Comparative Analysis of Tree-based Intrusion Detection Modelling and Machine Learning Classification Models using Cyber-Security Dataset

Motlatso Mokoele
Department of Computer Science University of Limpopo
Polokwane, South Africa

Sello Mokwena
Department of Computer Science University of Limpopo
Polokwane, South Africa

ABSTRACT
One of the critical problems organizations encounters is the increasing prevalence of cyber-criminals exploiting vulnerabilities, leading to identity theft. This breach of privacy not only threatens the organization's financial assets, but can also have long-lasting consequences such as damaged reputations and legal implications. To address these issues, the study presented a thorough comparative analysis between tree-based intrusion detection model and popular machine learning classifiers using the well-established KDD99 dataset. The approach leverages a hybrid feature selection method, integrating the Gini index and information gain within a decision tree framework to enhance model efficiency. Evaluation metrics encompass precision, F1 score, confusion matrix, precision, recall, and execution time. Rigorous dataset preprocessing eliminates noise and biases. The findings reveal nuanced insights into model strengths and weaknesses, emphasizing the efficacy of the hybrid feature selection method in tree-based models. This study offers valuable guidance for cybersecurity professionals, helping to select models based on specific performance criteria. Ultimately, the research contributes to the advancement of intrusion detection techniques, highlighting potential areas for further exploration and improvement in the pursuit of more efficient and accurate intrusion detection systems.

General Terms
Cyber Threats, Data Preprocessing, Evaluation Metrics, Classification Models, Digital Landscape, Denial-of-Service, Internet of Things.

Keywords
Cybersecurity, intrusion detection, machine learning, hybrid feature selection, tree-based intrusion detection modeling, Gini index, Information Gain.

1. INTRODUCTION
The requirement for cyber security and fortification in contrast to many forms of cyber security issues has been consistently accumulating. The primary reason for this accumulation is the influence of internet-of-things (IoT), the enormous amount of development and advancements in computer networks, and the large number of vital applications utilized by people or organizations [1]. Cyberattacks such as denial-of-service, computer malware, and unauthorized access caused catastrophic damage and financial losses. For example, as stated by Alqahtani [2] in May 2017, a single ransomware virus caused a large cost to various companies and industries, together with finance, healthcare, energy, and tertiary institutions, causing a loss of about 8 billion USD. Some studies have shown that a data breach expenditure on an impacted company usually costs 3.9-8.19 million US dollars [1]. Once again on 9 May 2022, in its State of Ransomware 2022 report, the British cybersecurity firm Sophos stated that 51% of South African organizations it surveyed had experienced ransomware in 2021. Other key findings included a significant portion (49%) of organizations paying ransom remands and the effects of a ransomware attack, the cost of recovery coming to about R11.5 million [3]. On 18 March 2022, it was revealed that the South African Credit Bureau TransUnion had been hacked for ransom and that hundreds of businesses were in danger. According to reports, hackers used an authorized client's log-in information to access the bureau's server. They were referred to as "criminal third parties" [4]. The organization suffers these attacks because its network access control comprises of traditional security mechanisms. The utilization of classifiers that produces inadequate prediction accuracy may also cause these damages since cyber-security dataset comprises several types of cyber-attacks with important parameters.

2. LITERATURE REVIEW
Intrusion detection techniques play an important role in safeguarding the integrity and security of modern information systems. Methods are designed to identify and respond to malicious activities, cyberattacks, and unauthorized access within a network or computing environment. With the ever-evolving landscape of cyber [5] threats, researchers and practitioners have been constantly developing and refining identification techniques to stay one step ahead of cybercriminals. This introduction will briefly explore some recent developments and key techniques in this field, including tree-based ID, SVM, KNN, and LR. Tree-based ID models, such as RFs and DTs, have gained significant attention due to their versatility and adaptability to identify anomalous activities within network traffic. Recent studies have demonstrated the effectiveness of RF and DT in accurately detecting network intrusions by analyzing network data patterns [6]–[8]. SVMs are another class of ID techniques which were illustrated in [8] to efficiently classify network traffic into normal and malicious categories. KNN and LR are also noteworthy techniques in ID. These were studied in [9], [10]. These techniques collectively contribute to the ongoing efforts to improve the security of digital systems by enabling early identification and mitigation of potential threats. Researchers are continually refining these methods and exploring novel approaches to address the ever-evolving landscape of cybersecurity challenges.

2.1 Intrusion detection systems
An intrusion detection system (IDS) serves as a device or software application designed to actively monitor data flow of

data across a computer network, identifying and detecting instances of malicious activities or policy breaches [11]. The system operates through various types, each customized to specific aspects of network security [11]:

1. Network Intrusion Detection Systems (NIDS):

NIDS refers to an IDS variant that examines the data traffic traversing a computer network. This involves a meticulous analysis of the patterns and characteristics of the network's data flow to identify potential threats or unauthorized activities.

2. Host-Based Intrusion Detection Systems (HIDS):

HIDS represents another category of IDS, focusing its surveillance on files within an operating system. By monitoring the activities and changes that occur in the operating system files, HIDS aims to detect any anomalies or deviations from established security policies.

To provide a comprehensive view of security requirements in a cloud environment [12], investigations were carried out to collect and classify both attacks and vulnerabilities related to the different cloud models. This study led to the development of a taxonomy that delineates cloud security threats and proposes possible measures to mitigate them. The main objective of this research was to emphasize the importance of detecting and prevention of intrusions as a service offered in cloud environments.

