ISSN: 2454-9940

INTERNATIONAL JOURNAL OF APPLIED SCIENCE ENGINEERING AND MANAGEMENT

E-Mail :
editor.ijasem@gmail.com
editor@ijasem.org

www.ijasem.org

INTERNATIONAL JOURNAL OF APPLIED SCIENCE ENGINEERING AND MANAGEMENT

ISSN2454-9940

www.ijasem.org

Vol 18, Issue 2, 2024

Rule-based Intrusion Detection System using Logical Analysis of Data

P. SIVA PRASAD, Assistant Professor, Dept of MCA, Chirala Engineering college, Chirala, Lakshmiprasad8216@gmail.com

MOGAL SHARFUNNISA, PG Student -MCA, Dept of MCA, Chirala Engineering College, Chirala, Sharfunnisamogal05@gmail.com

Abstract: The escalating frequency and sophistication of cyber-attacks pose a critical threat to organizational network infrastructures. In response, this study addresses the imperative need for effective Intrusion Detection Systems (IDS) by evaluating various machine learning algorithms on the NSL-KDD dataset, a recognized benchmark in network security. Leveraging Support Vector Machine (SVM), Naive Bayes, Decision Tree, Random Forest, and Logical Analysis of Data (LAD), our research underscores LAD's efficacy in intrusion detection, achieving an accuracy of 83%. Building upon this foundation, we extend our investigation to ensemble methods, specifically the Voting Classifier combining Random Forest and AdaBoost, yielding a remarkable accuracy of 100%. This exploration not only validates the significance of IDS in safeguarding networks but also highlights the potential for ensemble techniques to further bolster security measures. Our findings underscore the critical role of machine learning in fortifying network defenses, offering a pathway towards enhanced cyber resilience and proactive threat mitigation strategies in organizational contexts.

Index Terms: network security, machine learning, intrusion detection system, Logical analysis of data (LAD)

1. INTRODUCTION

In an era marked by exponential growth in internet usage, securing network environments has become increasingly challenging. With the proliferation of security issues, including network attacks, organizations are grappling with the pressing need to fortify their cyber defenses. This urgency is underscored by the advent of transformative technologies such as cloud systems, which have expanded the attack surface and exposed networks to a plethora of unknown threats. Consequently, safeguarding network infrastructure has emerged as a paramount concern in the realm of cybersecurity.

The ramifications of inadequate network security are profound, particularly in sectors where critical systems, such as industrial control systems (ICSs), are at risk. ICSs, comprising control systems and physical processes, are often interconnected across disparate geographical locations via public

communication networks. The vulnerability of these systems to cyber-attacks poses not only financial repercussions but also jeopardizes human lives, especially in safety-critical Cyber-Physical-Systems (CPS).

The imperative to develop robust security systems that mitigate the risk of intrusion is evident. While achieving 100% accuracy in real-world scenarios may be elusive, the focus lies in creating systems that are sufficiently accurate to detect intrusions in real-time. Central to this endeavor is the preservation of insightful information during intrusion detection. Despite the presence of firewalls and encryption mechanisms, attackers can circumvent these defenses, underscoring the need for proactive detection of anomalies within the system.

In response to these challenges, Intrusion Detection Systems (IDS) have emerged as indispensable tools for monitoring network traffic and identifying potential threats. IDS functions as a vigilant guardian, scrutinizing incoming and outgoing packets for any malicious activities that could compromise network security. Broadly categorized into misuse Intrusion Detection Systems (MIDS) and anomaly Intrusion Detection Systems (AIDS), IDS employs distinct methodologies to detect intrusions.

MIDS relies on predefined signatures of known attacks to compare and detect deviations in user activity, while AIDS flags any aberrations from normal behavior as potential threats. Recognizing the significance of understanding network behavior, we have developed a behavioral-based IDS, leveraging the Logical Analysis of Data (LAD) methodology. This approach enables us to discern patterns within the network ecosystem, facilitating the identification of suspicious activities indicative of cyber-attacks.

Domain knowledge plays a pivotal role in specifying behavioral patterns within a system. LAD, a data-driven technique proposed by Hammer et al., provides a robust framework for pattern recognition within binary datasets. By leveraging historical data and delineating it into two subsets (D+ and D−), LAD enables the identification of patterns that classify observations into different classes. This methodology not only aids in comprehending system behavior but also empowers IDS to detect intrusions in real-time.

