | F1_score | MCC |
|---|---|---|---|---|---|
| Decision Tree | 0.970 | 0.969 | 0.970 | 0.969 | 0.961 |
| Random Forest | 0.983 | 0.984 | 0.983 | 0.983 | 0.977 |
| Bagging DT | 0.974 | 0.974 | 0.974 | 0.974 | 0.966 |
| J48 | 0.971 | 0.971 | 0.971 | 0.971 | 0.962 |
| Extra Tree | 0.978 | 0.981 | 0.978 | 0.979 | 0.971 |
| **Extension** | **1.000** | **1.000** | **1.000** | **1.000** | **1.000** |

iJETRM
International Journal of Engineering Technology Research & Management
(IJETRM)
https://ijetrm.com/

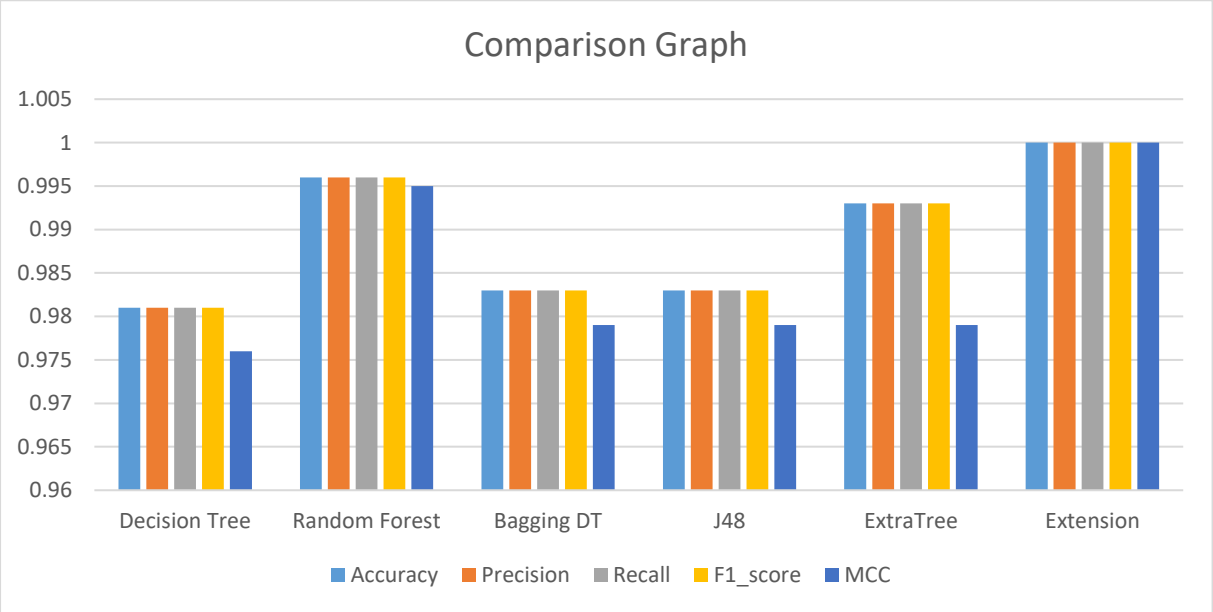
*Table.3 Performance Evaluation Table – UNSW-NB15 – With SMOTEENN*

| ML Model | Accuracy | Precision | Recall | F1_score | MCC |
|---|---|---|---|---|---|
| Decision Tree | 0.978 | 0.978 | 0.978 | 0.978 | 0.955 |
| Random Forest | 0.988 | 0.988 | 0.988 | 0.988 | 0.976 |
| Bagging DT | 0.982 | 0.982 | 0.982 | 0.982 | 0.964 |
| J48 | 0.984 | 0.984 | 0.984 | 0.984 | 0.968 |
| Extra Tree | 0.984 | 0.984 | 0.984 | 0.984 | 0.967 |
| **Extension** | **1.000** | **1.000** | **1.000** | **1.000** | **1.000** |