To ensure Internet security, effective detection, and mitigation of distributed denial of service (DDoS) attacks [13], novel collaborative intrusion prevention architecture (CIPA) was proposed, with the aim of antagonizing coordinated intrusion activities. The architecture was deployed as a virtual network of an artificial neural network over the substrate of the networks. The CIPA takes advantage of the parallel and simple mathematical manipulation of neurons in a neural network since it can separate its light-weight computation supremacy from the programable switches of the substrate.

2.2 Machine Learning in Intrusion Detection

According to research from reference [14], intrusions into computer or network systems are characterized as actions that disrupt the established attributes of a secure and stable system, compromising its security in terms of confidentiality, availability, or integrity. A central theme of this book revolves around anomaly characterization within networked computer systems, utilizing Machine Learning (ML) techniques for detection. The author explores vulnerabilities in different layers of network systems, often stemming from protocol weaknesses. ML-based approaches to combating network intrusions are categorized into supervised learning, unsupervised learning, probabilistic learning, soft computing, and combination learners.

In a comprehensive review paper [15], various types of intrusion detection (ID) are discussed, emphasizing anomaly detection as one of the categories. ML-based anomaly detection techniques, rooted in explicit or implicit models, are explored, including Genetic Algorithms, Fuzzy Logic, Neural Networks, Bayesian Network, and Outlier Detection.

A study [16] evaluated 12 ML algorithms, specifically focusing on their ability to identify anomalous behaviors within network operations. Using openly available datasets (CICIDS-2017, UNSW-NB15, ICS cyberattack datasets) and the ALICE high-performance computing facility, the study revealed that the Random Forest (RF) algorithm consistently demonstrated superior performance across multiple metrics for all datasets, suggesting its effectiveness in various scenarios.

In [17], a novel two-tier classification model based on ML techniques (NB, certainty factor voting KNN, linear discriminant analysis) was introduced. Experimentation with the NSL-KDD dataset yielded promising results, showcasing improved detection rates and reduced false alarms compared to existing models. The two-tier model efficiently addressed the challenges posed by imbalanced network anomaly datasets, demonstrating strong detection capabilities, particularly for rare and intricate attack types.

2.3 Tree-Based Intrusion Detection

Al-Omari (2021) introduced an intelligent intrusion detection (ID) model designed for predicting and detect intrusions in cyberspace. The model, utilizing Decision Tree (DT) concepts and considering the ranking of security features, demonstrated efficiency in detecting and predicting cyber-attacks on Network Intrusion Detection (NID) systems. The approach was validated using predefined performance metrics such as accuracy, precision, recall, and F score, revealing superior performance and reduced computational complexity compared to traditional machine learning (ML) techniques [18].

Sarker (2020) presented the ML-based security model, emphasizing the importance of security feature ranking in constructing a tree-based generalized ID model. The IntruDTree model proved effective in predicting unseen test cases while minimizing computational complexity. Evaluation metrics, including precision, recall, F-score, precision, and ROC values, demonstrated its efficacy compared to popular traditional ML methods such as the NB classifier, LR, SVM, and KNN [1].

Ingre (2018) introduced a DT-based intrusion detection system (IDS) for the NSL-KDD dataset. Incorporating the correlation feature selection method for enhanced prediction efficacy, the proposed IDS achieved high detection rate (DR) and accuracy in both five-class and binary-class classification scenarios, outperforming other reported techniques [19].

Tekln (2022) developed an intelligent IDS tailored for Internet of Things (IoT) devices, employing a DT classifier to classify assault varieties. The proposed model achieved a high classification accuracy of 97.43%, showcasing its effectiveness in responding to various cyber-attacks on IoT devices [20].

In the context of Intelligent Transportation Systems (ITSs), the paper [21] proposed an intelligent IDS based on tree structure ML models for safeguarding Autonomous Vehicles (AVs). Results demonstrated the system's capability to identify various cyber-attacks in AV networks, showcasing high detection rates and low computational costs.

Addressing the complexity of intrusion analysis due to automated data collection, the study [22] explored the methods and introduced a tree-based stacking ensemble technique (SET). Implemented on intrusion datasets (NSL-KDD and UNSW-NB15), the proposed SET excelled in identifying normal and anomalous traffic, emphasizing its potential for improving cybersecurity in IoT and large-scale networks.

2.4 Selection Techniques

To address the challenge of data dimensionality in machine learning for network security, this review of the literature introduces the GA-based Feature Selection (GbFS) method, aiming to enhance intrusion detection by preserving crucial

information with minimal features [23]. Recognizing the critical task of protecting networks from cyber threats, the review emphasizes the proven efficacy of machine learning in the development of IDSs [23]. GbFS is designed to optimize feature selection, incorporating parameter adjustment and a novel fitness function [23]. Rigorous tests on benchmark datasets showcases GbFS's superior performance, achieving a maximum accuracy of 99.80% and surpassing standard feature selection methods [23].

To address cyber threats, this study explores the synergy between artificial intelligence and IDS [24]. Focused on DDoS attacks, a novel model with a decision tree algorithm and enhanced Gini index is proposed, achieving 98% precision on the UNSW-NB15 dataset. The selection of features of the Gini index reduces dimensionality and mitigates overfitting, making the approach promising for real-world network security applications.

