Intrusion Detection System using NSL-KDD Dataset and Decision Tree Machine Learning Algorithm

Victor Somadina ND-ASOGU[1], John Oluwadara FATOKUN[2], Chukwudi Anthony UDEMBA[1], Olajide ADEGUNWA[1], Jelili Idris Olawale[3], Rashid Kehinde OLOKO[1], Theophilus Bamise AJALA[1*]

[1]Department of Computer Science, College of Computing and Information Sciences, Caleb University, KM 15, Ikorodu-Itoikin Road, Imota, P.M.B. 21238, Lagos State, Nigeria

[2]Department of Mathematics, College of Pure and Applied Sciences, Caleb University, KM 15, Ikorodu-Itoikin Road, Imota, P.M.B. 21238, Lagos State, Nigeria

[3]Department of Cybersecurity, School of Science and Technology, Christopher University, Mowe, Ogun State, Nigeria

***Corresponding Author**

DOI: https://doi.org/10.51244/IJRSI.2026.1307000395

Received: 08 August 2026; Accepted: 14 August 2026; Published: 20 August 2026

ABSTRACT

A firewall is designed to filter traffic, block unauthorized access, and allow authorized access. Modern network hardware contains vulnerabilities that can be exploited by attackers to compromise network security. The aim of this project is to propose an efficient machine learning model to analyze the vulnerabilities in firewall logs. The project implemented decision tree classifier on a NSL-KDD dataset pulled from Kaggle ML repository. This dataset contains 127,000+ instances of vulnerabilities. In order to build a web app, the input features of the dataset were reduced from 41 to 5 within the Streamlit interface to enhance prediction speed, reduce resource consumption etc. The following performance metrics were obtained for the selected model: Accuracy: 99.31%; Precision: 98.98%; Recall: 99.54% and F1-Score: 99.26%. The inclusion of quantitative metrics, visual evaluation and a real time web application prototype that helps bridge academic exploration and industrial relevance serves as a contribution to knowledge in firewall detection system. The outcome of the research proves that all these measures of accuracy show that the decision tree model is precise, reliable and suitable for practical implementation in analysis of firewall logs in order to detect the existing vulnerabilities for system engineers. There are some distinct benefits of applying machine learning to the problem of firewall vulnerability analysis, hence, this study has been able to contribute to the field of firewall and intrusion detection system in modern hardware system.

Keywords: Firewall, Intrusion detection, Machine Learning, Decision Tree Classifier, NSL-KDD Dataset

INTRODUCTION

The importance of network security cannot be denied as it has become an essential part of the data and information network management field, especially while ensuring the safety of sensitive and confidential information (Sunyaev & Sunyaev, 2020). No matter what size the business is or in which field it operates, businesses are dependent on the use of digital media to store, process, and transmit their data, which includes valuable information such as financial records, customer information, and intellectual property, all of which constitute the most valuable information of any organization (Bagale et al., 2021; Stergiou et al., 2018). A firewall is a network device that serves as security for a network. It is placed at the entrance of a network to filter outgoing or incoming network traffic based on configured commands. A firewall is one of the most critical components of the network system because it serves as a barrier between the internal trusted network and the external untrusted network, for example, the Internet. The firewall can be either hardware-based or

ISSN No. 2321-2705 | DOI: 10.51244/IJRSI |Volume XIII Issue VII July 2026

software-based. The Intrusion Detection System (IDS) is intended to analyze the network traffic to detect any attempts of unauthorized access or use and possible malicious activities. It is an essential element of the network security system. The primary goals of an IDS are to detect anomalies, identify known threats, and alert administrators (Sunyaev & Sunyaev, 2020). As a consequence of technological improvements that have been made in recent times, we have become highly dependent on global networks while performing our social, business and educational tasks. With the exponential increase in the usage of computer networks, several security problems in the Internet and computer systems have arisen. Therefore, it is vital to secure Internet-enabled systems from any possible threats so as to guarantee their availability and integrity (Anwar et. al., 2017). Machine Learning (ML) is one of the emerging technologies that can help boost different facets of cybersecurity. For example, machine learning can be used for enhancing the process of vulnerability assessment. Machine learning algorithms are capable of handling large amounts of data and making predictions with an unprecedented level of accuracy and efficiency (Ogundairo and Broklyn, 2024).

Statement of the Problem

Modern network hardware contains vulnerabilities that can be exploited by attackers to compromise network security. However, there is a lack of understanding on the characteristics of these vulnerabilities. Modern network hardware is often inadequately configured, leading to security vulnerabilities that can be exploited by attackers. Manual processes or static scanning tools are used to identify and classify the vulnerabilities. However, manual processes or static scanning tools may not keep pace with the rapid evolution of vulnerabilities. The effectiveness of machine learning is underexplored in the area of vulnerabilities in modern network hardware.

