Computer Modeling in Engineering & Sciences

Tech Science Press

Doi:10.32604/cmes.2025.067098

ARTICLE

Enhancing Healthcare Cybersecurity through the Development and Evaluation of Intrusion Detection Systems

Muhammad Usama[1], Arshad Aziz[2], Imtiaz Hassan[2], Shynar Akhmetzhanova[3], Sultan Noman Qasem[4,*], Abdullah M. Albarrak[4] and Tawfik Al-Hadhrami[5]

[1]Department of Cyber Security, Pakistan Navy Engineering College, National University of Sciences and Technology, Karachi, 75350, Pakistan

[2]Department of Computer Science, Main Campus, Iqra University, Karachi, 75500, Pakistan

[3]Department of Computer Engineering, Astana IT University, Astana, 010000, Kazakhstan

[4]Computer Science Department, College of Computer and Information Sciences, Imam Mohammad Ibn Saud Islamic University (IMSIU), Riyadh, 11432, Saudi Arabia

[5]Computer Science Department, School of Science and Technology, Nottingham Trent University, Nottingham, NG11 8NS, UK

*Corresponding Author: Sultan Noman Qasem. Email: snmohammed@imamu.edu.sa

Received: 25 April 2025; Accepted: 10 July 2025; Published: 31 July 2025

ABSTRACT: The increasing reliance on digital infrastructure in modern healthcare systems has introduced significant cybersecurity challenges, particularly in safeguarding sensitive patient data and maintaining the integrity of medical services. As healthcare becomes more data-driven, cyberattacks targeting these systems continue to rise, necessitating the development of robust, domain-adapted Intrusion Detection Systems (IDS). However, current IDS solutions often lack access to domain-specific datasets that reflect realistic threat scenarios in healthcare. To address this gap, this study introduces HCKDDCUP, a synthetic dataset modeled on the widely used KDDCUP benchmark, augmented with healthcare-relevant attributes such as patient data, treatments, and diagnoses to better simulate the unique conditions of clinical environments. This research applies standard machine learning algorithms Random Forest (RF), Decision Tree (DT), and K-Nearest Neighbors (KNN) to both the KDDCUP and HCKDDCUP datasets. The methodology includes data preprocessing, feature selection, dimensionality reduction, and comparative performance evaluation. Experimental results show that the RF model performed best, achieving 98% accuracy on KDDCUP and 99% on HCKDDCUP, highlighting its effectiveness in detecting cyber intrusions within a healthcare-specific context. This work contributes a valuable resource for future research and underscores the need for IDS development tailored to sector-specific requirements.

KEYWORDS: Cybersecurity; KDDCUP; HCKDDCUP; machine learning; anomaly detection; data privacy

1 Introduction

Integrating modern data science and communication technologies into the healthcare system has become essential for patients and healthcare professionals to efficiently collect, store, retrieve, and share health information. However, the rapid evolution of these technologies comes with the proliferation of cyberattacks that specifically target healthcare systems. Currently, numerous security solutions are available, such as firewalls, antivirus software, cryptography-based encryption [1], authentication systems [2], Intrusion Prevention Systems (IPS), and Intrusion Detection Systems (IDS) to strengthen healthcare systems security against growing cyber threats and attacks. However, a literature review confirms that IDS

is a preferred, robust, and effective solution for enhancing cyber defenses in healthcare care [3]. Given the potential consequences of successful intrusions, from data breaches to financial loss and reputational damage, implementing a robust IDS is imperative to protect healthcare systems.

Traditionally, IDS has been used in computer networks to enhance security against unauthorized entry and different types of data breaches. However, directly adopting conventional IDS strategies into healthcare systems is extremely problematic given the distinct nature of the healthcare environment. They involve the high sensitiveness of patient information, heterogeneity and complexity of networked medical devices and systems, and the often occurring class imbalance in network traffic data due to infrequent but significant attack events. Hence, it is very important to test the effectiveness of IDS for healthcare systems with growing sophistication of cyberattacks and dependence on networked systems.

Towards this end, the objective of this research is to perform comparative analysis of standard machine learning algorithms like Random Forest (RF), Decision Tree (DT) and K-Nearest Neighbors (KNN) to unearth the strengths and weaknesses of intrusion detection. Such analysis exploits the benchmark-proven KDDCUP dataset and a newly synthesized dataset called HCKDDCUP, specifically for healthcare systems. The HCKDDCUP database has been tuned and designed with respect to realistic network traffic patterns and cyber-attacks for healthcare establishments as derived from the KDDCUP dataset. Compared to existing IDS benchmarks such as NSLKDD, CICIDS2017, and UNSWNB15, which are either outdated or too generic for healthcare environments, HCKDDCUP introduces domain-specific attributes (e.g., patient records, diagnoses, treatments) and emulates healthcare-specific cyberattack vectors such as EHR manipulation and DDoS on medical services. This makes it a significant advancement in the development of contextualized IDS datasets for critical sectors. So, the thrust of the study eventually is to test transferability of IDS models trained on the KDDCUP dataset and on HCKDDCUP dataset reflecting contemporary network behaviors and emerging patterns of attacks in healthcare systems. The KDDCUP dataset represents a benchmark for the intrusion detection studies and a reference point to evaluating different algorithms of IDS. To this end, the research work has two major research goals:

- Evaluate and compare the performance of the IDS algorithms on KDDCUP dataset to provide performance insights for a standard dataset;
- Evaluate the effectiveness of the IDS algorithms on a newly developed HCKDDCUP dataset that reflects real-world network behavior including various cyberattack scenarios and patterns within the context of healthcare systems.

Further, this study aims to provide findings and insights from the proposed work for assessing the strength and weaknesses of different IDS algorithms and to offer more resilient IDS solutions for healthcare systems. Specifically, the performance evaluation of IDS algorithms for healthcare systems with various datasets will allow researchers and practitioners to select relevant datasets and algorithms by considering their advantages and disadvantages. Moreover, evaluating the transferability of IDS to datasets provides better perceptions of the adaptability and generalizability of IDS models. In the following subsequent sections, we provide a comprehensive literature review by including key studies that have analyzed the performance of IDS on well-known datasets. Through this review, we carefully analyze state-of-the-art studies' strengths, limitations, and implications to provide a solid foundation for proposed work. Additionally, we describe the proposed work, which includes the generation of a synthetic dataset, selection criteria, data collection, and evaluation of IDS algorithms. Finally, we present the research findings, draw conclusions, and describe the potential future research opportunities.

2 Related Work

In recent years, cyberattacks have increased in volume and complexity, allowing attackers to exploit new vulnerabilities and perform intrusions in healthcare systems constantly. Therefore, healthcare systems require proactive defense mechanisms to detect and mitigate intrusions and minimize vulnerabilities. It's about the highly sensitive data in health systems such as personal information, financial records, and patient care, as well as diagnosis details that become inaccessible to unauthorized users or to illegal access in case of data breach or attack, which is a significant security issue for legal liabilities, fines, and remediation costs. Thus, the reputational damage from a breach or attack can be profound in healthcare systems. Thus, systematic literature review is critical to enhance the understanding of complexities associated with IDS algorithms and employ diverse datasets to develop robust IDS for healthcare systems that accurately represent real-world network traffic patterns and attack scenarios. Hence, this research is intended to conduct a comparative study among the IDS algorithms and to justify the benefits and drawbacks of KDDCUP data against the newly originated HCKDDCUP data to assess the efficiency in healthcare systems. This KDD dataset, proposed back in 1999, is worthy to be considered a reference data set for evaluating IDS algorithms as per the attack scenarios of network traffic. However, there are added drawbacks such as excessive data and old information, which is misinterpreted as footprints of new attack patterns and/or behavior. It also would not be capable of capturing ever-evolving methods of attack.

To this end, the IDS was proposed by taking data from the KDDCUP dataset in [4] to classify attacks on the basis of feedforward neural networks. This study shed some light on the ability of neural networks to detect and classify network attacks. The detection rates, however, were found to be unsatisfactory for R2L and U2R attacks. A related work was presented in [5], which introduced a comprehensive comparative analysis of IDS algorithms using KDDCUP and NSLKDD datasets. The work evaluated the merits and demerits of various algorithms based on IDS which affect their performance based on the characteristics of the datasets. The other study which was cited in [6] carried on the comparative analysis for the IDS algorithm extensively across multiple KDDCUP, NSLKDD, and DARPA datasets to identify the most authentic dataset representing real-world situations for network traffic. In the same way, the KDDCUP dataset regarding intrusion detection was also analyzed by characterizing it into four classes in [7] for evaluation of detection vs. false alarm IDS algorithms. However, the paper [8] pointed out some major issues about the proposed works [4–7] and recommended a few interventions.

Furthermore, the study in [9] focused on Distributed Denial of Service (DDoS) attacks by employing data mining and machine learning algorithms. Their results may not be robust and generalizable, but they emphasized the importance of appropriate features and dataset selection for accurate intrusion detection. An overall 98.63% accuracy was achieved using the Multilayer Perceptron algorithm with promising performance. In contrast, RF and Naïve Bayes (NB) algorithms produced poor accuracy rates to detect DDoS attacks. Machine learning algorithms for intrusion detection were evaluated in [10] using multiple widely used datasets to ascertain their effectiveness in handling different types of attacks. The UNSWNB15 dataset was introduced to provide a more comprehensive benchmark for evaluating IDSs compared to the KDDCUP dataset, with a maximum accuracy of 98.2% achieved using RF. However, the proposed work was limited to static datasets. A voting method that utilizes an ensemble of DT, RF, KNN, and Deep Neural Network (DNN) was proposed in [11], achieving an accuracy of 85.2% on KDDCUP. Although the ensemble method outperformed individual DT, RF, KNN, and DNN methods, the technique's performance was not assessed on network datasets. Consequently, the robustness and generalizability of the process may be compromised.