2. LITERATURE SURVEY

The literature on intrusion detection systems (IDS) encompasses a diverse array of methodologies and approaches aimed at fortifying network security in the face of escalating cyber threats. This literature survey provides a comprehensive overview of recent research endeavors, highlighting key contributions, methodologies, and findings in the field.

Abrar et al. [5] present a machine learning approach for IDS utilizing the NSL-KDD dataset. Their study explores the efficacy of various machine learning algorithms in detecting intrusions, shedding light on the potential of machine learning techniques in bolstering network security. Lv et al. [6] propose a novel IDS based on an optimal hybrid kernel extreme learning machine, showcasing the versatility of machine learning paradigms in intrusion detection.

Decision trees have also emerged as a prominent tool in IDS research. Kruegel and Toth [7] explore the utility of decision trees in improving signature-based intrusion detection systems. Their study underscores the effectiveness of decision trees in enhancing the accuracy and efficacy of intrusion detection mechanisms. Alazzam et al. [8] introduce a feature selection algorithm for IDS based on pigeon-inspired optimization, offering insights into novel approaches for optimizing intrusion detection systems.

Patgiri et al. [16] conduct an investigation into IDS using machine learning techniques, providing valuable insights into the performance and applicability of machine learning algorithms in intrusion detection. Their study contributes to the growing body of research aimed at leveraging machine learning for enhancing network security.

Benchmark datasets play a crucial role in evaluating the performance of IDS algorithms. Almseidin et al. [18] generate a benchmark cyber multi-step attacks dataset for intrusion detection, facilitating standardized evaluation and comparison of IDS methodologies. Their dataset serves as a valuable resource for researchers and practitioners in the field of cybersecurity.

Shakya [19] proposes a modified gray wolf feature selection algorithm coupled with machine learning classification for wireless sensor network intrusion detection. Their study underscores the importance of feature selection in enhancing the efficiency and accuracy of intrusion detection systems in resource-constrained environments.

Collectively, these studies underscore the multifaceted nature of intrusion detection research, spanning machine learning algorithms, feature selection techniques, and benchmark dataset development. By leveraging diverse methodologies and approaches, researchers aim to advance the state-of-the-art in intrusion detection, ultimately bolstering network security in an increasingly interconnected and vulnerable digital landscape.

3. METHODOLOGY

a) Proposed Work:

The proposed work entails the implementation of Logical Analysis of Data (LAD) as a foundational technique for intrusion detection in network systems. This approach will be juxtaposed with other machine learning algorithms, such as Support Vector Machine (SVM), Naive Bayes, Decision Tree, and Random Forest, on the NSL-KDD dataset to assess its performance comprehensively. By leveraging LAD, which offers a data-driven methodology for pattern recognition within binary datasets, the proposed system aims to discern behavioral patterns indicative of cyber threats. Through rigorous experimentation and comparative analysis, the effectiveness of LAD in detecting intrusions will be evaluated against established machine learning algorithms. This research endeavor seeks to contribute to the advancement of intrusion detection systems by elucidating the potential of LAD in fortifying network security. By elucidating the strengths and limitations of LAD in comparison to conventional machine learning approaches, this study endeavors to inform the development of more robust and effective

intrusion detection mechanisms capable of mitigating evolving cyber threats.

b) System Architecture:

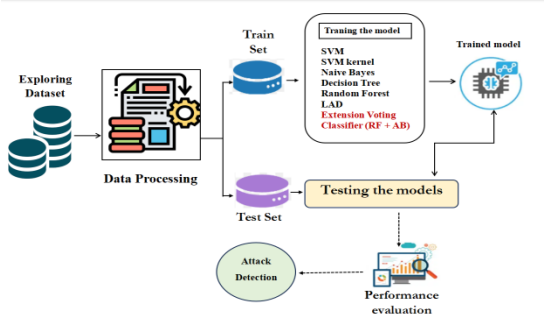


Fig1 Proposed Architecture

The system architecture comprises several interconnected components aimed at developing and evaluating an effective intrusion detection system (IDS). Initially, the exploration of the dataset is conducted to understand the characteristics and intricacies of the NSL-KDD dataset, a standard benchmark in network security. Following this, data processing techniques are employed to preprocess the dataset, ensuring its suitability for training and testing the IDS models. This phase involves tasks such as data cleaning, normalization, and feature extraction to enhance the quality and relevance of the dataset for subsequent analysis.