To protect networks and sensitive data from cyber threats, the IDS plays a crucial role, prompting the exploration of various methodologies. Taking advantage of the efficiency of ML methods, this study employs DT, Gradient Boosting Tree (GBT), Multilayer Perceptron (MLP), AdaBoost, Long-Short Term Memory (LSTM) and Gated Recurrent Unit (GRU) models for intrusion detection [25]. Using UNSW-NB 15 and Network TON_IoT datasets for offline analysis, the study focuses on modern-day attacks, with Gini Impurity-Based Weighted Random Forest (GIWRF) serving as an embedded feature selection technique. By selecting 20 features from UNSW-NB 15 and 10 from Network TON_IoT, the study optimizes the set to combat high-dimensional challenges.

To optimize data mining tasks, feature selection plays a crucial role in identifying and isolating significant features for quality information [26]. Ranker-based algorithms, including Relief-F, Information Gain, Gini Index, Correlation, and Minimum Redundancy Maximum Relevance, generate a rank-list based on feature scores, simplifying the mining process. In the context of intrusion detection, this [26] work employs rankers to select relevant features, conducting experiments on the SSE Net 2011 dataset. Using a machine learning classifier, precision plots guide the determination of the optimal number of features, revealing substantial insights into the data set.

3. METHODOLOGY

3.1 Proposed Method

In the proposed study, the intrusion detection model consists of three primary modules. The first module consists of three processes, namely data exploration, data pre-processing and standardization, and features evaluation and selection. These processes are crucial in order to construct the tree-based intrusion detection approach based on hybrid feature ranking and selection. The last two modules are concerned with model training and testing in order to construct a classification model that is capable of detecting intrusions in cyberspace. Fig. 1. illustrates how the proposed study is lined up, and each stage of the model is discussed in this section.

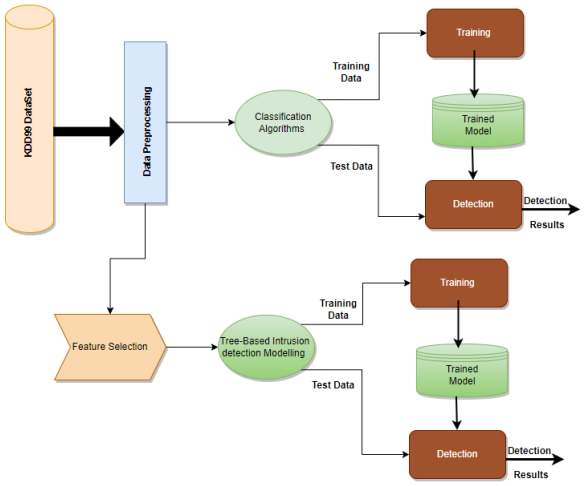

Fig 1 : Proposed method

3.2 Data Collection

In this study, the KDD-99 dataset was used, a comprehensive collection of simulated cybersecurity data. The data set encompasses various attack categories, including denial of Service (DoS), User to root (U2R), Remote to Local (R2L), and probing attacks. Each category represents different types of cybersecurity threat, providing a diverse and realistic set of scenarios for the analysis of intrusion detection. The choice of the KDD-99 dataset is based on its prominence within the field of intrusion detection. It has been widely used as a benchmark for evaluating the effectiveness of intrusion detection systems. The simulated nature of the data set allows for the examination of various attack scenarios, making it suitable for training and testing machine learning models in a controlled environment.

3.3 Tools and Platforms

The implementation of the research involved the use of various tools, with a primary focus on Google Colab (Collaboratory). Google Colab provides a cloud-based Jupyter notebook environment that facilitates collaborative coding and data analysis. This platform allows for seamless integration with Google Drive, enabling easy access to datasets and model output. The collaborative nature of Google Colab promotes efficient teamwork and code sharing. The utilization of Jupyter notebooks in a cloud-based environment ensures flexibility and accessibility. Google Colab not only supports the Python programming language but also provides access to GPU resources, enhancing the speed of model training and analysis. The integration with Google Drive streamlines data storage and sharing, contributing to a more organized and collaborative research process.

3.4 Data Analysis

Prior to model development, a thorough examination of the quality and characteristics was conducted. This involved assessing the completeness and integrity of the data, identifying any missing values or outliers, and ensuring a balanced distribution of instances across different attack categories. Understanding the nature of the data set is crucial to build robust intrusion detection models. A key step in the analysis process was to validate the number of records and characteristics within the data set. This step ensures consistency and precision in subsequent model training and evaluation. An accurate representation of the dimensions is fundamental for building reliable machine learning models for intrusion detection. Focused attention was given to instances

classified as attacks within the dataset. This involved a detailed analysis of attack patterns, distribution across categories, and exploration of the characteristics that contribute to the identification of malicious activities. Understanding the characteristics of attacks is essential for developing effective intrusion detection models. To enhance the interpretability of the models, variables were classified and feature correlation with the predicted attribute (attack or normal) was examined. This step helps identify the most influential features for intrusion detection. Understanding the relationships between variables contributes to the selection of relevant features, optimizing the model's performance.

3.5 Data Pre-Processing

Intrusion detection models are highly dependent on the quality and relevance of the input data. To enhance the effectiveness of the models, a systematic approach to data pre-processing and standardization was adopted. This involved several key steps to ensure the integrity and consistency of the data set.