The CICIDS17 and CSECICIDS18 datasets, as introduced in [12], were characterized by realistic network traffic and updated network attacks. A comparative analysis of IDS performance was conducted, evaluating the effectiveness of various IDS techniques in detecting network intrusions and assessing the strengths and

weaknesses of each dataset in representing real-world network traffic. However, both datasets were found to be prone to the issue of high-class imbalance, which may lead to low accuracy. In another related study, the performance of a multi-feature correlation method with Convolutional Neural Networks (CNN) and Long Short-Term Memory (LSTM)-based models was demonstrated in [13] on network-based datasets such as UNSWNB15 and CICIDS17. Even though the Deep Learning (DL) model implicitly extracts its features, the proposed method employs various preprocessing and feature selection methods with regular updates to address network changes. Similarly, the work in [14] demonstrated the performance of an ensemble machine learning-based method on NSLKDD, UNSWNB15, and CICIDS17 datasets for intrusion detection, with continuous updates to adapt to dynamically changing networks, which may increase the processing time and resource consumption.

Recent studies have highlighted the use of hierarchical and hybrid deep learning models in healthcare. For instance, Ali and Zoltan [15] introduced the CHDLCY framework, which is a combined hybrid deep learning architecture that reuses layers to improve intrusion detection in multi-cloud healthcare systems. The model demonstrated faster training times and achieved accuracy levels between 98% and 100%. These hierarchical methods show potential in handling complex healthcare network traffic while ensuring scalability and performance. In addition, Ali et al. [16] developed a multimodal AI approach that integrates machine learning, deep learning, and anomaly detection for real-time cybersecurity in healthcare. Their model improves the speed and accuracy of threat detection using the synergy of several AI approaches by demonstrating superiority in protecting sensitive patient information and healthcare facilities from advanced cyber attacks.

In [17], a scalable machine learning-based intrusion detection method was proposed by preprocessing and analyzing large network traffic from real-time and high-speed networks. The method was subjected to poor computation due to the expensive optimization process, which required the proper feature selection from network flows. In another work, reference [18] proposed a dynamic ensemble incremental learning-based Network Intrusion Detection System (NIDS) for handling dynamic network changes using the KDDCUP dataset without considering real-time and complex network datasets. In [19], an intrusion detection method utilizing word embedding with the KNN method was proposed for payload network data, while the packet headers were ignored. Consequently, the proposed method was vulnerable to packet header scanning and probing attacks. Moreover, the maximum accuracy was limited to 92% on the CICIDS17 dataset.

The hierarchical CNN method proposed in [20] involved transforming packet bytes of CICIDS17 into images for intrusion detection. However, the proposed method was deemed inadequate due to insufficient training samples for certain abnormal types, highlighting the limitations of classification-based DL detection methods in detecting abnormal traffic within small sample categories. Additionally, the extraction and preprocessing of packet bytes necessitated high computational resources, despite achieving 90% accuracy [21]. In [22], sparse autoencoder and DNN methods were proposed for intrusion detection on the KDDCUP, NSLKDD, and UNSWNB15 datasets. This method utilized a sparse autoencoder to select features from network statistical features and employed a DNN for intrusion detection. However, as noted in [21], machine learning and neural network-based methods may lack robustness in adversarial environments. This is because various Generative Adversarial Network (GAN)-based methods can be used to alter network features to evade these models.

Table 1 presents a comprehensive summary of selected intrusion detection studies. The comparative analysis highlights the diverse array of IDS and their corresponding accuracies, particularly regarding system security. Existing systems have primarily focused on statistical features extracted from network flows. Yet, they fail to address the unique security challenges inherent in healthcare environments, where the integrity

and confidentiality of patient data and critical infrastructure are on top. Thus, this study proposed the automated IDS to strengthen the cyber defenses of the healthcare system. Moreover, the proposed method employed the DT, RF, and KNN models on KDDCUP and new synthetic HCKDDCUP datasets to ensure better security and stronger protection against network intrusions.

Table 1: Summary of selected intrusion detection studies

Study focus	Dataset	Method	Accuracy (%)
Detection of DDoS attacks using data mining [9]	Not specified	Data mining techniques	98.63
Evaluation of machine learning algorithms for IDS [10]	UNSWNB15	Machine learning algorithms	98.2
Ensemble method for intrusion detection [11]	KDDCUP, NSLKDD	Ensemble of DT, RF, KNN, DNN	85.2
Intrusion detection method utilizing word embedding with KNN [19]	UNSWNB15	Word embedding with KNN	99
Hierarchical CNN method for intrusion detection [20]	CICIDS17, CSECICIDS18	Hierarchical CNN	90
Sparse autoencoder and DNN methods for intrusion detection [22]	KDDCUP, NSLKDD, UNSWNB15	Sparse autoencoder, DNN	99
Proposed method	KDDCUP, HCKDDCUP	DT, RF, KNN	99

3 Proposed Methodology

The proposed methodology includes several key steps to develop healthcare-adapted IDS, as shown in Fig. 1, to strengthen the cyber defenses of the healthcare systems. The process begins by creating the HCKDDCUP dataset, specifically designed to mimic attributes and attack scenarios in healthcare systems. Moreover, the HCKDDCUP reflects the characteristics of the KDDCUP dataset while adapting the unique security requirement to ensure the comprehensive evaluation of the IDS algorithms that closely approximate the real-world attack scenarios of the healthcare systems. Consistency and accuracy of data would provide better results over IDS algorithms. Hence, data preprocessing includes data cleaning and encoding variable categories in the proposed method to eliminate any inconsistency from KDDCUP and HCKDDCUP datasets.

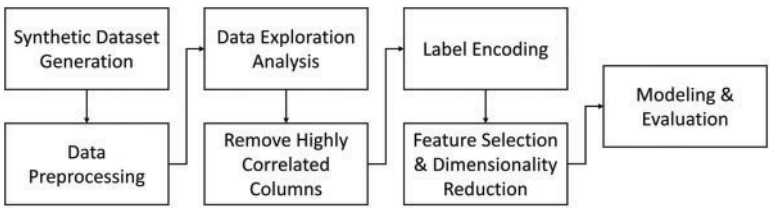

Figure 1: Proposed methodology

As an inevitable conclusion, there is a need to comprehend in-depth the data patterns and characteristics to conceive a meaningful IDS [23]. After preprocessing the KDDCUP and HCKDDCUP datasets, both were analyzed to provide distribution and prevalence information on different attack types through data exploring analysis. This analysis investigates the data, examining trends, anomalies, and likely security risks in the datasets. The subsequent necessary procedure is the elimination of highly correlated columns from the two datasets. This process optimizes the quality and reliability of data that may adversely affect the performance of machine learning algorithms. Furthermore, columns from both datasets will be numerically encoded using the label encoding technique. The label encoding process helps prevent compatibility issues by converting variable data into numerical form with various machine-learning models. Afterward, feature selection and dimensionality reduction techniques are employed to identify relevant features and reduce the complexities in both datasets. This step optimizes both datasets in selecting the most accurate features and reducing the dimensionality of both datasets for machine learning models. Finally, machine learning models, including RF, DT, and KNN, are trained on preprocessed datasets, and their performance in intrusion detection is evaluated to identify the most suitable algorithms for intrusion detection in healthcare systems, ultimately contributing to improving healthcare cyber defenses.

3.1 Synthetic HCKDDCUP Dataset Generation

In this study, we undertake an in-depth comparative analysis between a benchmarked KDDCUP dataset obtained from Kaggle and an HCKDDCUP dataset generated using the code outlined in Algorithm 1. While high-quality datasets are imperative for data-driven tasks and machine learning models, acquiring them from real-world sources can prove challenging and time-consuming [23,24]. Creating an HCKDDCUP dataset resembling the attributes and attack scenarios found in the KDDCUP dataset for healthcare involves several steps. Since the KDDCUP dataset pertains to network intrusion detection rather than healthcare, we must adjust the attributes and attack scenarios to suit the healthcare domain. The simplified code in Algorithm 1 outlines the process to generate the HCKDDCUP dataset that mirrors some attributes and attack scenarios like the KDDCUP dataset but is tailored for healthcare. Here, we define characteristics such as Patient_ID, Age, Gender, Diagnosis, and Treatment, which are relevant to healthcare. Meanwhile, the KDDCUP dataset comprises approximately 4.9 million vectors, each representing a single connection and consisting of 41 attributes that can be categorized as normal or indicating an attack [25]. The dataset features four main attack categories:

- **Denial of Service (DOS):** where the device's memory becomes overwhelmed, rendering it unable to respond to requests.
- **U2R Attack:** Involving a cybercriminal gaining access to a device and exploiting vulnerabilities to access the router.
- **R2L Attack:** occurring when a cybercriminal without device access sends packets from a computer to exploit system vulnerabilities and gain access to the device.
- **Probe Attack:** an attempt to obtain data from a computer network system to circumvent security controls.

Similar scenarios are defined within healthcare settings to adapt these attack scenarios [26,27]. The code then generates HCKDDCUP data for each attribute using appropriate distributions and combines them into a DataFrame representing the HCKDDCUP dataset. Additionally, the KDDCUP dataset from Kaggle is the benchmark in our analysis, comprising observations and measurements collected from a specific domain, thus representing real-world data. Conversely, the HCKDDCUP dataset is crafted by simulating data with predetermined patterns to replicate the characteristics and patterns observed in the KDDCUP dataset.