Aim and Objectives of the Study

The aim of this project is to propose an efficient machine learning model to analyze the vulnerabilities in firewall logs. The objectives are:

i. Collect data to help understand the vulnerabilities in modern network hardware.
ii. Implement machine learning on (i)
iii. Evaluate (ii) using accuracy, f1-score, recall and precision. This helps to evaluate the effectiveness of ML.

LITERATURE REVIEW

This section provides an overview on some of the related studies conducted by other researchers in the field of firewalls and IDS which is relevant to this current research:

Khraisat et al (2019) provides a taxonomy of contemporary intrusion detection systems, an extensive review of some prominent research studies, and a discussion of the datasets used to evaluate their performance. In addition, Khraisat et al describe various evasive tactics employed by malicious entities and the future research directions to mitigate such evasions. In the research by Shaheed and Kurdy (2021), the researchers focused on identification of web attacks in web application firewall by use of machine learning (Naive Bayes, logistic regression, decision tree, and support vector machine) and feature engineering of five features describing all aspects of HTTP requests. Features extracted include length of request, percentage of allowed characters, percentage of special characters, and attack weight. The features were calculated for four different sets of data including CSIC 2010, HTTPParams 2015, Hybrid dataset (CSIC 2010 and HTTPParams) and real logs of the attacked web server. The dataset included source IP, source port, flow duration, total number of forward packets and so forth. The performance of the proposed model was tested for efficiency. It achieved classification accuracy of 99.6% with datasets used in research studies in this field and 98.8% with datasets of real web servers.

PROPOSED METHODOLOGY

The study employs an experimental approach. A typical machine learning approach was followed in this project. This project adopted decision tree algorithm, a supervised machine learning models on the firewall

ISSN No. 2321-2705 | DOI: 10.51244/IJRSI |Volume XIII Issue VII July 2026

dataset. In this project, analytical approach will be employed, hence dataset will be collected, the dataset will be preprocessed to remove missing values, by utilizing the NSL-KDD dataset, a standard dataset for intrusion detection studies, this research emulates real-world network attack situations and assesses the model's proficiency in identifying patterns linked to security breaches while feature extraction and selection will be employed, while Decision Tree algorithm will be utilized on the trained and tested set of features and prediction will follow to classify the incoming data packets as either permitted or forbidden. Through exploratory analysis, we analyzed the dataset in detail and identified possible predictors of firewall vulnerabilities. Through various visualization techniques, we observed the separation between the incoming data packets as either permitted or forbidden. The architecture of the proposed model is represented below.

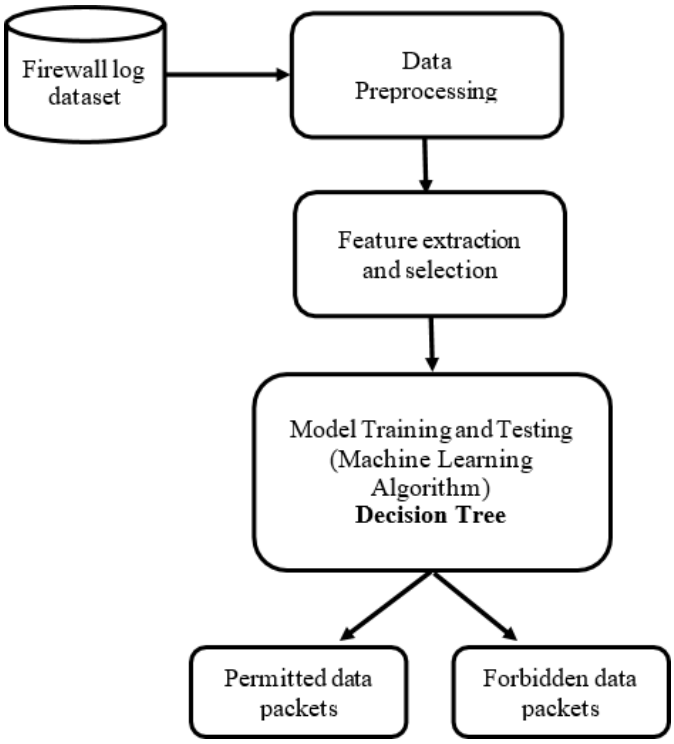

Figure 1: Flow diagram of the methodology

Data Collection: Collection of data involves gathering all relevant firewall vulnerabilities data from a source, ensuring that the data is comprehensive. Hence, the NSL-KDD was obtained from Kaggle.

After collection, the next thing is data preprocessing, the data undergo preprocessing steps including: Normalization: All numerical values were normalized to a similar scale to ensure that the machine learning model can process them effectively. Handling Missing Values: Missing or incomplete transaction details were handled by mean method or removal, depending on the data's impact on the model's training process.

The features needed for training are extracted and selected. Feature selection serves as an effective optimization technique in the realm of machine learning. By condensing the NSL-KDD dataset from 41 features down to 5 carefully selected ones, the system enhances its speed, simplicity, and efficiency. Reducing the number of input features from 41 to 5 in the NSL-KDD-based intrusion detection system provides numerous significant benefits, particularly regarding model performance, user experience, and application efficiency within the Streamlit interface.