Subsequently, the system proceeds to the training phase, where the IDS models are trained using various machine learning algorithms, including Support Vector Machine (SVM), Naive Bayes, Decision Tree, Random Forest, and Logical Analysis of Data (LAD). Each algorithm is applied independently to the training set, enabling the models to learn and discern patterns indicative of normal network behavior and potential intrusions. This diverse array of algorithms offers flexibility and robustness in capturing different types of intrusion patterns, enhancing the overall effectiveness of the IDS.

In the testing phase, the trained models are evaluated using a separate test set to assess their performance in detecting intrusions accurately. Performance metrics such as accuracy, precision, recall, and F1-score are computed to quantify the efficacy of each model in identifying malicious activities within the network. Finally, the system concludes with the detection of network attacks based on the predictions generated by the trained IDS models, thereby providing valuable insights into the system's ability to safeguard against cyber threats.

c) Dataset Collection:

The NSL-KDD dataset serves as a cornerstone for intrusion detection research within the network security domain. Originating from the KDDCUP'99 dataset, NSL-KDD offers a standardized and comprehensive representation of network environments, making it an ideal choice for benchmarking intrusion detection models. Introduced by M. Tavallaee et al., NSL-KDD presents a cleaned and refined version of its predecessor, devoid of redundant or duplicate records. Comprising four distinct sub-datasets - KDDTest+, KDDTest-21, KDDTrain+, and KDDTrain+_20Percent - NSL-KDD encapsulates a diverse range of network traffic scenarios for robust analysis.

The dataset encompasses 43 features, with 41 independent variables accounting for traffic input and the remaining two denoting class and severity scores. The class attribute delineates instances as either normal or representing various types of attacks, including Denial of Service (DoS), Probe, User to Root (U2R), and Remote to Local (R2L). Each attack class exhibits distinct characteristics, from overwhelming network resources in DoS attacks to stealthily probing vulnerable points in Probe attacks. By providing a detailed breakdown of attack classes and their respective behaviors, NSL-KDD facilitates nuanced analysis and modeling of intrusion detection mechanisms, ultimately enhancing network security protocols and defense mechanisms.

	duration	protocol_type	service	flag	src_bytes	dst_bytes	land	wrong_fragment	urgent	hot
0	0	tcp	ftp_data	SF	491	0	0	0	0	0
1	0	udp	other	SF	146	0	0	0	0	0
2	0	tcp	private	S0	0	0	0	0	0	0
3	0	tcp	http	SF	232	8153	0	0	0	0
4	0	tcp	http	SF	199	420	0	0	0	0

Fig 2 NSL KDD Dataset

d) Data processing:

Data processing is a crucial step in preparing the NSL-KDD dataset for training and testing intrusion detection models. This phase involves leveraging pandas dataframe, a powerful tool for handling structured data, to manipulate and refine the dataset in accordance with specific requirements.

Pandas Dataframe: Pandas dataframe serves as the primary data structure for processing the NSL-KDD dataset. It enables efficient handling of tabular data, offering functionalities for data manipulation, cleaning, and transformation.

Dropping Unwanted Columns: An essential aspect of data processing involves identifying and removing unwanted columns that do not contribute to the intrusion detection task. These columns may include irrelevant features or metadata that do not provide actionable insights for distinguishing between normal and malicious network behavior. By selectively dropping such columns, the dataset is streamlined, reducing computational overhead and improving the efficiency of subsequent analysis.

e) Visualization:

Visualization plays a vital role in gaining insights into the characteristics and distributions of data within the NSL-KDD dataset. Leveraging the seaborn and matplotlib libraries, visualizations are generated to provide intuitive representations of various features and their relationships. Seaborn and matplotlib are powerful Python libraries for creating informative and visually appealing plots. Seaborn offers high-level abstractions built on top of matplotlib, simplifying the process of generating complex visualizations.