Algorithm 1: Generating HCKDDCUP dataset with attributes and attack scenarios

Input:

- `basic_attributes`: List of basic attributes relevant to healthcare.
- `traffic_attributes`: List of traffic attributes relevant to healthcare.
- `content_attributes`: List of content attributes relevant to healthcare.
- `attack_scenarios`: List of attack scenarios relevant to healthcare.
- `num_samples`: Number of synthetic samples to generate.

Initialization:

- Create an empty dictionary `HCKDDCUP_data` to store synthetic data.
- Create a sequence of patient IDs from 1 to num_samples.

Generating Synthetic Data:

for each attribute in the combined list of `basic_attributes, traffic_attributes,` and `content_attributes` **do**

if attribute is in `basic_attributes` **then**

Generate random integer values between 0 and 100.

else if attribute is in `traffic_attributes` **then**

Generate random integer values between 0 and 1000.

else if attribute is in `content_attributes` **then**

Randomly choose from {`tcp, udp, icmp`}.

else if

end for

Generating Attack Scenarios:

- Randomly select attack scenarios from `attack_scenarios` for each sample.

Combining Data:

- Add the attack scenario column to the `HCKDDCUP` data dictionary.

Output:

- Create a DataFram `HCKDDCUP_df` from the `HCKDDCUP` data dictionary.

Furthermore, the attributes in the KDDCUP dataset are categorized into three groups: basic attributes, traffic attributes, and content attributes. The basic attributes category encompasses features that can be extracted from a TCP/IP connection, many contributing to detection delays. The traffic attributes category involves attributes calculated based on a window interval, divided into subcategories such as same host attributes and same service attributes, both of which are time-based. However, certain slow-moving probe attacks may not create intrusion patterns within a two-second window. To address these attributes, such as the same host and same service, are recomputed based on a connection window consisting of 100 connections, known as connection-based traffic attributes. Despite some attack types like U2R and R2L attacks lacking frequent sequential intrusion patterns, they often exhibit specific behaviors within the data portion of a packet. Detecting these attacks requires attributes that inspect suspicious behavior within the data portion, known as content attributes. The proposed work generates the HCKDDCUP dataset by adopting these attack attributes and scenarios in healthcare settings to facilitate comprehensive comparative analysis and enhance the reliability of data-driven tasks and machine learning models.

3.2 Data Preprocessing

During the data preprocessing phase, several key steps were taken to prepare the dataset for analysis and modeling. First, a list of corresponding column names was generated by extracting non-empty columns.

The process commenced by identifying and extracting non-empty columns to ensure only relevant data was retained as:

$$C = \{c \in \text{Columns} \mid \text{non-empty values exist in } c\} \tag{1}$$

where C represents the selected column names. This phase focused on critical provisions only to be carried out in the later stages of the data processing activity. Subsequently, an attack type dictionary was created for mapping attack types to categorical groups:

$$M : \text{Attack Type} \rightarrow \text{Categorical Group} \tag{2}$$

This dictionary identified the varieties of attacks and placed them in categorically ordered groups, thus making necessary classifications of the attacks within the dataset. This mapping was important for the right understanding and interpreting attack classifications. Afterward, the dataset was pulled by identifying the file path P and the column names selected from the created columns C and marking the missing data with a question mark '?' to facilitate and maintain the proper loading structure of the dataset, all of which form the basics of the other preprocessing steps to come.

$$D = \text{Load}(P, C), \; D[c][\text{missing}] = ? \tag{3}$$

Also, to improve the analysis of the data within the database, a new column with attack type was included using a lambda function to apply the mapping M:

$$D[\text{Attack Label}] = D[\text{Attack Type}].\text{apply}(\lambda x : M[x]) \tag{4}$$

It used the lambda function and attack type dictionary to label the target column for clear attack representation and easier classification in the analysis processes. A further HCKDDCUP dataset was created by performing additional filtering of the KDDCUP dataset based on specific attacks and their sources. A filtering step was applied to refine the dataset further. Rows were selected based on logical conditions L, such as specific attack types or sources. The '*loc*' function leads to the formation of a new dataset consisting of the filtered data:

$$D_{\text{filtered}} = D.\text{loc}[L] \tag{5}$$

Resulting in a dataset focused on meaningful subsets for targeted analysis. The filtered data includes several types of attacks, which allows for further analysis and modeling tasks. Numerical encoding was done to convert attack types into numerical forms by assigning each attack type an index based on its position in the attack type dictionary:

$$D_{\text{filtered}}[\text{Attack Encoded}] = D_{\text{filtered}}[\text{Attack Type}].\text{apply}(\lambda x : \text{index}(M[x])) \tag{6}$$

The process was done by combinations of '*apply*' and '*lambda*' functions, with each element denoting an attack type to which index the attack type is present in the attack types of lists. This approach made it possible to encode the attack types, so they were compatible with machine learning models. A sample of the processed dataset was shown to confirm that preprocessing was successful:

$$D_{\text{sample}} = D_{\text{filtered}}[: n] \tag{7}$$

Here, n represents the number of rows displayed. This stage of data preprocessing involves selecting relevant columns, mapping various attack types, loading the dataset, adding an attack type column, filtering, quantifying attack types (including encoding), and showing the transformed dataset. These steps ensured a clean, structured dataset ready for exploratory data analysis and machine learning tasks.

3.3 Data Exploration Analysis

In the exploration analysis, both KDDCUP and HCKDCCUP were examined for the distribution of number of attack types. General exploratory analysis serves the wide view of distribution and incidence occurrence with respect to the presence of attack types in datasets that are very useful to understand the structure and occurrence of its type. The frequency distribution of each attack type a_i was computed as $f(a_i)$, where a_i denotes a unique attack type. The frequency for each attack type was calculated using the $Kronecker\ delta$ function:

$$f(a_i) = \sum_{j=1}^{N} \delta(D_j[\text{Attack Type}], a_i) \tag{8}$$

where, N is the total number of records in the dataset, $\delta(x, y)$ is the $Kronecker\ delta$ function, which equals 1 if $x = y$, and 0 otherwise to count the occurrences of each attack type a_i. The proportion $p(a_i)$ of each attack type was then derived by dividing the frequency $f(a_i)$ by the total frequency of all attack types $\sum_{k=1}^{K} f(a_k)$:

$$p(a_i) = \frac{f(a_i)}{\sum_{k=1}^{K} f(a_k)} \tag{9}$$

where K is the total number of unique attack types. The Eq. (9) gives the relative distribution of each attack type and is thus useful in establishing the understanding of different attacks in the datasets. To visualize these distributions, bar charts regarding attack type frequencies were produced as shown in Fig. 2 using $Seaborn$. The height of each bar in the chart corresponds to the $f(a_i)$ frequency of every attack type a_i to compare the relative occurrences of attack types in both datasets. These charts revealed that the KDDCUP dataset had a higher proportion of DOS attacks, while HCKDDCUP dataset displayed a more well-distributed scenario of attack types with respect to one another.

Next, the principal component explained variance ratio was analyzed. The explained variance ratio for each principal component i was computed as:

$$\text{Variance Ratio} = \frac{\lambda_i}{\lambda_{\text{total}}} \tag{10}$$

where λ_i is the eigenvalue of the ith principal component, and λ_{total} is the sum of all eigenvalues. This ratio indicates the contribution of each feature (e.g., protocol_type, wrong_fragment, Attack Type) to the total variance in the dataset. For example, the Attack Type component had an explained variance ratio of 0.049 for both datasets, showing that the attack type feature contributed similarly to the variance in both datasets. Under observations, it was found that KDDCUP dataset contains a higher percentage of DOS attacks as compared to HCKDDCUP dataset. The basis for all these insights was through frequency calculations, proportions, and variance ratios, the most important concepts in understanding the structure and relationships among the datasets. Such analysis helps underpin the nature of the given datasets to be handy for any future compared to modeling or analysis.

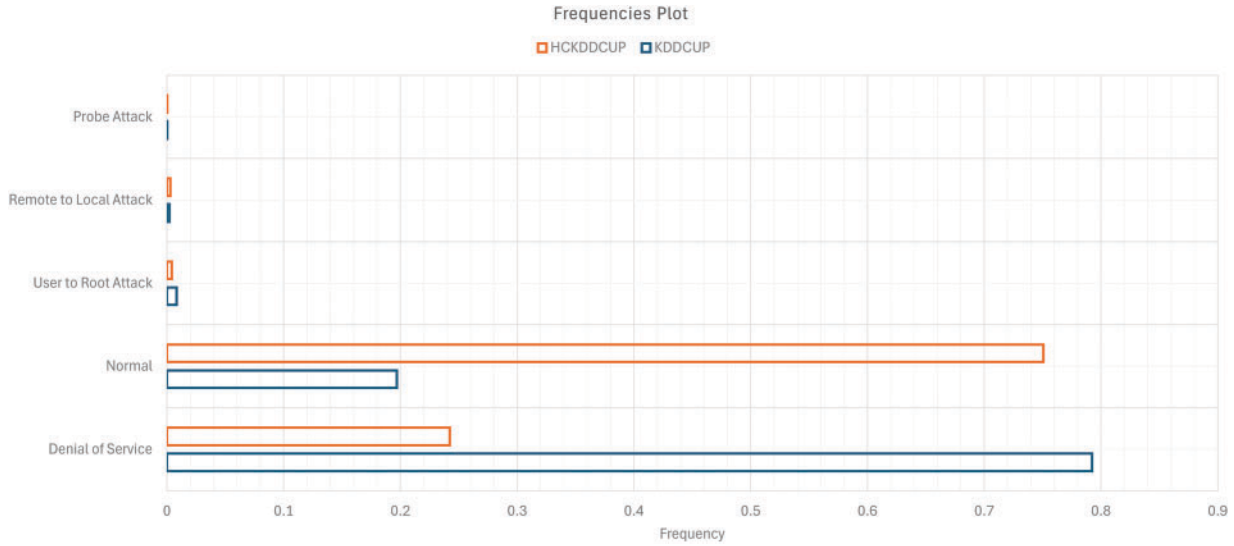

Figure 2: Bar charts comparing the frequency distribution of attack types in the KDDCUP and HCKDDCUP datasets

3.4 Remove Highly Correlated Columns

This phase aims to evaluate all datatypes in both datasets and perform necessary preprocessing for KDDCUP and HCKDDCUP . Initially, a Pearson correlation analysis was conducted to detect multicollinearity among features and ensure data integrity. The resulting correlation matrices are visualized as heatmaps in Figs. 3 and 4, with proper axis labels and color bars for interpretation.