The 41 features in the NSL-KDD dataset were filtered out to only five features using the Filter Method of feature selection. The Filter Method assesses the relevance of each individual feature by applying statistical measures independently of the Decision Tree classifier. In this approach, features are first evaluated based on their statistical association with the target class. Statistical methods such as variance analysis, correlation analysis, and univariate statistics was applied to filter out the features with discriminatory features that have statistical association with the target class. Features that are highly variant or contain discriminatory information are kept.

ISSN No. 2321-2705 | DOI: 10.51244/IJRSI |Volume XIII Issue VII July 2026

Based on the above feature selection, the five most important features were obtained: protocol_type, service, flag, src_bytes, and dst_bytes. Protocol_type is the protocol used in network connections, service is the service used in the network connection, flag refers to the status of network connection, while src_bytes and dst_bytes refer to the amount of data in bytes that are sent from the source address and received at the destination address. Therefore, the feature space was reduced from 41 features to 5 features. Features extraction is done as follows:

```
selected_features = ["protocol_type", "service", "flag", "src_bytes", "dst_bytes"]

X = data[selected_features].copy()
```

Model Training: The models will be initially trained using the data, which provides a controlled environment where vulnerabilities are predefined. After training, the models will be fine-tuned and tested. Finally, a prediction will be created on the firewall vulnerabilities.

Procedure: Following the feature selection process, the dataset was split into two parts – training set and testing set with an 80:20 train-test split. About 80% of the data was allocated for training of the Decision Tree classifier, in order for the model to learn how the patterns of the selected network features relate to their classes of intrusions. The rest 20% of the data was allocated for testing of the algorithm and was not involved in the training of the model. Testing dataset was used in order to determine the ability of the model to properly classify network connections which were not seen before.

The NSL-KDD dataset is an open-source dataset that has been used in this study, and it can be accessed on Kaggle website. The dataset consists of different types of generated network traffic classified as normal or attack. Each observation in the NSL-KDD dataset consists of 43 attributes, where: 41 attributes are input attributes, 1 attribute is the target class attribute and 1 attribute is difficulty level (for certain assessments only).

	duration	protocol_type	...	attack	level
0	0	udp	...	normal	15
1	0	tcp	...	neptune	19
2	0	tcp	...	normal	21
3	0	tcp	...	normal	21
4	0	tcp	...	neptune	21
...	...	...	...	...	...
125967	0	tcp	...	neptune	20
125968	8	udp	...	normal	21
125969	0	tcp	...	normal	18
125970	0	tcp	...	neptune	20
125971	0	tcp	...	normal	21

```
RangeIndex: 125972 entries, 0 to 125971
Data columns (total 42 columns):
 #   Column                       Non-Null Count   Dtype
---  ------                       --------------   -----
 0   duration                     125972 non-null  int64
 1   protocol_type                125972 non-null  object
 2   service                      125972 non-null  object
 3   flag                         125972 non-null  object
 4   src_bytes                    125972 non-null  int64
 5   dst_bytes                    125972 non-null  int64
 6   land                         125972 non-null  int64
 7   wrong_fragment               125972 non-null  int64
 8   urgent                       125972 non-null  int64
 9   hot                          125972 non-null  int64
 10  num_failed_logins            125972 non-null  int64
```

Figure 2: Sample of NSL-KDD Dataset (left column) (Kaggle, 2025)

Figure 3: Screenshot of NSL-KDD dataset using describe() function (right column)

In the above screenshots, the .info() method has been used to analyze the structure of the data set. The information it provides includes the number of entries, whether the feature is categorical or numeric in nature, as well as identifying null values within the data set. The analysis proved that the data set was complete and properly structured.

Algorithm

The learning process in case of decision trees consists of a number of steps through which data is segmented into homogeneous sets, see Figure 1. The root node, which is the first node in the decision tree, is a node that represents the whole dataset. The algorithm selects the variable and the threshold that results in the optimal

ISSN No. 2321-2705 | DOI: 10.51244/IJRSI |Volume XIII Issue VII July 2026

split according to some criteria. The splitting process is repeated recursively in each subset of data. The process repeats itself until some termination conditions are met; usually they include purity of nodes (all points in a node belong to the same class) or predetermined maximum tree depth. Leaf or terminal nodes, which are nodes where decision tree stops, correspond to outcomes or class labels (Mienye & Jere, 2016). The decision tree approach is a very efficient statistical tool used in classification, prediction, interpretation and data manipulation that has a number of possible applications in computing (Song & Lu, 2015).