3.5.1 Redundancy Elimination

To address the potential pitfalls associated with redundant data, an advanced techniques was deployed. Feature correlation analysis, leveraging the Pearson correlation coefficient, allowed the study to identify and eliminate redundant features. Additionally, dimensionality reduction through principal component analysis (PCA) was applied. This not only streamlined the data set, but also retained essential information, ensuring a judicious balance between data reduction and preservation.

3.5.2 Transformation of Categorical Variables

Given that machine learning models often require numerical input, the transformation of categorical variables becomes imperative. The chosen methodology for this task was one-hot encoding. This process involves creating binary columns for each category, thereby indicating the presence or absence of that category in the original data. This transformation ensures that the models can effectively interpret and utilize categorical information during the training phase.

3.5.3 Feature Scaling

A critical aspect of data preprocessing is feature scaling, aimed at preventing certain features from disproportionately influencing the model training process due to differences in scale. The approach embraced Min-Max scaling, in which the values of numerical features were transformed to a specific range, typically between 0 and 1. This normalization ensured that each feature contributed proportionately to the model's learning process.

3.5.4 Feature Selection and Ranking

Effective feature selection is paramount to optimizing both model performance and interpretability. In this study, Gini index for decision trees and information gain for entropy-based models was used. These metrics allowed us to evaluate the significance of each feature in relation to the predicted attribute. By discerning the contribution of each attribute, we could selectively retain those that wielded significant influence in the intrusion detection context.

4. TREE-BASED INTRUSION DETECTION

4.1 Development of a Tree-Based Model

The development of the Tree-Based Intrusion Detection (ID) model centers around the use of Decision Trees. This section provides a detailed description of the model, emphasizing its construction based on essential security features. Decision trees offer an interpretable and strategic approach to classification, with a particular focus on the distinctive characteristics of network activities. The model construction involves a meticulous process, integrating specific settings such as balanced class weights, a Gini impurity criterion, and controlled depth and node splitting conditions. Decision trees are trained on a dataset split into training and testing sets, ensuring robustness and accuracy in the classification of network connections.

4.2 Root node selection

In the development of the Tree-Based Intrusion Detection (ID) model, a pivotal step involves the thoughtful selection of the root node. This critical decision-making point is achieved through a strategic application of the Gini index technique. The Gini index method meticulously evaluates the impurity of nodes within the Decision Tree, guiding the algorithm in choosing the most optimal root node. The approach to root node selection involved a comprehensive analysis of Gini impurity across potential nodes, ensuring that the chosen root effectively discriminates between normal and intrusive network activities. By prioritizing nodes with lower Gini impurity, the Decision Tree establishes a strong foundation for subsequent branching, ultimately enhancing its precision and reliability in classifying network connections. This methodical root node selection process contributes significantly to the overall effectiveness of the tree-based ID model in accurately identifying and categorizing cybersecurity threats.

4.3 Selected Classification Algorithms

4.3.1 Support Vector Machine (SVM)

The SVM model was meticulously crafted for ID within the KDD99 dataset. Leveraging the Radial Basis Function (RBF) kernel, known for its prowess in handling complex and nonlinear data patterns, the SVM is tailored to excel in distinguishing between normal and intrusive network activities. With a carefully configured set of hyperparameters, including a gamma value of 0.1 and a regularization parameter (C) of 1.0, the SVM is optimized for precision and recall. The RBF kernel's capability to capture intricate decision boundaries aligns seamlessly with the need to identify subtle patterns in high-dimensional data.

4.3.2 K-Nearest Neighbors (KNN)

The KNN classifier, developed for ID in the KDD99 dataset, adopts a distinctive and flexible approach. As an instance-based learning algorithm, KNN categorizes network connections according to their proximity to neighboring data points. Its design emphasizes simplicity, focusing on selecting the appropriate number of nearest neighbors (k) for classification. The construction of the KNN model showcases adaptability, making it suitable for scenarios where similar network activities tend to cluster together in feature space.

4.3.3 Logistic Regression (LR)

The Logistic Regression model for ID within the KDD99 dataset follows a classic and transparent approach to classification. LR, valued for its versatility in handling binary and multiclass classification tasks, aims to accurately classify network connections into intrusion and non-intrusion categories. Key settings, such as the use of the “lbfgs” solver and “auto” multiclass classification, are thoughtfully chosen to ensure efficient parameter optimization. LR's interpretability and simplicity make it an ideal choice for problems where the relationship between features and the target variable is

approximately linear.

5. EVALUATION METRICS AND RESULTS

5.1 Performance metrics

In the evaluation of ID models, a set of performance metrics was chosen to provide a comprehensive understanding of their effectiveness. The selected metrics include precision, precision, recall, F1 score, and confusion matrix.

5.1.1 Accuracy

This metric measures the overall accuracy of the model's classifications. It is calculated as the ratio of correctly predicted instances to the total instances. Higher accuracy indicates a better-performing model in terms of overall classification.

5.1.2 Precision

Precision is the ratio of true positive predictions to the sum of true positive and false positive predictions. Measures the accuracy of positive predictions made by the model, highlighting its ability to avoid false positives.

5.1.3 Recall (sensitivity)

Recall, or sensitivity, is the ratio of true positive predictions to the sum of true positive and false negative predictions. Assesses the model's capability to capture all relevant instances, minimizing false negatives.