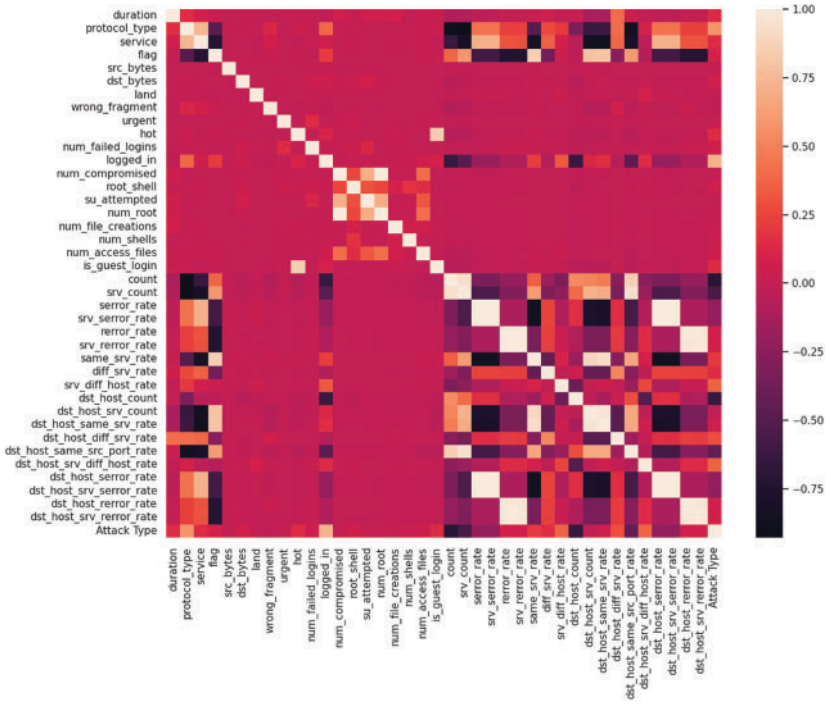

Figure 3: Correlation Heatmap for the KDDCUP dataset with labeled axes and a color legend. Features with high correlation are easily identifiable

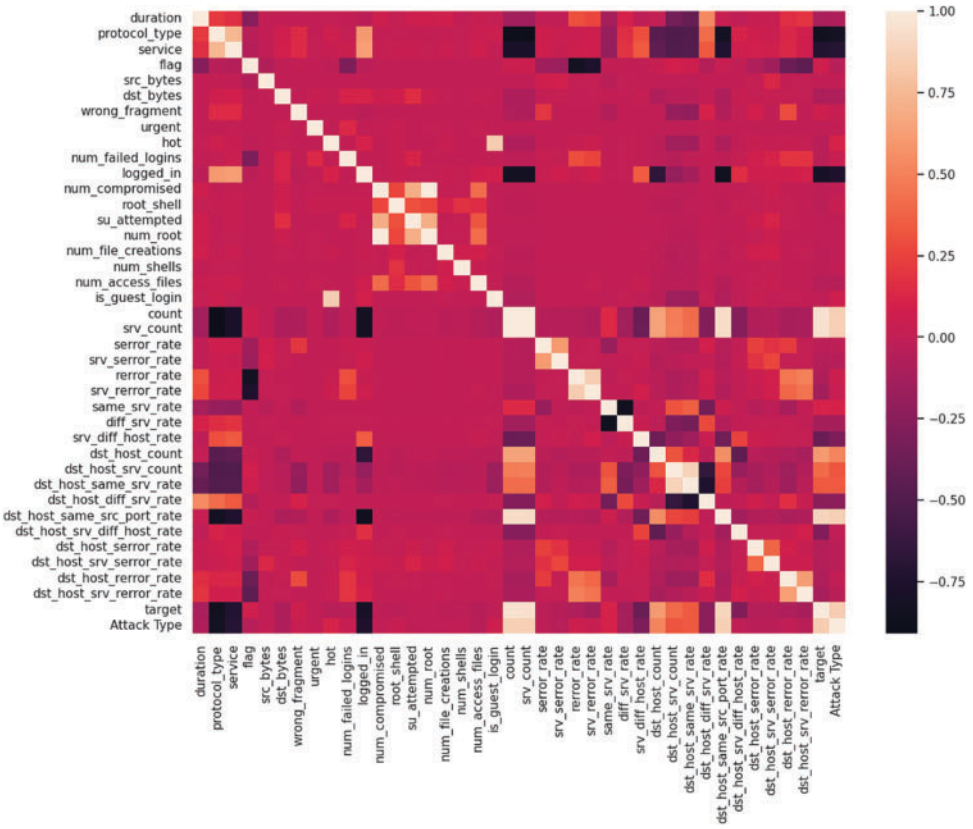

Figure 4: Correlation Heatmap for the HCKDDCUP dataset illustrating variable relationships and dependencies with proper labels and legend

Fig. 3 shows the correlation heatmap for the KDDCUP dataset, while Fig. 4 presents the same for HCKDDCUP. Each heatmap is 13 × 10 in size, with bright-colored cells indicating stronger correlations. The heatmaps were generated using the Seaborn library with a diverging color palette, and the axes are labeled with the respective feature names for clear reference. A legend is included to help interpret the correlation values from −1 to 1.

The Pearson correlation coefficient between two features X_i and X_j was calculated using:

$$\text{corr}(X_i, X_j) = \frac{\text{cov}(X_i, X_j)}{\sigma_{X_i}\sigma_{X_j}} \tag{11}$$

where $\text{cov}(X_i, X_j)$ is the covariance between the features, and σ_{X_i} and σ_{X_j} are their standard deviations. If $\text{corr}(X_i, X_j) > 0.9$, one of the correlated features was removed to reduce redundancy.

Additionally, a data quality report was generated to detect missing values, high cardinality, and low variance features. For missing values, the number of nulls in column D_i, denoted as m_i, was computed by:

$$m_i = \sum_{j=1}^{N} 1(D_j[\text{Column}] = \text{NaN}) \tag{12}$$

where N is the total number of records and $1(x)$ is the indicator function. Columns with significant missing values or negligible variance were removed.

The data quality was reassessed post-cleaning using an updated report with the quality parameter set to level 1. This reevaluation helped validate the impact of preprocessing and ensured that only relevant and clean features were passed to subsequent modeling stages.

3.5 Label Encoding

Encoding categorical variables into numerical format is essential for machine learning algorithms, which typically operate on numerical inputs. In this work, label encoding was applied to categorical columns such as "$protocol\_type$", "$service$", and "$flag$" in both the KDDCUP and HCKDDCUP datasets. These features were transformed using the LabelEncoder class from the `scikit-learn` library, which assigns a unique integer to each category. For instance, in the $protocol\_type$ column:

- tcp → 0
- udp → 1
- icmp → 2

This transformation ensures compatibility with models that require numeric inputs. After encoding, the $attack\_type$ column was removed from the KDDCUP dataset to prevent data leakage during training. For the HCKDDCUP dataset, however, this column was retained, as the dataset is intended for use with its target variable intact for subsequent analysis. The encoding process, therefore, involves only transforming the categorical columns without modifying the target column. The label encoding process is summarized in Algorithm 2, avoiding redundancy while keeping the methodology complete.

Algorithm 2: Label encoding process

Inputs:

- `KDDCUP` and `HCKDDCUP Datasets`: The datasets with categorical columns requiring encoding.

Outputs:

- `Encoded Datasets`: `KDDCUP` and `HCKDDCUP` datasets with categorical columns replaced by numerical representations.

Step 1: Identify Categorical Columns

- Extract the list of categorical columns to be encoded from the specified column list of the `KDDCUP` dataset.

Step 2: Encode Categorical Columns

for each categorical column in the column list **do**

Apply the `fit_transform` method from the `LabelEncoder` in the `scikit-learn` library to encode the column values.

Replace the original categorical values with corresponding numerical representations.

end for

Step 3: Drop Target Column (KDDCUP Dataset)

- Drop the target column from the `KDDCUP` dataset using the axis=1 parameter, indicating column-wise operation.

Step 4: Finalize Encoding (HCKDDCUP Dataset)

- Apply the same encoding process (steps 1 and 2) to the `HCKDDCUP` dataset without dropping the target column (step 3).