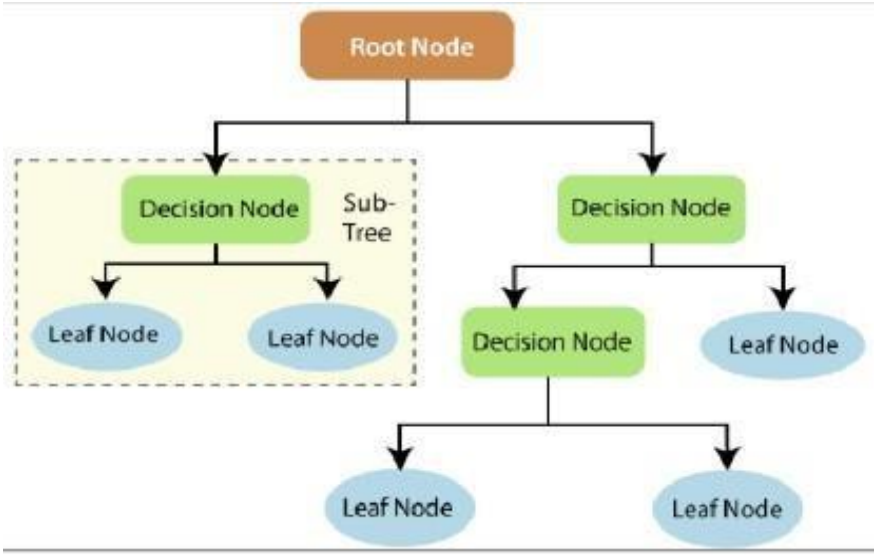

Figure 4: Working of Decision Tree Algorithm (Saha & Manickavasagan, 2021)

RESULT AND DISCUSSION

In our current project, we tried to address the very important need of introducing an effective machine learning algorithm to assess vulnerabilities in firewall logs. The use of decision tree machine learning algorithms in AI-based firewall detection systems is very important from the point of view of network security, especially while working on the open-source NSL-KDD dataset. The use of machine learning algorithms has made our firewall systems more efficient and secure.

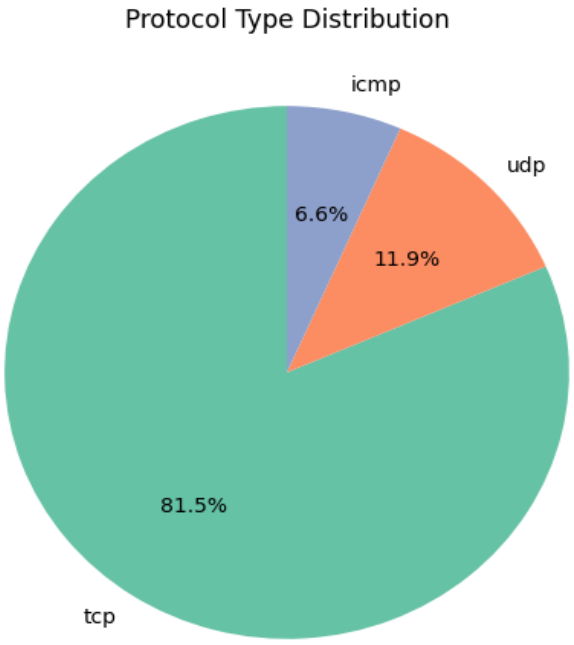

Figure 5: Protocol Type Distribution of NSL-KDD dataset

ISSN No. 2321-2705 | DOI: 10.51244/IJRSI |Volume XIII Issue VII July 2026

The protocol type attribute in the figure above comprises groups such as TCP, UDP, and ICMP, denoting the various communication protocols employed in the network connection process. Comprehending the protocol distribution assists in identifying which of the protocols are mostly associated with normal or abnormal behavior. From the pie chart depicting the distribution of the different protocols for the NSL-KDD data set, it is evident that the most prevalent protocol is TCP with 81.5% usage, followed by UDP at 11.9% and ICMP at 6.6%.