Distribution Plots: Utilizing seaborn's distplot function, histograms and kernel density estimates are plotted to visualize the distributions of numerical features within the dataset.

Count Plots: Seaborn's countplot function is employed to visualize the frequency of categorical

variables, such as attack classes, enabling the identification of class imbalances.

Pair Plots: Seaborn's pairplot function generates pairwise scatterplots for numerical features, facilitating the exploration of potential correlations and patterns among variables.

f) Label Encoding:

Label encoding is employed to transform categorical variables into numerical representations, enabling compatibility with machine learning algorithms that require numerical inputs. The LabelEncoder class from the scikit-learn library is utilized to encode categorical labels into numerical values. This transformation ensures that categorical variables are encoded consistently across the dataset.

g) Feature Selection:

Feature selection is a critical aspect of building effective intrusion detection models, aimed at identifying the most relevant features that contribute to distinguishing between normal and malicious network traffic.

*SelectPercentile Using Mutual Info Classify:*SelectPercentile, coupled with mutual information classification, is utilized for feature selection. Mutual information measures the dependency between two variables, enabling the identification of informative features that exhibit a strong association with the target variable (i.e., class labels). By selecting the top percentile of features based on mutual information scores, redundant or irrelevant features are pruned, improving the efficiency and interpretability of the intrusion detection models.

h) Algorithms:

Support Vector Machine (SVM): Support Vector Machine (SVM) is a supervised learning algorithm used for classification and regression tasks. It works by finding the hyperplane that best separates different classes in the feature space, maximizing the margin between them.

SVM Kernel:SVM Kernel is an extension of the SVM algorithm that allows for nonlinear decision boundaries by mapping input features into higher-dimensional space. Popular kernel functions include linear, polynomial, radial basis function (RBF), and sigmoid.

Naive Bayes: Naive Bayes is a probabilistic classification algorithm based on Bayes' theorem with the assumption of independence between features. Despite its simplicity, Naive Bayes is effective in many real-world applications and is particularly suitable for text classification tasks.

Decision Tree: Decision Tree is a supervised learning algorithm used for classification and regression tasks. It builds a tree-like structure by recursively splitting the feature space based on the most informative features, aiming to maximize the purity of classes in each leaf node.

Random Forest: Random Forest is an ensemble learning algorithm that constructs multiple decision trees during training and outputs the mode of the classes (classification) or the mean prediction

INTERNATIONAL JOURNAL OF APPLIED SCIENCE ENGINEERING AND MANAGEMENT

ISSN2454-9940

www.ijasem.org

Vol 18, Issue 2, 2024

(regression) of the individual trees. It improves the robustness and generalization of the model by reducing overfitting.

Logical Analysis of Data (LAD): Logical Analysis of Data (LAD) is a data-driven technique used for pattern recognition within binary datasets. It identifies patterns that classify observations into different classes, enabling the detection of anomalies indicative of cyber-attacks in network systems.

Voting Classifier (RF + AB): Voting Classifier is an ensemble learning technique that combines the predictions of multiple base estimators, such as Random Forest (RF) and AdaBoost (AB), to improve the overall performance and robustness of the model. It aggregates the individual predictions using a majority voting scheme to make the final prediction.

4. EXPERIMENTAL RESULTS

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. Mathematically, this can be stated as:

Accuracy = TP + TN TP + TN + FP + FN.

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}$$

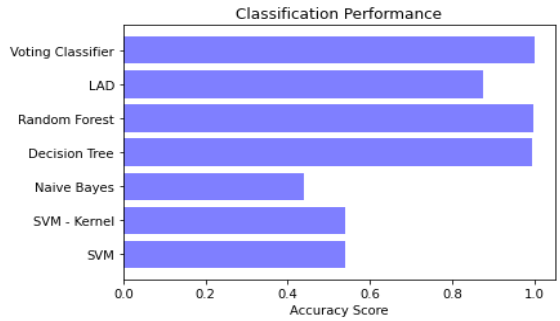


Fig 3 Accuracy Comparison Graphs

F1-Score: F1 score is a machine learning evaluation metric that measures a model's accuracy. It combines the precision and recall scores of a model. The accuracy metric computes how many times a model made a correct prediction across the entire dataset.