3.6 Feature Selection and Dimensionality Reduction

This phase optimizes the KDDCUP and HCKDDCUP datasets through feature selection and dimensionality reduction. The aim is to retain the most relevant features while reducing dataset complexity [28]. The improved process involves the SelectKBest method for feature selection and Principal Component Analysis (PCA) for dimensionality reduction, as outlined in Algorithm 3.

Algorithm 3: Improved feature selection and dimensionality reduction process

Input:

- `KDDCUP` and `HCKDDCUP` datasets
- Number of top features to select (k = 10)

Output:

- Selected features from both datasets are identified and scaled with reduced dimensionality.

Step 1: Feature Selection

- Select the top K features with `chi-squared` (chi2) scoring from each dataset.
- Scale the selected features using the `min-max` scaler to ensure non-negative values.

Step 2: Standardization

- Scale both datasets using the `standard_scaler` method to standardize feature distributions.

Step 3: Dimensionality Reduction with PCA

- Configure PCA to select the top 10 components.
- Apply PCA to reduce the dimensionality of both datasets.

The $SelectKBest$ method evaluates each feature's significance using a statistical scoring function. Here, the chi-squared test was used to measure the dependency between each feature and the target variable. Features with the highest scores are considered the most important for prediction. The $chi-squared$ statistics for a feature were calculated using the formula:

$$X^2 = \sum \frac{(O_i - E_i)^2}{E_i} \tag{13}$$

where, O_i represents the observed frequency and E_i represents the expected frequency for category i. Using this method, the top 10 features from the dataset were selected. These selected features are then scaled using the Min-Max Scaler, which transforms the values into a fixed range (typically 0 to 1) to ensure non-negative values. Once the relevant features are selected, the datasets (KDDCUP and HCKDDCUP) are standardized using the Standard Scaler. Standardization ensures that features are centered around zero with a standard deviation of one, helping to avoid discrepancies in feature distributions. The standardized value of a feature is calculated as:

$$x_{\text{standardized}} = \frac{x - \mu}{\sigma} \tag{14}$$

where, x is the original feature value, μ is the meaning of the feature, σ is the standard deviation of the feature. This ensures all features are on a comparable scale, which is essential for algorithms that rely on distances, such as KNN. After feature selection and standardization, Principal Component Analysis (PCA) is applied to reduce dataset dimensionality. PCA transforms the original features into a new set of orthogonal components ordered by the amount of variance each component explains. The first few components capture most of the

variance, allowing the dataset to be reduced to fewer dimensions without losing significant information. The PCA transformation is represented as:

$$X_{\text{reduced}} = X \cdot W_k \tag{15}$$

where, X is the original data matrix (after scaling), W_k is the matrix of the top k eigenvectors (principal components), X_{reduced} is the reduced data matrix with fewer dimensions. For this analysis, PCA retains the first 10 components. The explained variance ratios for these components are shown in Fig. 5.

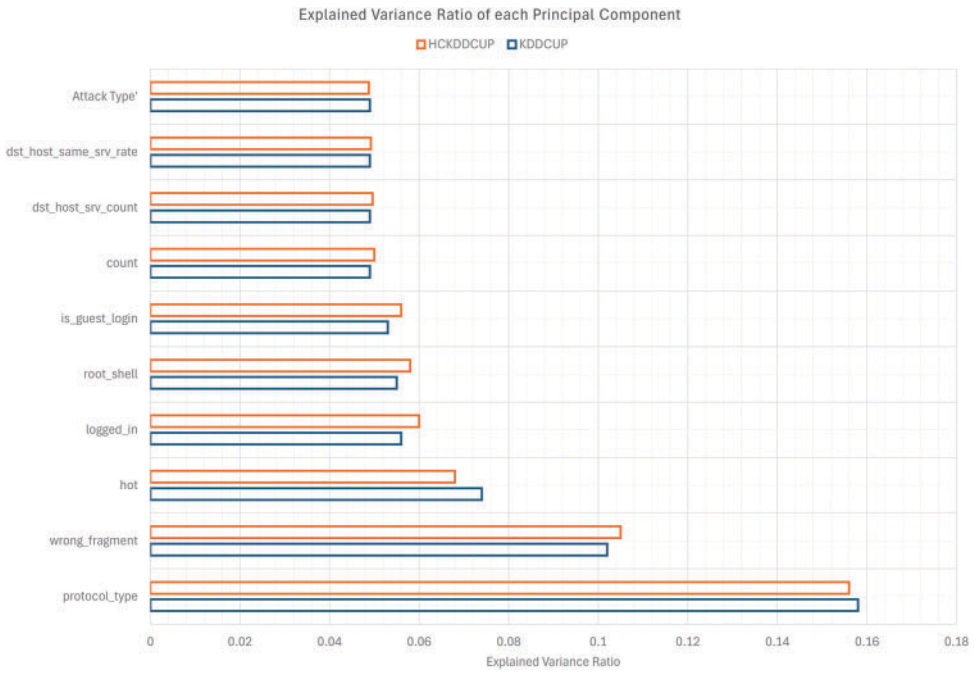

Figure 5: Explained variance ratios of the top 10 principal components from PCA applied to the KDDCUP and HCKDDCUP datasets

The feature selection and dimensionality reduction methods have been used in this phase to optimize datasets (KDDCUP and HCKDDCUP) for machine learning tasks. Applying a *chi – squared* test to top ten valuable features and dimensionality reduction using PCA, these datasets retain the most essential information. This made both datasets more efficient and applicable for analyzing models without too high complexity while still preserving variance. Eventually, the output datasets have been produced as a final set of datasets with dimensionality reduction so that future machine learning operations over them become better in performance interpretability.

3.7 Modeling and Evaluation

The use of machine learning classifiers is of utmost importance in the intrusion detection systems as they facilitate automatic examination of network traffic to locate possible security breaches. For this section, we explore a closer detail of the machine learning classifiers that include RF, DT, and KNN [29]. Understanding these algorithms is vital for the purpose of value assessment concerning the performed results from using them and for intrusion detection capability. RF is an example of an ensemble learning algorithm that consists of several different DTs as compared to the conventional learning algorithms. A collection of tree-like things, called DT, is constructed where each tree is trained on a different portion of the data. The predictions from

different built trees are aggregated using methods like majority voting or averaging to result in the final prediction. Many advantages exist to using RF in intrusion detection like:

- **Robustness:** RF is resilient to overfitting and adept at handling noisy and complex datasets.
- **Feature Importance:** RF quantifies feature importance, facilitating the identification of critical features for intrusion detection.
- **Parallelization:** RF can be parallelized for efficient processing of large-scale datasets.

On the other hand, DT is a very simple and straightforward supervised learning technique which can support modeling of decisions and results in tree structure. This algorithm splits the dataset based on features and forms a set of decision rules at every internal node to decide the corresponding class label. The following are the highlights of DT for intrusion detection system:

- **Interpretability:** DT provides transparent and easily interpretable rules, aiding in understanding intrusion detection decisions.
- **Nonlinear Relationships:** DT captures nonlinear relationships between features and class labels, making it effective in detecting complex attack patterns.
- **Overfitting:** DT is susceptible to overfitting if not properly pruned or regularized, which can impede generalization on unseen data.

However, KNN is a non-parametric algorithm that classifies data points based on their proximity to other data points in the feature space. When presented with a new data point, KNN identifies its nearest neighbors in the training dataset and assigns a class label based on majority voting. Key features of KNN for intrusion detection include:

- **Flexibility:** KNN accommodates both binary and multi-class classification problems and adapts to diverse attack scenarios.
- **Local Patterns:** KNN considers the local structure of the data, making it proficient in detecting anomalies or intrusions deviating from normal behavior.
- **Computational Intensity:** KNN necessitates storing the entire training dataset, rendering it memory-intensive and computationally demanding, especially for large-scale datasets.

Understanding these algorithms is pivotal for assessing their efficacy in detecting and preventing network attacks. The section of the paper has discussed at length the strong points of RF, DT, KNN in robustness, interpretability, and flexibility. This will serve as a basis for evaluating these classifiers on intrusion detection datasets for classifier selection depending on the precise security requirements of healthcare systems. The experiment conditions were effectively designed to achieve full observation and measurement, including modeling and analysis on the KDDCUP and HCKDDCUP datasets. Algorithm 4 provides machine learning model evaluation process. The primary goal was to allow strict, systematic, and quantitative assessment of the performance differences between different machine learning algorithms and their effectiveness in detection and response. In doing so, the datasets were first partitioned into training and testing datasets, which provided a strong evaluation framework. After that, Parameter grids were created for RF, DT, and KNN classifiers to test many hyperparameter combinations.

Algorithm 4: Experiment scenarios for machine learning model evaluation

Input:

- KDDCUP dataset, HCKDDCUP dataset

Output:

- Best parameters and cross-validated accuracy scores for RF, DT, and KNN classifiers.

Step 1: Split the KDDCUP and HCKDDCUP datasets into training and testing sets

- Use train_test_split function with a test size of 20% and random state of 42.
- Ensure consistency by rearranging the column order to match between test and training sets.

Step 2: Define parameter grids for RF, DT, and KNN classifiers

- Specify a range of hyperparameters for each classifier.

Step 3: Perform grid search with cross-validation to find the best parameters for each classifier

- Set cross-validation folds to 2 and use accuracy score as the evaluation metric.

Step 4: Fit the training data for each classifier using grid search with cross-validation

- Search for the best parameters that maximize the accuracy score.
- Employ cross-validation to assess model performance.