*Table.4 Performance Evaluation Table – UNSW-NB15 – Without SMOTEENN*

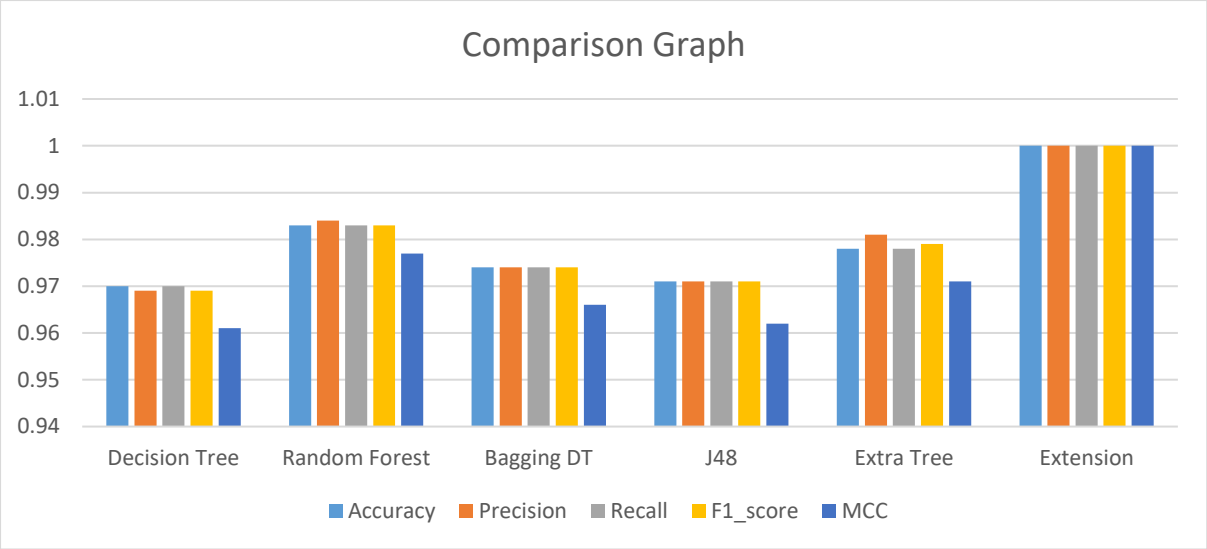
| ML Model | Accuracy | Precision | Recall | F1_score | MCC |
|---|---|---|---|---|---|
| Decision Tree | 0.908 | 0.908 | 0.908 | 0.908 | 0.817 |
| Random Forest | 0.924 | 0.925 | 0.924 | 0.924 | 0.849 |
| Bagging DT | 0.923 | 0.924 | 0.923 | 0.923 | 0.847 |
| J48 | 0.903 | 0.903 | 0.903 | 0.903 | 0.807 |
| ExtraTree | 0.920 | 0.921 | 0.920 | 0.920 | 0.841 |
| **Extension** | **0.920** | **0.920** | **0.920** | **0.920** | **0.840** |