5.1.4 F1 score

The F1 score is the harmonic mean of precision and recall. It provides a balanced measure, which is particularly useful when there is an uneven class distribution. A higher F1 score indicates a model with high precision and recall.

5.1.5 Confusion matrix

The confusion matrix is a table that illustrates the performance by comparing actual and predicted classifications. It comprises four key components: true positives (TP), true negatives (TN), false positives (FP) and false negatives (FN).

In the subsequent section, the study will interpret the specific results obtained from the evaluation metrics, shedding light on the performance of tree-based intrusion detection models and traditional machine learning models in the context of cybersecurity.

5.2 Results

This comparative analysis explores the effectiveness of Decision Tree ID model, in contrast to SVM, KNN, and LR. Through this analysis, cybersecurity professionals are equipped with valuable guidance for making informed choices in the development of IDSs.

5.2.1 Accuracies

In the comparative analysis, the models displayed varying degrees of accuracy, as depicted in Figure 3. SVM and LR achieved 100% prediction accuracy, correctly classifying all instances, which is a significant achievement in cybersecurity. This level of precision is particularly essential to minimize false positives and false negatives, where misclassification can have significant consequences. DT and KNN followed closely, with accuracies of 99.98% and 99.99%, respectively. These models showed good performance, accurately identifying intrusions with high fidelity. Although the variations in accuracy were relatively minor, the choice between these models should also consider other factors, such as computational resources and interpretability.

	Model	Training Accuracy %	Testing Accuracy %
0	Intelligent Tree	99.998283	99.979394
1	Support Vector Machine	100.000000	100.000000
2	K-Nearest Neighbours	99.980252	99.986263
3	LogisticRegression	100.000000	100.000000

Fig. 2: Model accuracy

5.2.2 Classification Reports

In this comparative analysis, the study explores the performance evaluation of multiple classification models based on the attached classification reports. The reports cover key metrics such as precision, recall, and the F1 score, providing valuable insights into the effectiveness in handling specific classification tasks. In Figure 4, precision and recall values close to 100% for classes such as 'Dos', 'Normal', and 'Probe' indicate the model's capability to minimize false positives and false negatives in classifying network connections, crucial for identification. Consistently high F1 scores for these classes reveal a balanced trade-off between precision and recall, demonstrating the effectiveness in identifying intrusion patterns while minimizing false detections. Despite these strengths, other matrices, including prediction accuracy, indicate the model's limitations in correctly predicting all classes, emphasizing the importance of the included confusion matrix for assessing misclassifications.

CLASSIFICATION REPORT:

	dos	normal	probe	r2l	u2r	accuracy	\
precision	0.999908	1.0	0.995402	0.988304	0.857143	0.999794	
recall	1.000000	1.0	0.997696	0.982558	0.750000	0.999794	
f1-score	0.999954	1.0	0.996548	0.985423	0.800000	0.999794	
support	10894.000000	17610.0	434.000000	172.000000	8.000000	0.999794	

	macro avg	weighted avg
precision	0.968151	0.999789
recall	0.946051	0.999794
f1-score	0.956385	0.999790
support	29118.000000	29118.000000

Fig 3: DT Classification Report

In Fig. 5, precision and recall values of 100% for all classes, including 'dos', 'normal', 'probe', 'r2l' and "u2r", showcase the SVM model to minimize false alarms and capture true threats, crucial in ID. The F1 scores reaching 100% across all classes highlight the accuracy and capability to identify intrusion patterns with a low false detection rate. Despite these perfect scores, the inclusion of the confusion matrix in the evaluation aims to scrutinize potential misclassifications and ensure that all instances are correctly classified, which is included in Fig. 9.

CLASSIFICATION REPORT:

	dos	normal	probe	r2l	u2r	accuracy	macro avg	\
precision	1.0	1.0	1.0	1.0	1.0	1.0	1.0	
recall	1.0	1.0	1.0	1.0	1.0	1.0	1.0	
f1-score	1.0	1.0	1.0	1.0	1.0	1.0	1.0	
support	10894.0	17610.0	434.0	172.0	8.0	1.0	29118.0	

	weighted avg
precision	1.0
recall	1.0
f1-score	1.0
support	29118.0

Fig. 4: SVM Classification Report

In Fig. 6, the KNN model demonstrates good precision and recall values, near 100% for classes like "dos", "normal", and "r2l", highlighting its ability to minimize false alarms and identify genuine threats in ID. Consistently high F1 scores emphasize the KNN classifier's ability to strike a balance between reducing false detections and maximizing the capture of real threats, a critical aspect of ID. Despite achieving high accuracy, other matrices, including prediction accuracy, suggest limitations in correctly predicting all classes,

necessitating the inclusion of the confusion matrix for a comprehensive evaluation of potential misclassifications.