Step 5: Verify the parameters and corresponding mean cross-validated accuracy scores for each classifier.

Afterward, a grid search with cross-validation was performed to find the best parameters for each classifier. This involved exploring various combinations of parameters while measuring the performance with the use of cross-validation folds. The accuracy score was chosen as an evaluation metric since it enables effective assessment of the classifiers' ability to predict class labels. Cross-validation was used to polish their hyper parameters to ensure the classifiers performed optimally in predicting the training data. In the subsequent stage after the parameter's optimization, both the KDDCUP and HCKDDCUP datasets underwent train-test splits using the train-test split function. This facilitated the inclusion of a fair share of the data for the purpose of blind testing, thus helping reduce overfitting. Further, the order of columns in the training and testing datasets was also kept the same to ensure there were no discrepancies in the datasets.

With the datasets appropriately prepared, the grid parameters were defined to explore various hyperparameter configurations for RF, DT, and KNN classifiers. Cross-validation folds were set to 2 to balance computational efficiency with reliable parameter selection. The evaluation metric was consistently applied to assess the classifiers' performance based on their accuracy scores. Through this careful process, insights into the optimal hyperparameters and expected performance on the training data were garnered as given in Table 2, laying the foundation for robust model evaluation and comparison. For the RF, DT, and KNN models, the estimators that proved to be the best after a grid search with a cross-validation routine were used for comparative analysis purposes. It indicates that these estimators are models that have been tuned for optimal hyperparameters.

In the subsequent phase of the study, the classifiers RF, DT, and KNN were trained using both the KDDCUP and HCKDDCUP datasets. Following training, predictions were generated for their respective test datasets. To evaluate the performance of each classifier, accuracy control metrics were calculated using the standard accuracy scoring functions. Table 3 presents the accuracy and other performance metrics for each classifier on the KDDCUP dataset, while Table 4 provides corresponding results for the HCKDDCUP dataset. This comparative analysis facilitates a deeper understanding of each model's generalization capabilities and highlights potential overfitting on the balanced HCKDDCUP dataset in comparison to the original KDDCUP dataset.

Table 2: Best parameters and the corresponding mean cross-validated for each classifier

Classifier	Best parameters	Best cross-validated accuracy
RF	{'max_depth': 10, 'min_samples_leaf': 1, 'min_samples_split': 5, 'n_estimators': 50}	0.999885991
DT	{'criterion': 'gini', 'max_depth': 50, 'max_features': 'auto', 'min_samples_leaf': 1, 'min_samples_split': 2, 'splitter': 'random'}	0.999729231
KNN	{'algorithm': 'ball_tree', 'n_neighbors': 3, 'weights': 'distance'}	0.999729231

Table 3: Performance metrics on KDDCUP dataset

Metric	RF	DT	KNN
Accuracy	0.986194	0.985130	0.980459
Precision	0.945150	0.885154	0.839249
Recall	0.828830	0.845338	0.817641
F1-score	0.870354	0.860005	0.826714
AUC	0.996203	0.993743	0.989091

Table 4: Performance metrics on HCKDDCUP dataset

Metric	RF	DT	KNN
Accuracy	0.999658	0.999544	0.999886
Precision	0.995376	0.997752	0.997661
Recall	0.997889	0.998405	0.998945
F1-score	0.996600	0.998072	0.998294
AUC	0.981452	0.992541	0.994673

In order to address the issue of class imbalance and provide a comprehensive assessment of model performance, five key evaluation metrics were used: Accuracy, Precision, Recall, F1-Score, and AUC (Area Under the Curve). Moreover, confusion matrices were generated for each classifier to visualize their classification behavior in terms of true positives, false positives, false negatives, and true negatives. Figs. 6 and 7 display the confusion matrices for RF, DT, and KNN, respectively. These visualizations help illustrate the ability of each model to distinguish between attack and normal traffic classes, as well as highlight tendencies toward false positives or false negatives, a crucial consideration in imbalanced datasets like intrusion detection.

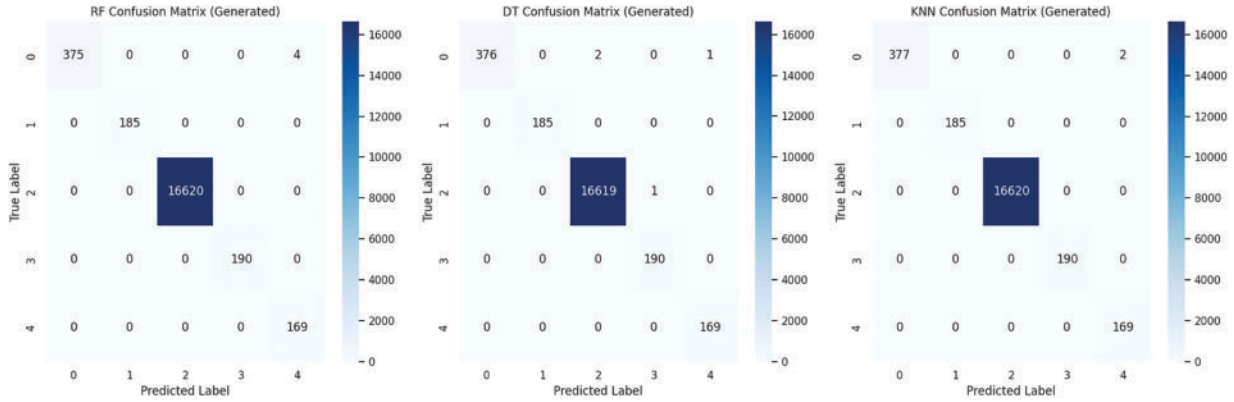

Figure 6: Confusion matrices for RF, DT, and KNN on HCKDDCUP dataset

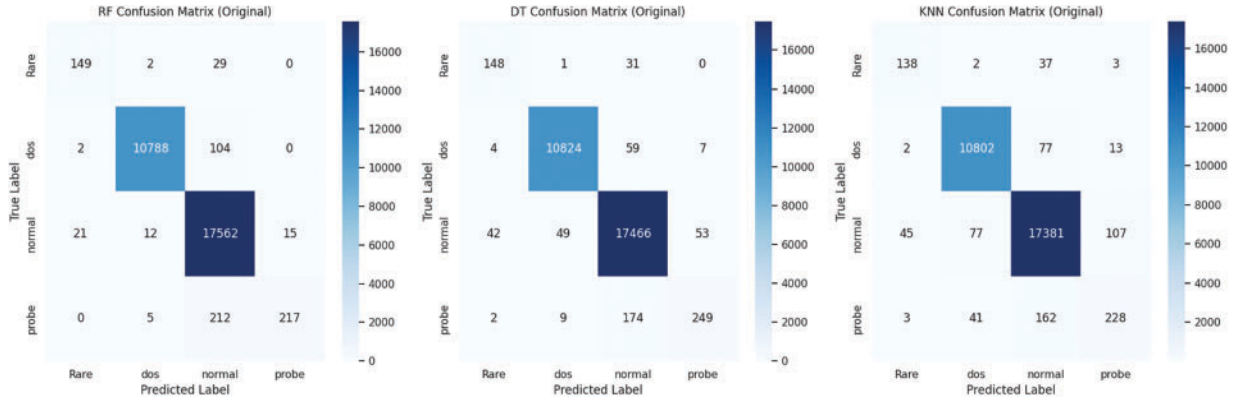

Figure 7: Confusion matrices for RF, DT, and KNN on KDDCUP dataset

Tables 3 and 4 summarize the classification performance across both datasets. On the KDDCUP dataset, RF attained the highest precision (0.945), indicating its strong ability to minimize false positives. However, its recall (0.829) was slightly lower than that of DT (0.845), which demonstrates better detection of actual attack instances. The F1-scores of RF (0.870), DT (0.860), and KNN (0.827) illustrate the balance each model strikes between precision and recall. In terms of overall accuracy, RF led with a value of 0.986, followed closely by DT (0.985) and KNN (0.980). The lower recall of KNN (0.818) suggests a challenge in correctly identifying all attack cases, potentially due to dataset imbalance.

Significant performance improvements were observed for all classifiers on the HCKDDCUP dataset. This indicates that HCKDDCUP offered a more balanced and representative distribution of classes, enabling the classifiers to generalize better and yield more accurate predictions.

- **Performance Improvement:** All models exhibited improvements across all evaluation metrics when applied to the HCKDDCUP dataset. For instance, RF's F1-score improved from 0.870 on KDDCUP to 0.997 on HCKDDCUP, indicating enhanced generalization under more favorable data conditions.
- **Potential Overfitting:** The near-perfect results obtained on the HCKDDCUP dataset raise concerns about potential overfitting, particularly if the dataset lacks the variability and complexity of real-world network traffic. The narrow performance gap among all classifiers further supports this hypothesis.

- **Model Ranking Stability:** While RF maintained top-tier performance across both datasets, KNN displayed substantial gains on HCKDDCUP, even achieving the highest accuracy and recall. This underscores the impact of data quality on model effectiveness.

This comparative analysis emphasizes the importance of data quality and preprocessing in maximizing model performance. While RF was uniformly strong, the enhancement of DT and KNN performance on the HCKDDCUP dataset indicates that less complex models can compete when used with well-prepared data. For deployment into real-world intrusion detection systems, it is suggested to have additional validation on more heterogeneous datasets to prevent overfitting and to guarantee robust generalization abilities.