*Graph.1 Comparison Graph – NSL-KDD – With SMOTEENN*

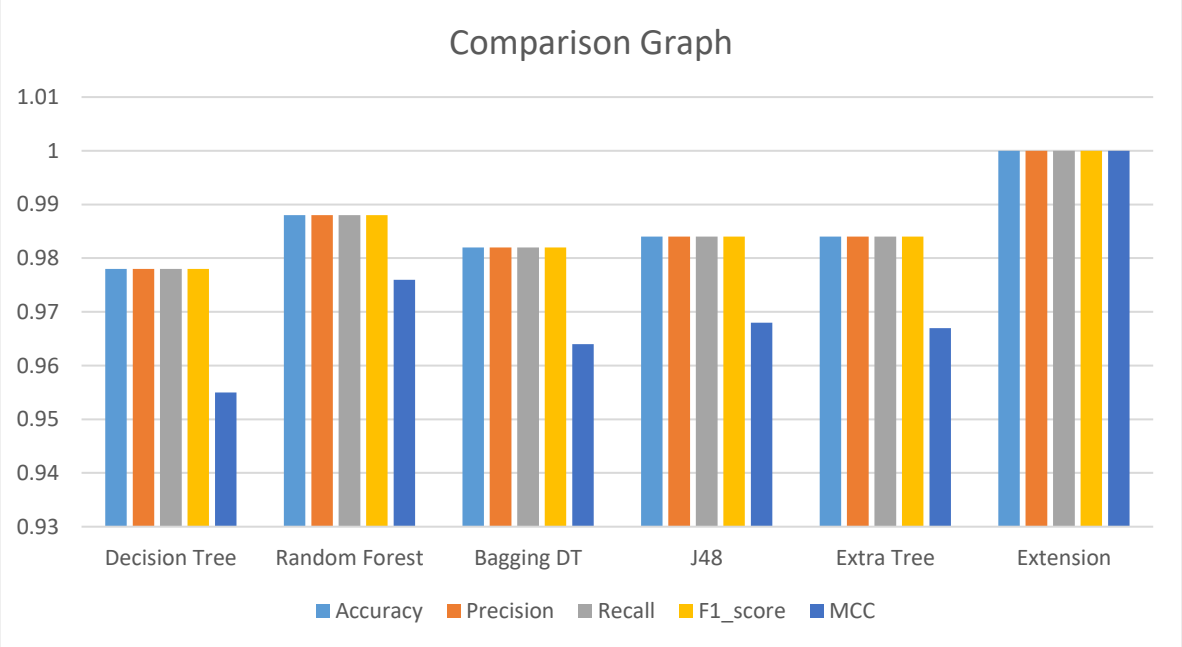



*Graph.2 Comparison Graph – NSL-KDD – Without SMOTEENN*

iJETRM
International Journal of Engineering Technology Research & Management
(IJETRM)
https://ijetrm.com/

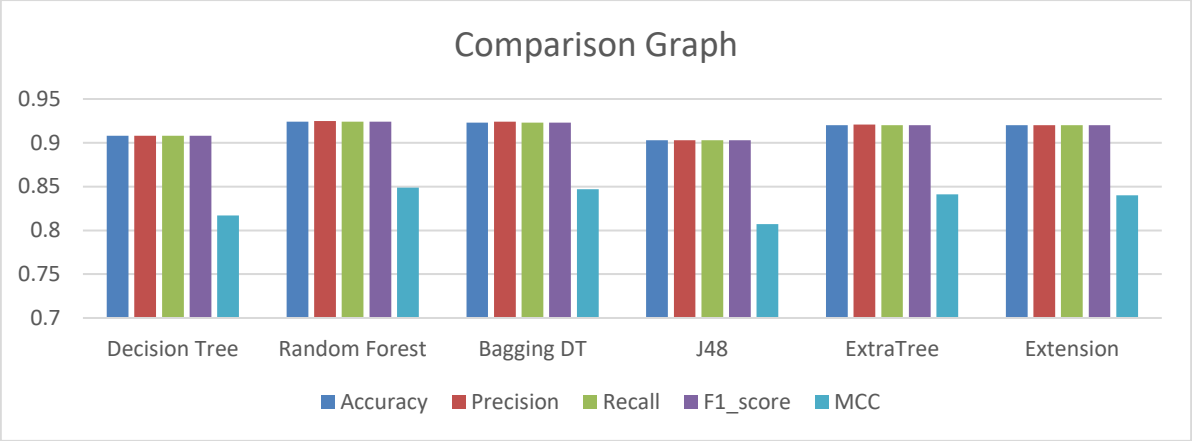



***Graph.3 Comparison Graph – UNSW-NB15 – With SMOTEENN***




***Graph.4 Comparison Graph – UNSW-NB15 – Without SMOTEENN***

iJETRM
International Journal of Engineering Technology Research & Management
(IJETRM)
https://ijetrm.com/




In graph (1, 2, 3 & 4) accuracy is represented in light blue, precision in orange, recall in grey, F1 Score in yellow and MCC and dark blue. The Graphs illustrate the Voting Classifier's superior performance across all metrics and datasets, consistently achieving the highest accuracy, precision, recall, and F1 scores, both with and without SMOTEENN, demonstrating its robustness and effectiveness in intrusion detection.

**User Dashboard**

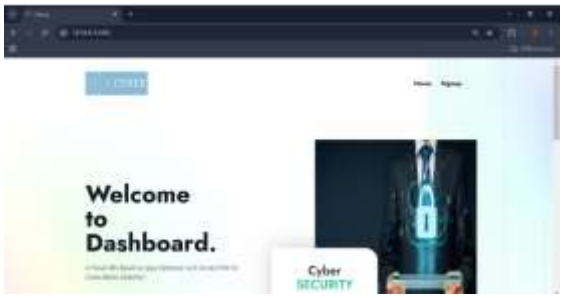



***Fig. 4 Dash Board***

The Fig. 4 shows the user dashboard of a cybersecurity application. The title is "Welcome to Dashboard" and the tagline is "A Novel Q3 Based on Java Optimizer and SmoteENN for Cyberattacks Detection."