CLASSIFICATION REPORT:

	dos	normal	probe	r2l	u2r	accuracy	\
precision	0.999908	0.999830	1.000000	1.000000	1.0	0.999863	
recall	1.000000	1.000000	0.993088	0.994186	1.0	0.999863	
f1-score	0.999954	0.999915	0.996532	0.997085	1.0	0.999863	
support	10894.000000	17610.000000	434.000000	172.000000	8.0	0.999863	

	macro avg	weighted avg
precision	0.999948	0.999863
recall	0.997455	0.999863
f1-score	0.998697	0.999862
support	29118.000000	29118.000000

Fig. 5: KNN Classification Report

In Fig. 7, the LR model shows precision and recall values, each at 100% for all classes, including ‘dos’, ‘normal’, ‘probe’, ‘r2l’, and “u2r”. This outstanding performance underscores the proficiency of the LR model in minimizing false alarms and capturing true threats in ID. With 100% precision and recall, the LR model proves to be an incredibly robust ID tool, maintaining a harmonious balance between reducing false detections and maximizing the capture of genuine threats, as reflected in consistently high F1 scores.

CLASSIFICATION REPORT:

	dos	normal	probe	r2l	u2r	accuracy	macro avg	\
precision	1.0	1.0	1.0	1.0	1.0	1.0	1.0	
recall	1.0	1.0	1.0	1.0	1.0	1.0	1.0	
f1-score	1.0	1.0	1.0	1.0	1.0	1.0	1.0	
support	10894.0	17610.0	434.0	172.0	8.0	1.0	29118.0	

	weighted avg
precision	1.0
recall	1.0
f1-score	1.0
support	29118.0

Fig. 6: LR Classification Report

5.2.3 Confusion matrix

The confusion matrix was included as one of the evaluation matrices to check how many classes were misclassified. The confusion matrix in the figures below provides insight into the classification results. It reveals that the model correctly identified instances and incorrectly or misclassified instances. In Fig. 8, the confusion matrix presents the model's accurate classification of most instances, particularly for ‘dos’ and “normal”. However, limitations were observed, with misclassifications, including instances of ‘r2l’ as “probe” and “u2r” as “r2l”. These findings suggest a slight challenge in precisely predicting attacks within the ‘probe’, ‘r2l’ and “u2r” classes.

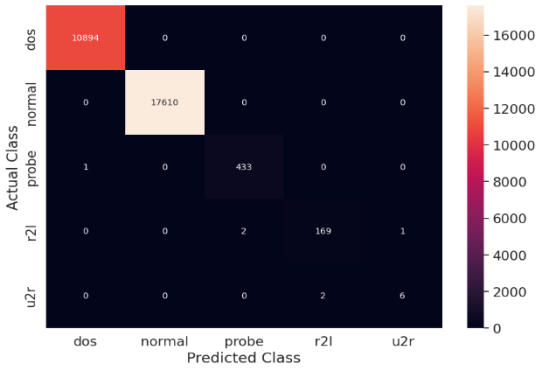

Fig. 7: DT confusion matrix

Fig. 9 illustrates that the SVM model successfully classified all classes, highlighting its discriminatory power and robustness in ID within the KDD99 dataset. Renowned for defining clear decision boundaries and capturing complex patterns, the proficiency is evident in accurately distinguishing various characteristics of network connection, including “dos”, “normal”, “probe”, “r2l”, and ‘u2r’. Its ability to maximize class margins contributes to highly accurate and reliable classifications.

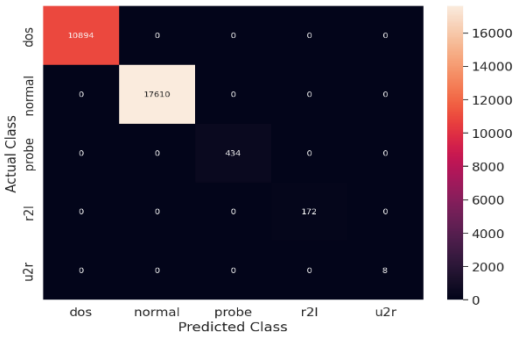

Fig. 8: SVM Confusion Matrix

The confusion matrix in Fig. 10 reveals the strong performance in accurately classifying instances, particularly for the “dos” and “normal” classes. However, it showed limitations, misclassifying instances, particularly in the ‘probe’ and “r2l” classes, suggesting some challenges in predicting attacks within these categories.

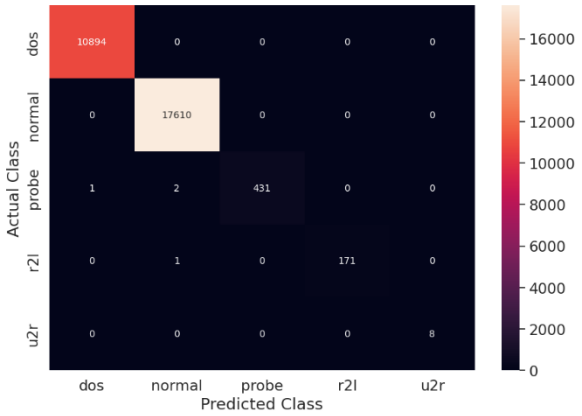

Fig. 9: KNN Confusion Matrix

Fig. 11 indicates that the LR model successfully classified all classes in the confusion matrix, demonstrating its simplicity and transparency. LR’s efficacy in accurately classifying underrepresented attack classes, like “r2l” and ‘u2r’, highlights its robustness, especially in the context of a data set with class imbalance. This underscores the adaptability and effectiveness of LR to discern patterns within network connections.