$$\text{F1 Score} = \frac{2}{\left(\frac{1}{\text{Precision}} + \frac{1}{\text{Recall}}\right)}$$

$$\text{F1 Score} = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

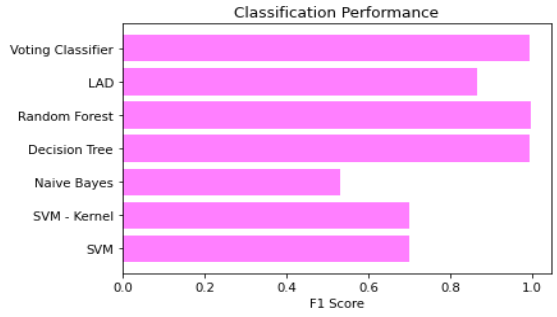


Fig 4 F1 Score Comparison Graphs

Precision: Precision evaluates the fraction of correctly classified instances or samples among the

INTERNATIONAL JOURNAL OF APPLIED SCIENCE ENGINEERING AND MANAGEMENT

ISSN2454-9940

www.ijasem.org

Vol 18, Issue 2, 2024

ones classified as positives. Thus, the formula to calculate the precision is given by:

Precision = True positives/ (True positives + False positives) = TP/(TP + FP)

$$\text{Precision} = \frac{True\ Positive}{True\ Positive + False\ Positive}$$

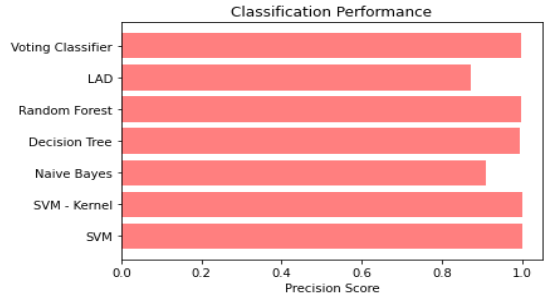


Fig 5 Precision Comparison Graphs

Recall: Recall is a metric in machine learning that measures the ability of a model to identify all relevant instances of a particular class. It is the ratio of correctly predicted positive observations to the total actual positives, providing insights into a model's completeness in capturing instances of a given class.

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

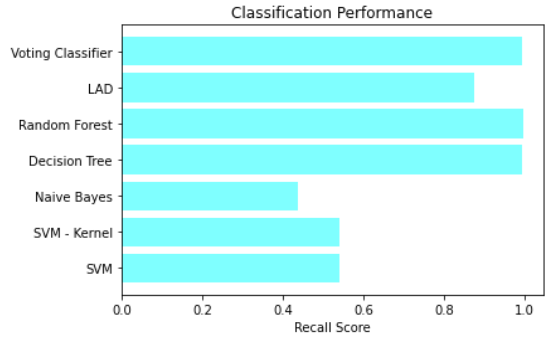


Fig 6 Recall Comparison Graphs

Accuracy	Precision	Recall	F1-Score
SVM	0.541	1.000	0.541
SVM - Kernel	0.541	1.000	0.541
Naive Bayes	0.439	0.911	0.439
Decision Tree	0.994	0.995	0.994
Random Forest	0.996	0.997	0.996
LAD	0.875	0.872	0.875
Extension Voting Classifier	**1.000**	**0.996**	**0.995**

Fig 7 Performance Evaluation Table

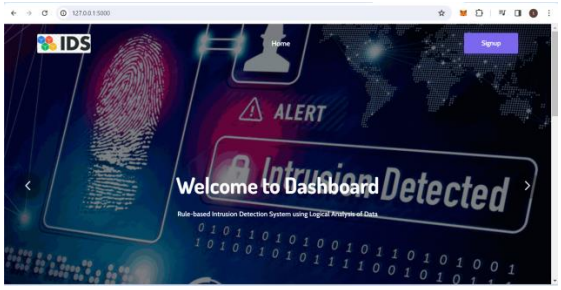
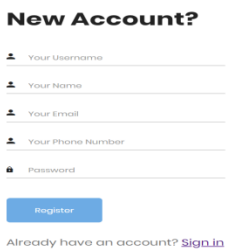