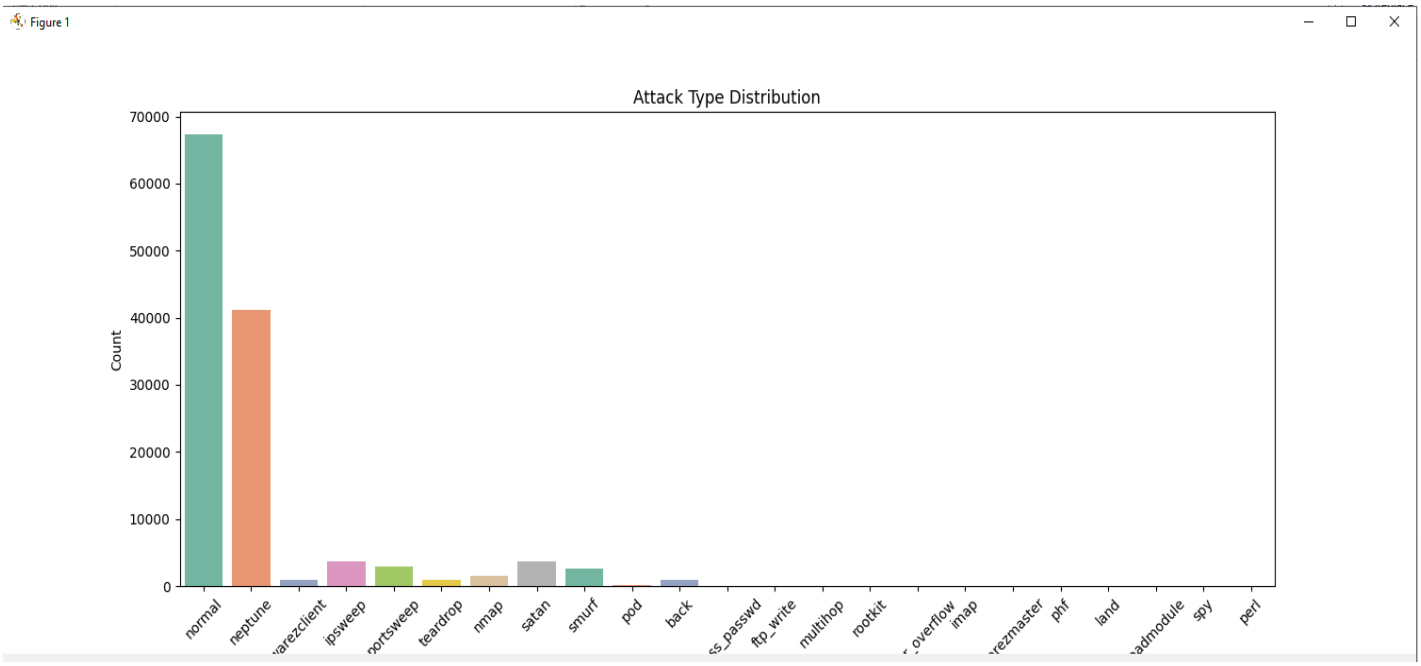

Figure 6: Attack Type Distribution of NSL-KDD dataset

Furthermore, the above diagram has also looked into the type of attacks on the basis of frequency of the class labels in the dataset. The classes are divided into five main categories, which include Normal, DoS (Denial of Service), Probe, R2L (Remote to Local), and U2R (User to Root).

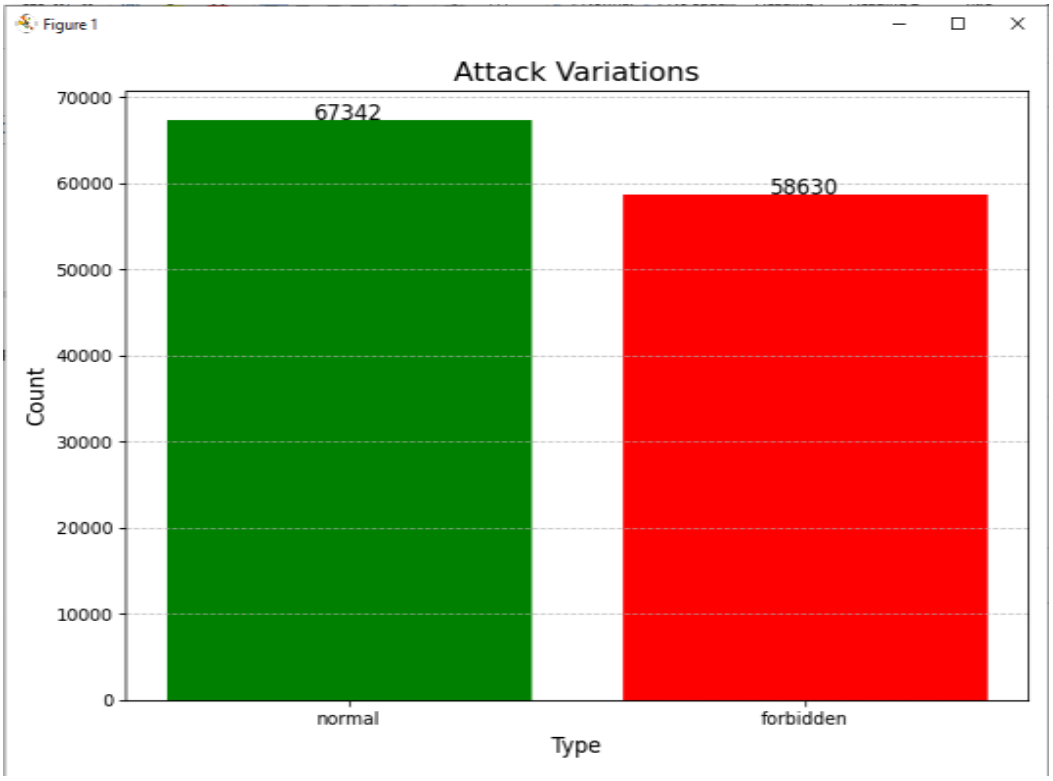

Figure 7: Attack Variations of NSL-KDD dataset

ISSN No. 2321-2705 | DOI: 10.51244/IJRSI |Volume XIII Issue VII July 2026

From the graph presented above, the attack variations table provides a binary classification of the NSL-KDD data set into two classes: normal (67,342 observations) and forbidden (58,630 observations). The presentation simplifies the visualization by combining all types of attacks into these two categories.