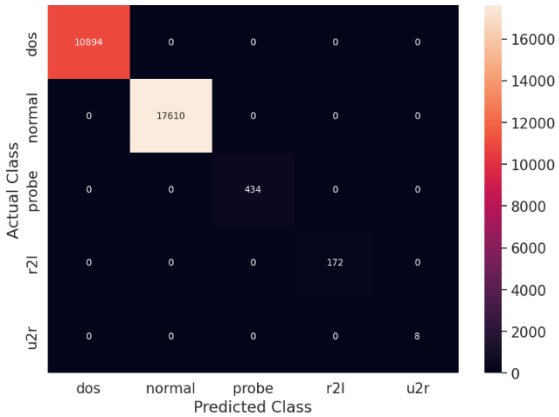

Fig. 10: LR confusion matrix

6. DISCUSSION

Comparative analysis of tree-based ID models and ML classification algorithms, using the cyber-security dataset, yielded notable results. SVM and LR models achieved remarkable 100% accuracy in classifying network connections, showcasing their potential for highly accurate IDSs. Although they did not reach the 100% accuracy mark, DT and KNN models showed strong performances with rates of 99.98% and 99.99%, respectively, striking a balance between accuracy and interpretability, making them appealing choices for ID applications.

Furthermore, the study highlighted the importance of precision, recall, and F1 score alongside accuracy. SVM and LR not only exhibited high accuracy, but also demonstrated impressive precision and recall, minimizing false positives and false negatives in ID scenarios. These findings have profound implications for the field, suggesting that SVM and LR are suitable for accuracy-centric applications, while DT and KNN can offer transparency and interpretability. Examination of the confusion matrix reinforced the effectiveness of ML models in correctly classifying network connections, highlighting their potential to enhance network security and mitigate cyber threats with reduced operational burdens on security teams.

7. CONCLUSION

Comparative analysis of tree-based ID models and ML classification algorithms has revealed significant findings. The SVM and LR models exhibited an exceptional 100% precision in classifying network connections, highlighting their potential for precise identification. The DT and KNN models delivered commendable performance with accuracy rates of 99.98% and 99.99%, respectively, showcasing a valuable balance between accuracy and interpretability. This research contributes valuable information to cybersecurity, offering a nuanced understanding of machine learning models for identification. The 100% accuracy of the SVM and LR models makes them ideal for precision-centric applications such as critical infrastructure protection. Meanwhile, the robust performance of the DT and KNN models suggests their effectiveness in ID with transparency and ease of interpretation. This study empowers cybersecurity professionals with informed choices for designing IDSs, tailored to specific organizational needs, thus fortifying network security and protecting digital assets.

8. ACKNOWLEDGMENTS

I want to express my gratitude to the Almighty for the strength and guidance that have helped me on this journey. I am very grateful to my mother for her constant love and support, which have meant the world to me. I also wish to thank Prof. SN Mokwena, my research supervisor, for his guidance. I appreciate my classmates, MR Mulaudzi T and MS Ndleve NM, for the camaraderie and encouragement we shared. To my family, I am grateful for their support throughout this journey.

9. REFERENCES

[1] I. H. Sarker, Y. B. Abushark, F. Alsolami, and A. I. Khan, 'IntruDTree: A machine learning based cyber security intrusion detection model", Symmetry (Basel)., vol. 12, no. 5, May 2020, doi: 10.3390/SYM12050754.

[2] H. Alqahtani, I. H. Sarker, A. Kalim, S. M. Minhaz Hossain, S. Ikhlaq, and S. Hossain, 'Cyber intrusion detection using machine learning classification techniques', in Communications in Computer and Information Science, Springer, 2020, pp. 121–131. doi: 10.1007/978-981-15-6648-6_10.

[3] 'South African companies getting hit by ransomware — and they are paying for it, Dotnetworx'. https://dotnetworx.co.za/south-african-companies-getting-nailed-by-ransomware-and-they-are-paying-up/ (accessed Oct. 23, 2023).

[4] 'TransUnion Credit Bureau hacked for ransom - hundreds of companies under threat | Business.' https://www.news24.com/fin24/companies/credit-bureau-transunion-hacked-for-ransom-hundreds-of-companies-under-threat-20220318 (accessed October 23, 2023).

[5] N. H. Al-A'araji et al., "A Survey on Anomaly Based Host Intrusion Detection System You may also like Research on Intrusion Detection Method Based on Cloud Computing Mengmeng Cai and Honglin Wang-Classification and Clustering Based Ensemble Techniques for Intrusion Detection Systems: A Survey An Improved Network Intrusion Detection Based on Deep Neural Network A Survey on Anomaly Based Host Intrusion Detection System," IOP Conf. Ser. J. Phys. Conf. Ser., vol. 1000, p. 12049, 2018, doi: 10.1088/1742-6596/1000/1/012049.

[6] K. Rai, M. S. Devi and A. Guleria, "Decision Tree Based Algorithm for intrusion detection', Int. J. Adv. Netw. Appl., vol. 07, no. 04, pp. 2828–2834, 2016, [online]. Available: https://www.researchgate.net/publication/298175900

[7] N. Farnaaz and Jabbar, "Random Forest Modeling for Network Intrusion Detection System', Procedia Comput. Sci., vol. 89, pp. 213–217, 2016, doi: 10.1016/j.procs.2016.06.047.

[8] I. Ahmad, M. Basheri, M. J. Iqbal and A. Rahim, "Performance Comparison of Support Vector Machine, Random Forest and Extreme Learning Machine for Intrusion Detection', IEEE Access, vol. 6, pp. 33789-3795, 2018, doi: 10.1109/ACCESS.2018.2841987.