4 Discussion

The growing dependence on digital infrastructure within healthcare has rendered the industry a high-priority target for advanced cyberattacks, threatening patient confidentiality, core operations, and overall system security. In light of these emerging threats, this work presents a targeted assessment of intrusion detection systems (IDS) in healthcare settings. A significant contribution is the construction of the HCKDDCUP dataset, a synthetic benchmark that aims to mimic healthcare-specific attack behaviors by augmenting the popular KDDCUP dataset.

4.1 Key Findings

Data preprocessing was the first step in the multi-step research methodology, which was used to guarantee consistency and quality across the two datasets. While feature selection and dimensionality reduction techniques optimized the datasets for machine learning modeling, exploratory analysis offered important insights into the distribution of attack types. To determine how well RF, DT, and KNN models could identify intrusions and generalize to new data, the study tested their performance on both datasets. The results showed a number of important insights:

- RF consistently demonstrated superior performance, achieving an accuracy of 0.986 on the KDDCUP dataset and 0.9997 on the HCKDDCUP dataset. It also achieved the highest precision (0.945) and a strong F1-score (0.870) on KDDCUP, highlighting its effectiveness in reducing false positives while maintaining a balanced performance. Its ensemble nature, robustness, and ability to handle complex, high-dimensional data made it the most effective model for intrusion detection in this context.
- DT showed competitive performance, with an accuracy of 0.985 on KDDCUP and 0.9995 on HCKDDCUP. Its recall on KDDCUP (0.845) surpassed that of RF, indicating better sensitivity to detecting true attack instances. While DT performed well across both datasets, the slight performance improvement on HCKDDCUP and consistently high recall values support its reliability, especially when interpretability and simplicity are prioritized.
- KNN exhibited lower performance on KDDCUP with an accuracy of 0.980 and a recall of 0.818, indicating limitations in detecting all attack classes under imbalanced conditions. However, on HCKDDCUP, KNN's accuracy rose to 0.9999, with the highest recall (0.999) and F1-score (0.998), suggesting strong generalization on balanced datasets. These results underscore KNN's sensitivity to data quality and distribution, where high-quality preprocessing can significantly enhance its classification effectiveness.

4.2 Limitations

The experimental results underscored the superior performance of the RF model, which achieved the highest accuracy across both datasets, emphasizing its potential for real-world applications in enhancing healthcare cybersecurity. DT also demonstrated promising results, offering interpretability and efficiency,

though its performance slightly declined on the more complex HCKDDCUP dataset. KNN, while competitive, showed limitations in precision and computational efficiency, particularly on the healthcare-specific dataset. Overall, the study offers valuable insights; however, several limitations must be acknowledged:

- **Dataset Complexity:** The HCKDDCUP dataset introduced realistic attack patterns but also increased complexity, which exposed model limitations.
- **Generalization:** Certain models, particularly DT and KNN, exhibited reduced performance on the HCKDDCUP dataset, indicating the need for further optimization.
- **Computational Demands:** KNN's reliance on the entire training dataset rendered it computationally intensive, especially for large-scale implementations.
- **Class Imbalance:** The datasets exhibited some degree of class imbalance, which may have impacted model performance.
- **Deep Learning Omission:** Deep learning models such as CNN or LSTM were not included due to scope constraints. This is acknowledged as a limitation and is proposed as a future enhancement.
- **External Validation:** The HCKDDCUP dataset lacks formal validation from clinical or domain experts. Involving hospital IT staff or healthcare professionals is essential to ensure its realism.

4.3 Dataset Impact

The HCKDDCUP dataset provided a more balanced and representative distribution of attack and normal instances, leading to significantly improved model performance across all classifiers. It introduced more diverse and realistic healthcare-related attack patterns, offering a significant advancement in IDS research. All models showed notable improvements in accuracy, precision, recall, and F1-score when trained on HCKDDCUP compared to KDDCUP, with RF achieving an F1-score of 0.997, and KNN achieving the highest accuracy (0.9999).

This performance enhancement indicates that the HCKDDCUP dataset was effective in mitigating class imbalance and improving classifier generalization. Nonetheless, the complexity of this dataset revealed weaknesses in some models, particularly DT and KNN, reinforcing the need for optimizations specific to domain. Moreover, usage constraints created by computational resources present challenges when deploying IDS in resource-limited healthcare environments.

This research has wider implications for developing IDS algorithms to meet the specific requirements of a healthcare system. RF was revealed to be a strong candidate for deployment, with high reliability, and accuracy, whereas DT provides a compelling alternative in cases where an explainability is required. Furthermore, while KNN showed improvement in performance on the HCKDDCUP dataset, particularly in terms of recall and F1-score. Further, this study exemplified the benefit of balanced, domain-targeted data to increase usefulness of the models.

The HCKDDCUP dataset contributes to the progress of IDS research; however, it needs to be validated with an expert review or partially benchmarked with empirical datasets, such as MIMIC or CICIDS2017, to establish credibility and acceptance.

4.4 Implications for Healthcare Cybersecurity

The findings highlight the critical role of robust machine learning models in strengthening healthcare cyber defenses. RF turned out to be the most promising option for use in practical applications due to its highest accuracy. DT, as an interpretable model, presents a complementary option for situations where explainability is a concern. The outcomes, however, show that more work must be done to enhance generalizability, particularly on the HCKDDCUP dataset. Key implications include:

- **Real-World Applicability:** RF emerged as the most promising candidate for deployment in healthcare settings due to its high accuracy and reliability.
- **Interpretability:** DT provides a complementary alternative for scenarios prioritizing explainability, aiding decision-makers in understanding intrusion detection processes.
- **Explainability:** DT models provide transparency in decision-making, making them suitable for settings where interpretability is essential for trust and compliance.
- **Need for Generalizability:** The current comparison is limited to the KDDCUP and HCKDDCUP datasets. To enhance generalizability, future studies should include real-world datasets such as UNSW-NB15, CICIDS2017, or CSE-CIC-IDS2018.
- **Dataset Validation:** Synthetic healthcare-specific datasets like HCKDDCUP must be validated by domain experts or supplemented with partial use of real datasets to ensure practical relevance.
- **Targeted Development:** The study highlights the need for IDS solutions tailored specifically to healthcare environments, addressing unique cyberattack vectors and operational constraints.
- **Adversarial Robustness:** Evaluating how models perform under evasion and poisoning attacks is essential to ensure IDS reliability under adversarial conditions.
- **Federated Learning Potential:** Federated learning approaches can help develop privacy-preserving IDS across hospitals without sharing sensitive patient data.

4.5 Challenges and Future Directions

While this study demonstrates the potential of classical machine learning algorithms, particularly RF and DT, for healthcare intrusion detection, several challenges and opportunities for future work remain. We outline key directions to advance the development and deployment of intelligent IDS frameworks in real-world healthcare settings:

1. **Dataset Refinement and Domain Validation:** Although HCKDDCUP was constructed using healthcare schema references and known attack signatures, it remains a synthetic dataset. Future efforts should focus on enhancing its realism by incorporating feedback from hospital IT staff and clinical domain experts. This will improve its alignment with actual Electronic Health Record (EHR) systems and Health Information Exchanges (HIEs). Class imbalance, noise, and rare attack representations must also be addressed to ensure robust model generalization.
2. **Application to Real Hospital Network Data:** To bridge the gap between simulation and practice, our next phase will involve applying and validating the proposed models using real or semi-real hospital network traffic data, potentially obtained from pilot deployments in controlled healthcare environments. This will also allow the evaluation of false positive and false negative impacts in realistic clinical workflows.
3. **Deep Learning and Transformer-Based IDS:** Future work should incorporate and benchmark deep learning models such as CNNs, LSTMs, and Transformer-based architectures. These models can learn temporal and spatial attack patterns from raw traffic and log data, potentially outperforming classical models in complex threat scenarios.
4. **Adversarial Robustness Testing:** Current models have not been tested against adversarial attacks. As attackers increasingly employ evasion and poisoning techniques, future research must include experiments with adversarial scenarios such as Fast Gradient Sign Method (FGSM), label flipping, and GAN-based attacks. Developing robust detection mechanisms against such threats will be crucial.
5. **Time-Series and Streaming Data Integration:** HCKDDCUP, like many benchmark datasets, lacks temporal continuity. Future datasets and models should support time-series and streaming data to enable real-time detection, sequence-aware learning, and improved incident forensics.

6. **Federated Learning for Privacy-Preserving IDS:** In the healthcare domain, data privacy regulations limit data sharing across institutions. Federated learning offers a promising solution by allowing decentralized model training without exchanging raw data. Future work will explore federated IDS frameworks enabling collaborative defense while preserving patient confidentiality.
7. **Virtualized Testing Environments:** Models must be tested in simulated testbeds prior to deployment in the real world, using systems such as OpenMRS, GNS3, and Docker-emulated hospital setups. This transitional step will enable rigorous performance testing under near-realistic scenarios, such as response time, scalability, and alert prioritization.
8. **Scalability and Deployment Efficiency:** Practical deployment of IDS in hospitals requires scalable, lightweight, and computationally lean solutions. Future work should prioritize reducing model inference time, resource usage, and edge-device compatibility for deployment on hospital firewalls, routers, and EHR gateways.
9. **Cross-Dataset Evaluation and Generalizability:** Lastly, future research needs to test model generalizability across a variety of datasets, such as CICIDS2017, CSE-CIC-IDS2018, and actual hospital traffic. This will allow for the discovery of whether models trained on HCKDDCUP generalize to larger healthcare cybersecurity scenarios.