Model evaluation metrics

As such, on the chosen dataset, the performance of the chosen algorithm was evaluated and validated using the following criteria: accuracy, F1-score, precision and recall as shown in the table below:

Table 1: Result of the evaluation metrics

	Decision Tree Values
Accuracy	99.31%
Precision	98.98%
Recall	99.54%
F1 Score	99.26%

GUI implementation

This intuitive navigation and real-time interaction make the application accessible for users with varying technical backgrounds, enabling them to explore machine learning–driven intrusion detection in a practical and user-friendly manner. The Multipage web interface which consists of prediction page, about page, and developer page is designed using Streamlit for vulnerability simulation and prediction.

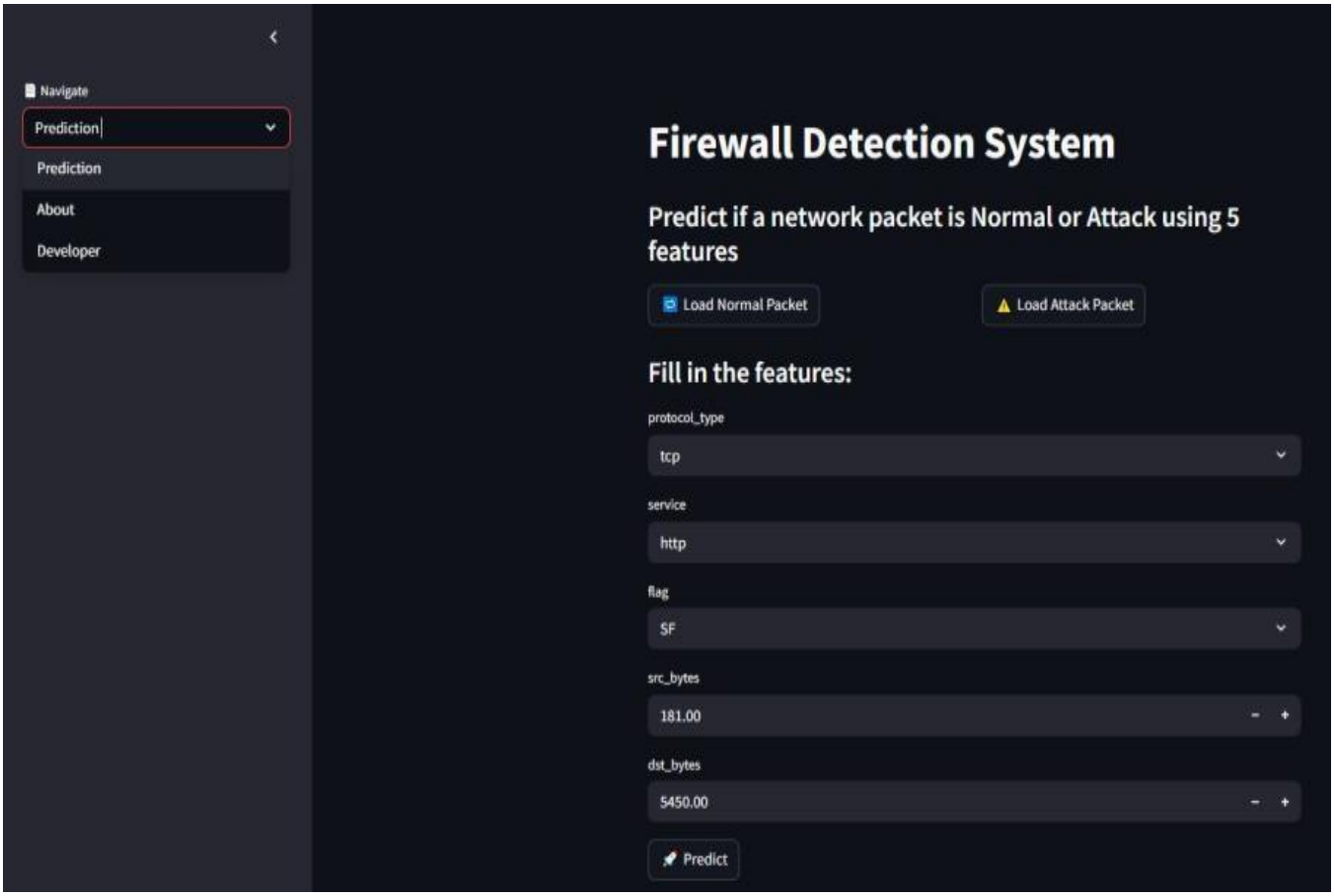

Figure 8: Screenshot of the Application

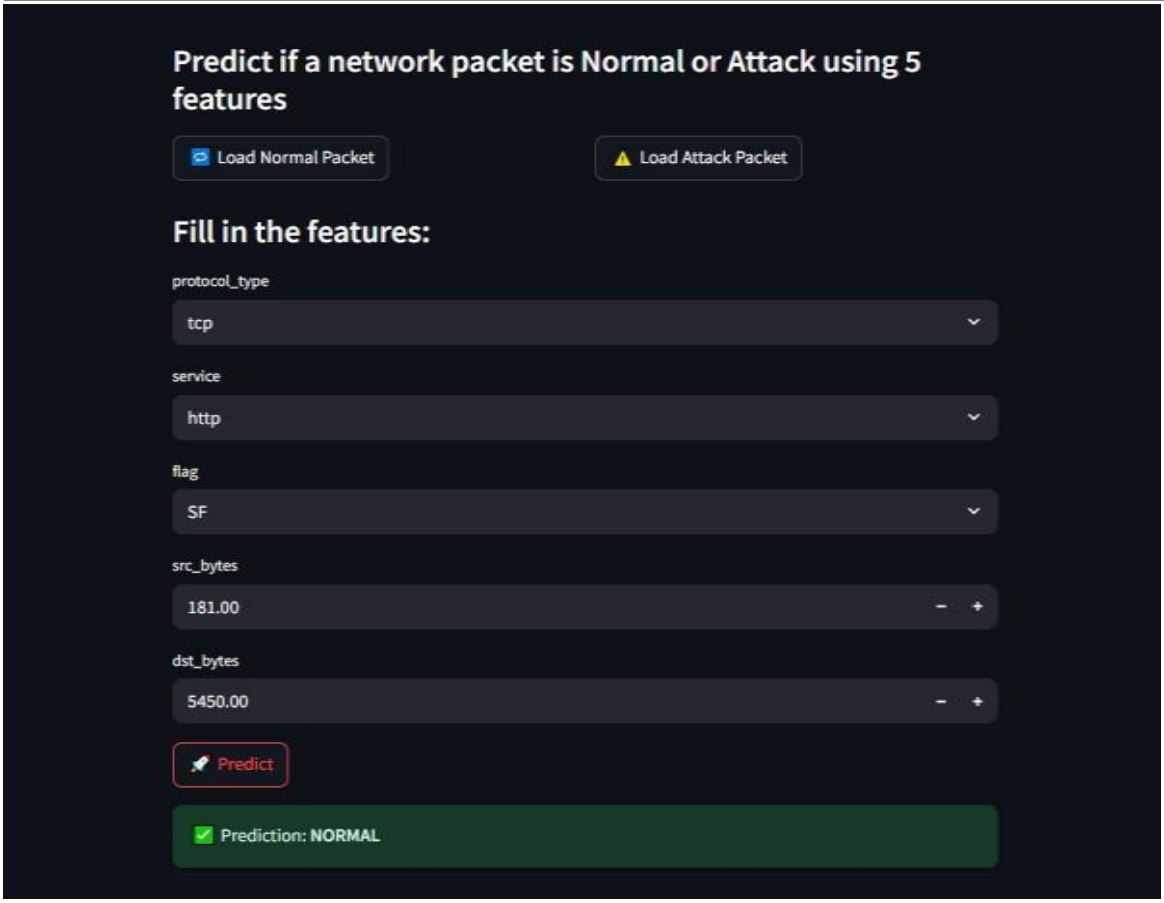

Figure 9: Screenshot of the Firewall Detection System testing Normal Packet (left column)