[9] P. Garca-Teodoro, J. Daz-Verdejo, G. Maciá-Fernández, and E. Vázquez, 'Anomaly-based network intrusion detection: Techniques, systems, and challenges,' Comput. Secur., vol. 28, no. 1–2, pp. 18–28, 2009, doi: 10.1016/j.cose.2008.08.003.

[10] S. Malhotra, V. Bali and K. K. Paliwal, "Genetic programming and the K nearest neighbor classifier-based intrusion detection model', Proc. 7th int. Conf. Conflu. 2017 Cloud Computing. Data Sci. Eng., pp. 42–46, 2017, doi: 10.1109/CONFLUENCE.2017.7943121.

[11] "Intrusion Detection System." https://www.barracuda.com/support/glossary/intrusion-detection-system (accessed January 31, 2024).

[12] S. Iqbal et al., 'On cloud security attacks: A Taxonomy and Intrusion Detection and prevention as a service," J. Netw. Comput. Appl., vol. 74, pp. 98–120, Oct. 2016, doi: 10.1016/J.JNCA.2016.08.016.

[13] X. F. Chen and S. Z. Yu, 'CIPA: A collaborative intrusion prevention architecture for the programmable network and SDN 'Comput. Secur., vol. 58, pp. 1–19, May 2016, doi: 10.1016/J.COSE.2015.11.008.

[14] D. K. Bhattacharyya, 'Network Anomaly Detection: A Machine Learning Perspective Big Data Analytics View project Gene Expression Data View project", 2013, doi: 10.1201/b15088.

[15] H. Kaur, G. Singh, and J. Minhas, "A Review of Machine Learning-based Anomaly Detection Techniques," Int. J. Comput. Appl. Technol. Res., vol. 2, no. 2, pp. 185–187, Jul. 2013, doi: 10.7753/ijcatr0202.1020.

[16] N. Elmrabit, F. Zhou, F. Li and H. Zhou, "Evaluation of Machine Learning Algorithms for Anomaly Detection', Int. Conf. Cyber Secur. Prot. Digit. Serv. Cyber Secur. 2020, Jun. 2020, doi: 10.1109/CYBERSECURITY49315.2020.9138871.

[17] H. Haddad Pajouh, G. Dastghaibyfard, S. Hashemi and S. Hashemi hashemi, "Two-tier network anomaly detection model: a machine learning approach', J Intell Inf Syst, vol. 48, pp. 61–74, 2017, doi: 10.1007/s10844-015-0388-x.

[18] M. Al-Omari, M. Rawashdeh, F. Qutaishat, M. Alshira 'H and N. Ababneh, 'An Intelligent Tree-Based Intrusion Detection Model for Cyber Security', J. Netw. Syst. Manag., vol. 29, no. 2, April 20,21, doi: 10.1007/s10922-021-09591-y.

[19] B. Ingre, A. Yadav and A. K. Soni, "Decision tree based intrusion detection system for the NSL-KDD dataset', Smart Innov. Syst. Technol., vol. 84, no. Ictis 2017, pp. 207–218, 2018, doi: 10.1007/978-3-319-63645-0_23.

[20] R. TEKN, O. YAMAN and T. TUNCER, "Decision Tree Based Intrusion Detection Method in the Internet of Things", Int. J. Innov. Eng. Appl., vol. 6, no. 1, pp. 17–23, June 2022, doi: 10.46460/IJIEA.970383.

[21] L. Yang, A. Moubayed, I. Hamieh and A. Shami, "Tree-based intelligent intrusion detection system in the Internet of vehicles', 2019 IEEE Glob. Commun. Conf. GLOBECOM 2019 - Proc., no. Ml, 2019, doi: 10.1109/GLOBECOM38437.2019.9013892.

[22] M. Rashid, J. Kamruzzaman, T. Imam, S. Wibowo, and S. Gordon, "A tree-based stacking ensemble technique with feature selection for network intrusion detection," Appl. Intell., vol. 52, no. 9, pp. 9768–9781, 2022, doi: 10.1007/s10489-021-02968-1.

[23] Z. Halim et al., 'An effective genetic algorithm-based feature selection method for intrusion detection systems', Comput. Secur., vol. 110, p. 102448, Nov. 2021, doi: 10.1016/J.COSE.2021.102448.

[24] M. A. Bouke, A. Abdullah S. H. ALshatebi, M. T. Abdullah, and H. El Atigh, 'An intelligent DDoS attack detection tree-based model using the Gini index feature selection method', Microprocess. Microsyst., vol. 98, p. 104823, April 2023, doi: 10.1016/J.MICPRO.2023.104823.

[25] R. A. Disha and S. Waheed, 'Performance analysis of machine learning models for intrusion detection system using Gini impurity-based weighted random forest (GIWRF) feature selection technique' Cybersecurity, vol. 5, no. 1, pp. 1–22, 2022, doi: 10.1186/s42400-021-00103-8.

[26] K. K. Vasan and B. Surendiran, 'Feature subset selection for intrusion detection using various rank-based algorithms', Int. J. Comput. Appl. Technol., vol. 55, no. 4, pp. 298–307, 2017, doi: 10.1504/IJCAT.2017.086017.