**Step - 7**

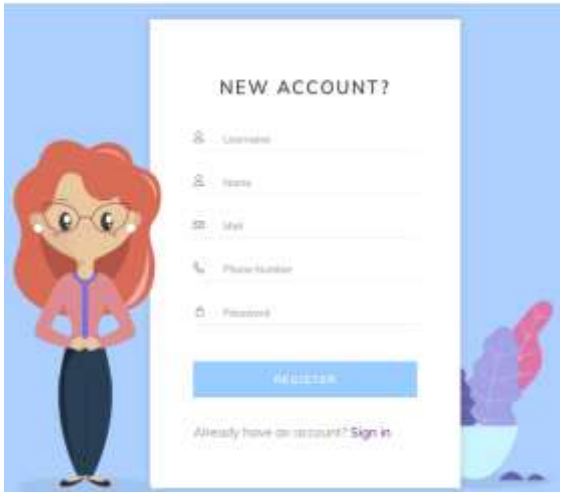



***Fig. 5 Register page***

The Fig. 5 shows a user registration form. It requires a username, name, email, phone number, and password. It also includes a "REGISTER" button and a link to "Sign in" for existing users.

iJETRM
International Journal of Engineering Technology Research & Management
(IJETRM)
https://ijetrm.com/

**Step - 8**

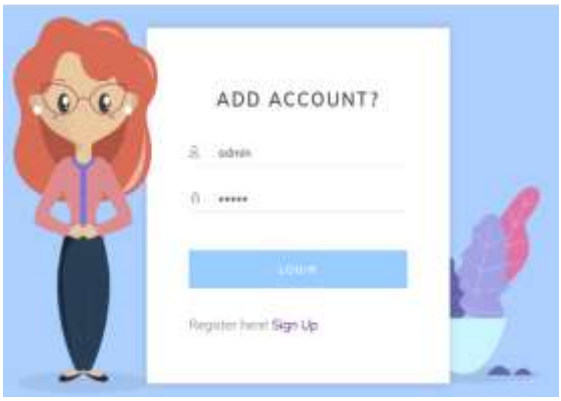



***Fig. 6 Login page***

The Fig. 6 shows a login page with a cartoon woman illustration. It has a pre-filled username "admin" and asks for a password. There's a "LOGIN" button and a link to "Sign Up" for new users.

**Step - 9**


***Fig. 7 Home page***

The Fig. 7 shows the main page of a web application related to intrusion detection. The user is selecting the "NSL - KDD" option from a dropdown menu under the "Prediction" tab.

**Step – 10, Test case 1**

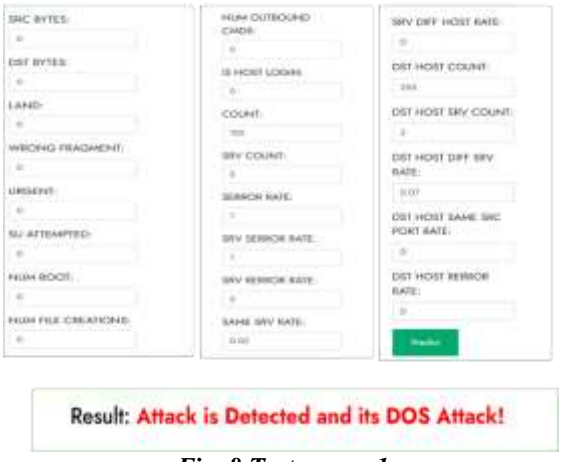



***Fig. 8 Test case – 1***

The Fig. 8 shows a network intrusion detection form. It collects data like source bytes, destination bytes, and other network statistics. After inputting data, the form predicts the outcome as a "DOS" attack.

iJETRM
International Journal of Engineering Technology Research & Management
(IJETRM)
https://ijetrm.com/


*Fig. 9 Test case – 2*

The Fig. 9 shows a network intrusion detection form. It collects data like source bytes, destination bytes, and other network statistics. After inputting data, the form predicts that "No Attack Detected and Its NORMAL!"


*Fig. 10 Test case – 3*

The Fig. 10 shows a network intrusion detection form. It collects data like source bytes, destination bytes, and other network statistics. After inputting data, the form predicts the outcome as a "PROBE" attack.


*Fig. 11 Test case - 4*

The Fig. 11 shows a network intrusion detection form. It collects data like source bytes, destination bytes, and

iJETRM
International Journal of Engineering Technology Research & Management
(IJETRM)
https://ijetrm.com/

other network statistics. After inputting data, the form predicts the outcome as an "R2L" attack.