5 Conclusion

This study explored the application of classical machine learning algorithms Random Forest, Decision Tree, and K-Nearest Neighbors for intrusion detection in healthcare environments. Recognizing the limitations of traditional benchmark datasets, we introduced HCKDDCUP, a synthetic dataset tailored to reflect realistic healthcare-specific cyberattack scenarios. Experimental results demonstrated that domain-specific data significantly enhanced detection accuracy, with Random Forest exhibiting the most robust and consistent performance. The findings emphasize the value of healthcare-adapted datasets in improving IDS effectiveness and highlight the trade-offs among different models in terms of accuracy, interpretability, and computational efficiency. While classical models remain promising for practical deployment, especially in resource-constrained settings, further research is warranted to validate synthetic data, explore deep learning methods, and evaluate cross-dataset generalizability. By contributing a novel dataset, a comparative evaluation framework, and empirical evidence, this work supports the advancement of intelligent, context-aware, and deployable intrusion detection systems for strengthening cybersecurity in modern healthcare infrastructures.

Acknowledgement: The authors extend their appreciation to the Deanship of Scientific Research at Imam Mohammad Ibn Saud Islamic University (IMSIU) for funding this work through grant number IMSIU-DDRSP2501.

Funding Statement: This work was supported and funded by the Deanship of Scientific Research at Imam Mohammad Ibn Saud Islamic University (IMSIU) (grant number IMSIU-DDRSP2501).

Author Contributions: The authors confirm their contributions to the paper as follows: Study conception and design: Muhammad Usama; Data collection: Muhammad Usama, Arshad Aziz; Draft manuscript preparation: Muhammad Usama, Arshad Aziz, Imtiaz Hassan, Shynar Akhmetzhanova, Abdullah M. Albarrak, Tawfik Al-Hadhrami; Funding acquisition: Sultan Noman Qasem. All authors reviewed the results and approved the final version of the manuscript.

Availability of Data and Materials: The datasets used in this study are publicly available benchmark datasets. Specifically, the standard KDDCUP has been utilized, which can be accessed online at https://kdd.org/kdd-cup (accessed on 29 March 2025).

Ethics Approval: Not applicable.

Conflicts of Interest: The authors declare no conflicts of interest to report regarding the present study.

References

1. Saracevic M, Selimi A, Selimovic F. Generation of cryptographic keys with algorithm of polygon triangulation and Catalan numbers. Comput Sci. 2018;19(3):243–56. doi:10.7494/csci.2018.19.3.2749.
2. Saračevič M, Elhoseny M, Selimi A, Lončeravič Z. Possibilities of applying the triangulation method in the biometric identification process. In: Biometric identification technologies based on modern data mining methods. Cham, Switzerland: Springer International Publishing; 2021. p. 1–17. doi:10.1007/978-3-030-48378-4_1.
3. Mishra P, Varadharajan V, Tupakula U, Pilli ES. A detailed investigation and analysis of using machine learning techniques for intrusion detection. IEEE Commun Surv Tutor. 2019;21(1):686–728. doi:10.1109/COMST.2018.2847722.
4. Haddadi F, Khanchi S, Shetabi M, Derhami V. Intrusion detection and attack classification using feed-forward neural network. In: 2010 Second International Conference on Computer and Network Technology; 2010 Apr 23–25; Bangkok, Thailand. p. 262–6. doi:10.1109/ICCNT.2010.28.
5. Tavallaee M, Bagheri E, Lu W, Ghorbani AA. A detailed analysis of the KDD CUP 99 data set. In: 2009 IEEE Symposium on Computational Intelligence for Security and Defense Applications; 2009 Jul 8–10; Ottawa, ON, Canada. p. 1–6. doi:10.1109/CISDA.2009.5356528.
6. Amudha P, Karthik S, Sivakumari S. Classification techniques for intrusion detection: an overview. Int J Comput Appl. 2013;76(16):33–40. doi:10.5120/13334-0928.
7. Aggarwal P, Sharma SK. Analysis of KDD dataset attributes-class wise for intrusion detection. Procedia Comput Sci. 2015;57:842–51. doi:10.1016/j.procs.2015.07.490.
8. McHugh J. Testing intrusion detection systems: a critique of the 1998 and 1999 DARPA intrusion detection system evaluations as performed by Lincoln Laboratory. ACM Trans Inf Syst Secur. 2000;3(4):262–94. doi:10.1145/382912.382923.
9. Alkasassbeh M, Al-Naymat G, Hassanat BAA, Almseidin M. Detecting distributed denial of service attacks using data mining techniques. Int J Adv Comput Sci Appl. 2016;7:1. doi:10.14569/IJACSA.2016.070159.
10. Rasane K, Bewoor L, Meshram V. A comparative analysis of intrusion detection techniques: machine learning approach. SSRN Electron J. 2019. doi:10.2139/ssrn.3418748.
11. Gao X, Shan C, Hu C, Niu Z, Liu Z. An adaptive ensemble machine learning model for intrusion detection. IEEE Access. 2019;7:82512–21. doi:10.1109/ACCESS.2019.2923640.
12. Thakkar A, Lohiya R. A review of the advancement in intrusion detection datasets. Procedia Comput Sci. 2020;167:636–45. doi:10.1016/j.procs.2020.03.330.
13. Lei S, Xia C, Li Z, Li X, Wang T. HNN: a novel model to study the intrusion detection based on multi-feature correlation and temporal-spatial analysis. IEEE Trans Netw Sci Eng. 2021;8:3257–74. doi:10.1109/TNSE.2021.3109644.
14. Das S, Saha S, Priyoti AT, Roy EK, Sheldon FT, Haque A, et al. Network intrusion detection and comparative analysis using ensemble machine learning and feature selection. IEEE Trans Netw Serv Manag. 2022;19:4821–33. doi:10.1109/TNSM.2021.3138457.
15. Ali TE, Zoltan AD. Hierarchical deep learning for robust cybersecurity in multi-cloud healthcare infrastructures. Eng Technol Appl Sci Res. 2025;15:20358–66. doi:10.48084/etasr.6583.
16. Ali TE, Ali FI, Eyvazov F, Zoltán AD. Integrating AI models for enhanced real-time cybersecurity in healthcare: a multimodal approach to threat detection and response. Procedia Comput Sci. 2025;259:108–19. doi:10.1016/j.procs.2024.12.213.
17. Viegas E, Santin AO, Abreu Jr V. Machine learning intrusion detection in big data era: a multi-objective approach for longer model lifespans. IEEE Trans Netw Sci Eng. 2021;8:366–76. doi:10.1109/TNSE.2020.3038618.
18. Wu Z, Gao P, Cui L, Chen J. An incremental learning method based on dynamic ensemble RVM for intrusion detection. IEEE Trans Netw Serv Manag. 2022;19:671–85. doi:10.1109/TNSM.2021.3102388.
19. Hassan M, Haque ME, Tozal ME, Raghavan V, Agrawal R. Intrusion detection using payload embeddings. IEEE Access. 2022;10:4015–30. doi:10.1109/ACCESS.2021.3139835.

20. Yu L, Dong J, Chen L, Li M, Xu B, Li Z, et al. PBCNN: packet bytes-based convolutional neural network for network intrusion detection. Comput Netw. 2021;194:108117. doi:10.1016/j.comnet.2021.108117.
21. Chen J, Gao X, Deng R, He Y, Fang C, Cheng P. Generating adversarial examples against machine learning-based intrusion detector in industrial control systems. IEEE Trans Dependable Secur Comput. 2022;19:1810–25. doi:10.1109/TDSC.2020.3037500.
22. Rao KN, Venkata Rao K, Prasad Reddy PVGD. A hybrid intrusion detection system based on sparse autoencoder and deep neural network. Comput Commun. 2021;180:77–88. doi:10.1016/j.comcom.2021.08.026.
23. Neupane S, Ables J, Anderson W, Mittal S, Rahimi S, Banicescu I, et al. Explainable intrusion detection systems (X-IDS): a survey of current methods, challenges, and opportunities. IEEE Access. 2022;10:112392–415. doi:10.1109/ACCESS.2022.3216617.
24. Paleyes A, Urma R-G, Lawrence ND. Challenges in deploying machine learning: a survey of case studies. ACM Comput Surv. 2023;55:1–29. doi:10.1145/3533378.
25. Amine AM, Khamlichi YI. Optimization of intrusion detection with deep learning: a study based on the KDD Cup 99 database. Int J Saf Secur Eng. 2024;14:1029–38. doi:10.18280/ijsse.140402.
26. Gupta L, Salman T, Ghubaish A, Unal D, Al-Ali AK, Jain R. Cybersecurity of multi-cloud healthcare systems: a hierarchical deep learning approach. Appl Soft Comput. 2022;118:108439. doi:10.1016/j.asoc.2022.108439.
27. Weber SB, Stein S, Pilgermann M, Schrader T. Attack detection for medical cyber-physical systems-A systematic literature review. IEEE Access. 2023;11:41796–815. doi:10.1109/ACCESS.2023.3270225.
28. Seghir F, Drif A, Selmani S, Cherifi H. Wrapper-based feature selection for medical diagnosis: the BTLBO-KNN algorithm. IEEE Access. 2023;11:61368–89. doi:10.1109/ACCESS.2023.3287484.
29. Qureshi KA, Shams Malick RA, Sabih M, Cherifi H. Deception detection on social media: a source-based perspective. Knowl-Based Syst. 2022;256(6380):109649. doi:10.1016/j.knosys.2022.109649.