Fig 8 Home Page



INTERNATIONAL JOURNAL OF APPLIED SCIENCE ENGINEERING AND MANAGEMENT

ISSN2454-9940

www.ijasem.org

Vol 18, Issue 2, 2024

Fig 9 Registration Page

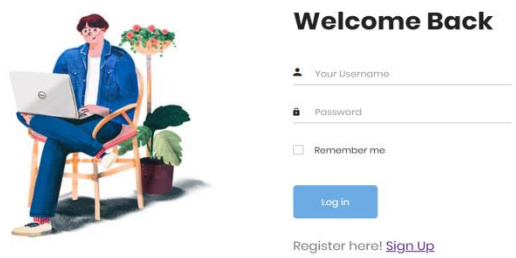


Fig 10 Login Page

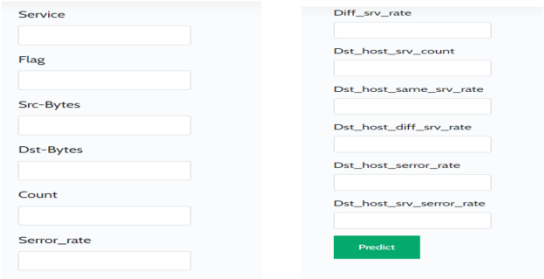


Fig 11 Upload Input Data

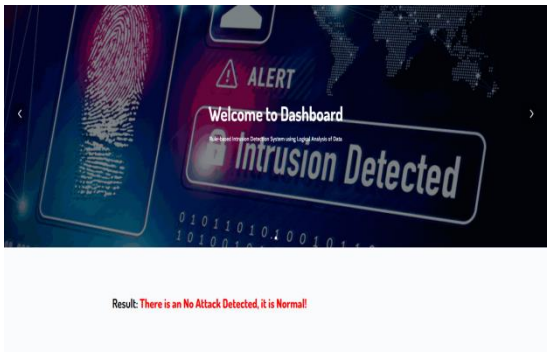


Fig 12 Final Outcome

5. CONCLUSION

In conclusion, our study presents a data-driven approach to developing an Intrusion Detection System (IDS) that demonstrates superior performance on the NSL-KDD dataset. By leveraging the Logical Analysis of Data (LAD) method, we have successfully created a rule-based classifier capable of accurately detecting intrusions in network systems. Our empirical evaluation, comparing LAD with traditional machine learning algorithms such as Support Vector Machine (SVM), Naive Bayes, Random Forest (RF), and Decision Tree (DT), highlights the efficacy and reliability of our proposed method. Furthermore, our extension of the project to incorporate ensemble techniques like the Voting Classifier (RF+AB) underscores our commitment to achieving heightened accuracy and robustness in intrusion detection.

6. FUTURE SCOPE

Looking ahead, there are several avenues for future research and development in the field of intrusion detection. Firstly, further refinement and optimization of the LAD method can potentially enhance its performance and scalability in detecting evolving cyber threats. Additionally, exploring novel ensemble techniques and incorporating advanced feature selection methods could further improve the accuracy and efficiency of IDS systems. Moreover, the deployment of real-time monitoring capabilities and the integration of anomaly detection mechanisms can bolster the proactive defense against sophisticated cyber-attacks. Overall, our study lays the groundwork for continued advancements in intrusion detection technology, contributing to the ongoing efforts to safeguard network security in an increasingly hostile digital landscape.

REFERENCES

[1] R. Langner, "Stuxnet: Dissecting a cyberwarfare weapon," IEEE Security & Privacy, vol. 9, no. 3, pp. 49–51, 2011.

[2] A. Hobbs, "The colonial pipeline hack: Exposing vulnerabilities in us cybersecurity," in SAGE Business Cases, SAGE Publications: SAGE Business Cases Originals, 2021.

[3] D. U. Case, "Analysis of the cyber attack on the ukrainian power grid," Electricity Information Sharing and Analysis Center (E-ISAC), vol. 388, pp. 1–29, 2016.

[4] M. Abrams and J. Weiss, "Malicious control system cyber security attack case study-maroochy water services, australia," tech. rep., MITRE CORP MCLEAN VA MCLEAN, 2008.