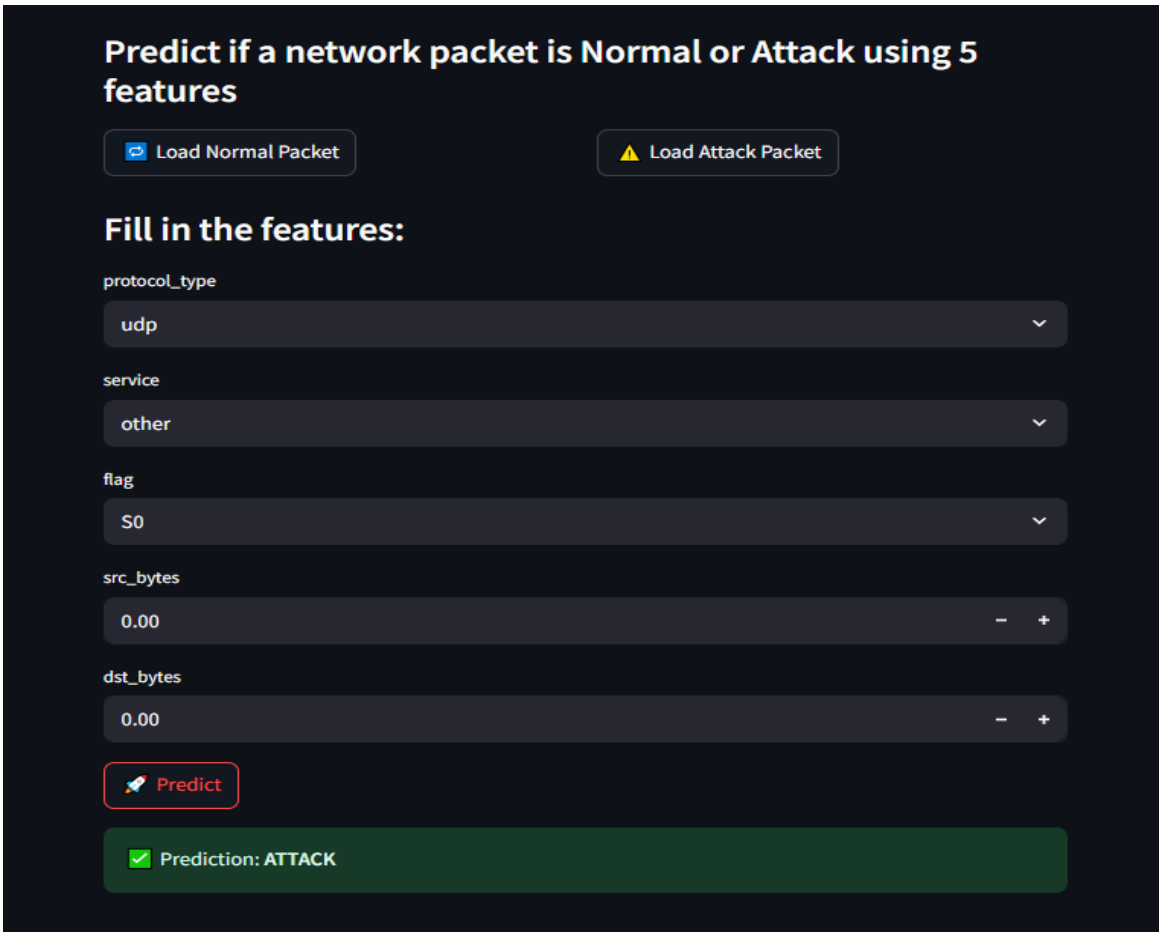

Figure 10: Screenshot of the Firewall Detection System testing Forbidden Packet (right column)

CONCLUSION

This study has been able to contribute to the field of firewall and intrusion detection system in modern hardware system. The output of the model for prediction clearly displays "Prediction: NORMAL" in green text, which means that this particular combination of features corresponds to patterns observed in legitimate network traffic. This kind of classification is a good example of how the model recognizes the properties of regular HTTP traffic on the web using the TCP protocol, through standard connection flags and data sizes. In addition, the following screen shot represents the ability of the firewall system to detect malicious network traffic by analyzing the same five features observed in the previous example for normal packets. The interface illustrates different combinations of features, which lead to an "ATTACK" classification, thus showing the practical application of patterns obtained during the analysis of the NSL-KDD dataset.

REFERENCES

1. Bagale, G. S., Vandadi, V. R., Singh, D., Sharma, D. K., Garlapati, D. V. K., Bommisetti, R. K., & Sengan, S. (2021). Small and medium-sized enterprises' contribution in digital technology. Annals of Operations Research, 1-24.
2. Khraisat, A., Gondal, I., Vamplew, P., and Kamruzzaman, J. (2019). Survey of intrusion detection systems: techniques, datasets and challenges. https://doi.org/10.1186/s42400-019-0038-7.
3. Mienye, I.D., and Jere, N. (2016). ASurvey of Decision Trees: Concepts, Algorithms, and Applications. https://doi.org/10.1109/ACCESS.2024.3416838.
4. Ogundairo, O., and Broklyn, P. (2024). Automated Vulnerability Assessment Using Machine Learning.
5. Saha, D., and Manickavasagan, A. (2021). Machine learning techniques for analysis of hyperspectral images to determine quality of food products: A review. https://doi.org/10.1016/j.crfs.2021.01.002.
6. Shaheed, A., and Kurdy, M.H.D. (2021). Web Application Firewall Using Machine Learning and Features Engineering. https://doi.org/10.1155/2022/5280158.
7. Song, Y., and Lu, Y. (2015). Decision tree methods: applications for classification and prediction. http://dx.doi.org/10.11919/j.issn.1002-0829.215044.
8. Stergiou, C., Psannis, K. E., Gupta, B. B., & Ishibashi, Y. (2018). Security, privacy & efficiency of sustainable cloud computing for big data & IoT. Sustainable Computing: Informatics and Systems, 19, 174-184.
9. Sunyaev, A., & Sunyaev, A. (2020). Internet computing (pp. 237-264). New York, NY, USA: Springer International Publishing.