**Step – 10, Test case 5**

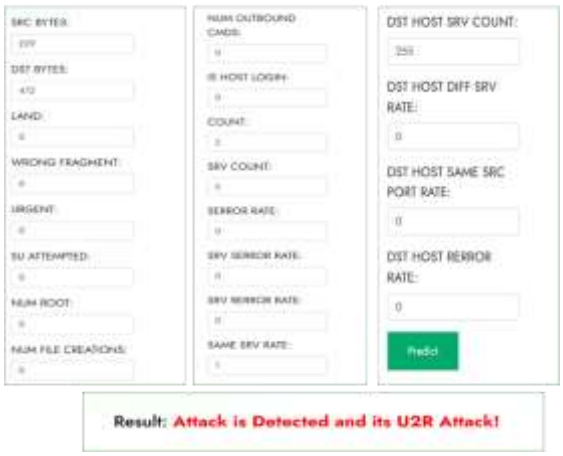



***Fig. 12 Test case – 5***

The Fig. 12 shows a network intrusion detection form. It collects data like source bytes, destination bytes, and other network statistics. After inputting data, the form predicts the outcome as a "U2R" attack.

**Step – 11**


***Fig. 13 Home page***

The Fig. 13 shows the main page of a web application related to intrusion detection. The user is selecting the "UNSW – NB15" option from a dropdown menu under the "Prediction" tab.

**Step – 12, Test case 1**


***Fig. 14 Test case – 1***

The Fig. 14 shows a network intrusion detection form. It collects data like packets, bytes, load, and other network statistics. After inputting data, the form predicts that "No Attack Detected!"

iJETRM
International Journal of Engineering Technology Research & Management
(IJETRM)
https://ijetrm.com/

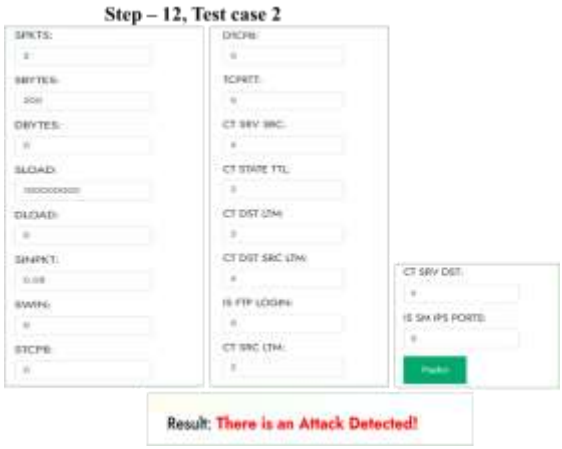



***Fig. 15 Test case – 2***

The Fig. 15 shows a network intrusion detection form. It collects data like packets, bytes, load, and other network statistics. After inputting data, the form predicts that "An Attack Detected!"

## V. Limitations and Future Scope

Future work could focus on integrating real-time streaming data analysis, allowing the IDS to respond swiftly to emerging threats. Additionally, incorporating advanced anomaly detection techniques and deep learning architectures could improve adaptability, enabling the IDS to identify novel attack patterns and evolving threats more effectively.

## VI. Conclusion

The rapid expansion of the Internet and its widespread connectivity has created extensive security vulnerabilities, leading to a rise in unauthorized access and cyberattacks on computer networks. This study addresses these vulnerabilities by proposing an enhanced Network Intrusion Detection System (NIDS) designed to improve detection accuracy and reliability. By using Jaya Optimization for feature selection and SMOTE-ENN for balancing data, the system effectively mitigates common issues like class imbalance and irrelevant features that often hinder the performance of traditional intrusion detection systems.

## VII. References

1) Statista. (2024). Number of Internet and Social Media Users Worldwide as of Jan. 2024. [Online]. Available: https://www.statista.com/statistics/617136/digital-population-worldwide/
2) G. Karatas, O. Demir, and O. K. Sahingoz, ‘‘Increasing the performance of machine learning-based IDSs on an imbalanced and up-to-date dataset,’’ IEEE Access, vol. 8, pp. 32150–32162, 2020.
3) T. T. Bhavani, M. K. Rao, and A. M. Reddy, ‘‘Network intrusion detection system using random forest and decision tree machine learning techniques,’’ in Proc. 1st Int. Conf. Sustain. Technol. Comput. Intell., 2019, pp. 637–643.
4) N. Farnaaz and M. A. Jabbar, ‘‘Random forest modeling for network intrusion detection system,’’ Proc. Comput. Sci., vol. 89, pp. 213–217, Jul. 2016.
5) N. Koroniotis, N. Moustafa, E. Sitnikova, and J. Slay, ‘‘Towards developing network forensic mechanism for botnet activities in the IoT based on machine learning techniques,’’ in Proc. Int. Conf. Mobile Netw. Manag., 2017, pp. 30–44.