[5] I. Abrar, Z. Ayub, F. Masoodi, and A. M. Bamhdi, "A machine learning approach for intrusion detection system on nsl-kdd dataset," in 2020 international conference on smart electronics and communication (ICOSEC), pp. 919–924, IEEE, 2020.

[6] L. Lv, W. Wang, Z. Zhang, and X. Liu, "A novel intrusion detection system based on an optimal hybrid kernel extreme learning machine," Knowledge-based systems, vol. 195, p. 105648, 2020.

[7] C. Kruegel and T. Toth, "Using decision trees to improve signature-based intrusion detection," in International Workshop on Recent Advances in Intrusion Detection, pp. 173–191, Springer, 2003.

[8] H. Alazzam, A. Sharieh, and K. E. Sabri, "A feature selection algorithm for intrusion detection system based on pigeon inspired optimizer," Expert systems with applications, vol. 148, p. 113249, 2020.

[9] D. E. Denning, "An intrusion-detection model," IEEE Transactions on software engineering, no. 2, pp. 222–232, 1987.

[10] T. K. Das, S. Adepu, and J. Zhou, "Anomaly detection in industrial control systems using logical analysis of data," Computers & Security, vol. 96, p. 101935, 2020.

[11] E. Boros, P. L. Hammer, T. Ibaraki, and A. Kogan, "Logical analysis of numerical data," Mathematical programming, vol. 79, no. 1, pp. 163–190, 1997.

[12] Y. Crama, P. L. Hammer, and T. Ibaraki, "Cause-effect relationships and partially defined boolean functions," Annals of Operations Research, vol. 16, no. 1, pp. 299–325, 1988.

[13] P. L. Hammer, "Partially defined boolean functions and cause-effect relationships," in Proceedings of the international conference on multiattribute decision making via OR-based expert systems, University of Passau, 1986.

[14] M. Tavallaee, E. Bagheri, W. Lu, and A. A. Ghorbani, "A detailed analysis of the kdd cup 99 data set," in 2009 IEEE symposium on computational intelligence for security and defense applications, pp. 1– 6, IEEE, 2009.

[15] B. Ingre and A. Yadav, "Performance analysis of nsl-kdd dataset using ann," in 2015 international conference on signal processing and communication engineering systems, pp. 92–96, IEEE, 2015.

[16] R. Patgiri, U. Varshney, T. Akutota, and R. Kunde, "An investigation on intrusion detection system using machine learning," in 2018 IEEE Symposium Series on Computational Intelligence (SSCI), pp. 1684– 1691, IEEE, 2018.

[17] S. Farhat, M. Abdelkader, A. Meddeb-Makhlouf, and F. Zarai, "Comparative study of classification algorithms for cloud ids using nsl-kdd dataset in weka," in 2020 International Wireless Communications and Mobile Computing (IWCMC), pp. 445–450, IEEE, 2020.

[18] M. Almseidin, J. Al-Sawwa, and M. Alkasassbeh, "Generating a benchmark cyber multi-step attacks dataset for intrusion detection," Journal of Intelligent & Fuzzy Systems, no. Preprint, pp. 1–15.

[19] S. Shakya, "Modified gray wolf feature selection and machine learning classification for wireless sensor network intrusion detection," IRO Journal on Sustainable Wireless Systems, vol. 3, no. 2, pp. 118–127, 2021.

[20] W. Liu, "Research on dos attack and detection programming," in 2009 Third International Symposium on Intelligent Information Technology Application, vol. 1, pp. 207–210, IEEE, 2009.

[21] P. G. Jeya, M. Ravichandran, and C. Ravichandran, "Efficient classifier for r2l and u2r attacks," International Journal of Computer Applications, vol. 45, no. 21, pp. 28–32, 2012.

[22] G. Saporito, "A deeper dive into the nsl-kdd data set," 2019.

[23] H. Almuallim and T. G. Dietterich, "Learning boolean concepts in the presence of many irrelevant features," Artificial intelligence, vol. 69, no. 1-2, pp. 279–305, 1994.

Dataset Link:

NSL – KDD:

https://www.kaggle.com/datasets/kaggleprollc/nsl-kdd99-dataset