Open Access
Full Length Article

ORIGINAL ARTICLE

An Ensemble Machine Learning Based Cybersecurity Intrusion Detection System (C-IDS)

Amna Ilyas[1], Muhammad Faisal Siddiqui[2], Farheen Faisal Siddiqui[3], Abdullah Ayub Khan[4], Muhammad Zohaib Khan[5]

amna.amu.ilyas1@gmail.com, faisalsiddiqui.bukc@behria.edu.pk, farheen.faisal@yahoo.com
abdullahayub.bukc@bahria.edu.pk, zohaib_khan2017@yahoo.com

[1]*Department of computer science and Cybersecurity, Ravensbourne University London, England*
[2]*Department of Computer Science, University of Karachi Campus, Karachi, Pakistan*
[3]*Department of Computer Science, University of Karachi Campus, Karachi, Pakistan*
[4]*Department of Computer Science, Behria University, Karachi, Pakistan*
5 Department of Information Technology, Shaheed Mohtarma Benazir Bhutto Institute of Trauma Karachi, Pakistan

ABSTRACT

An essential component of network security is intrusion detection, which shields computer systems against attacks and illegal access. Traditional intrusion detection systems (IDS) rely on signature-based detection, which limits their ability to detect unknown complex threats. Machine learning-based methods have demonstrated encouraging outcomes in detecting unidentified harmful intrusions. An approach to developing an IDS model based on ensemble learning is presented in this research. Compared to its individual classifiers, the model presented in this research performs comparatively better overall. Lightweight ML approaches, such as QDA, NB, SGD, and DT, are used to build this ensemble model. The dataset is utilized to train and assess the performance of this suggested model as well as the individual classifiers used to construct the ensemble model. The binary class is used to evaluate the performance. In order to choose only the most pertinent features, the suggested approach additionally makes use of a feature selection technique. The empirical findings unequivocally demonstrate that employing an ensemble classifier can be very beneficial in the field of IDS with unbalanced datasets, where misclassifications might be expensive. The suggested strategy using the Decision Tree (DT) technique beats current approaches in terms of accuracy, usually reaching 95% with improved assessment metrics according to our research using various ensemble methods. This tactic could be helpful in bolstering computer systems' and networks' defenses against new cyberthreats

Keywords: Cybersecurity, Intrusion Detection Systems (IDS), Anomaly-Based, Ensemble Machine Learning (EML) & Logistic Regression (LR), Stochastic Gradient Descent (SGD), Stacking Ensemble Classifier (SEC), Quadratic Discriminant Analysis (QDA), Naive Bayes (NB), Agglomerative Hierarchical Clustering (AHC), and Decision Tree (DT)

Author's Contribution
Data analysis, interpretation, and manuscript writing, Active participation in ^ Conception, synthesis, planning of research. Interpretation and discussion

Address of Correspondence
Amna Ilyas
Email: amna.amu.ilyas1@gmail.com

Article info.
Received: June 11, 2026
Accepted. July 10, 2026
Published: August 28, 2026

Cite this article: Amna Ilyas, Muhammad Faisal Siddiqui, Farheen Faisal Siddiqui, Abdullah Ayub Khan, Muhammad Zohaib Khan
An Adaptive Sensor Data Access Framework for Mobile and Web Environments. A Review. J. inf. commun. technol. robot. app/.2026; *17(1):01-12*

Funding Source Nil
Conflict of Interest: Nil

INTRODUCTION

Network intrusion refers to any unauthorized or malicious activity directed at computer networks or information systems with the aim of accessing resources, disrupting operations, or compromising sensitive information [1,2]. Such intrusions may originate from external attackers or internal users and can occur in various forms, including denial-of-service attacks, distributed denial-of-service attacks, malware infections, brute-force attacks, port scanning, phishing, botnet activities, and social engineering attacks [3, 4]. These threats can lead to severe consequences for organizations, such as financial losses, operational disruption, legal liabilities, reputational damage, and the exposure of confidential data [5]. With the rapid growth of Internet usage and increasing reliance on digital infrastructures, cyber-attacks have become more frequent, sophisticated, and difficult to detect [6].

To overcome these limitations, ML–based IDs approaches have attracted considerable attention [7]. ML models analyze network traffic data to learn patterns associated with normal and malicious behavior, enabling anomaly-based detection and the identification of previously unseen attacks [8]. However, IDS solutions based on a single ML classifier often suffer from issues such as bias, over-fitting, sensitivity to noise, and limited generalization capability, particularly when dealing with imbalanced datasets commonly found in IDs scenarios [9, 10, 11]. Ensemble ML provides an effective approach to addressing these challenges by combining multiple learning models into a unified decision-making framework [12]. An ensemble learner aggregates the outputs of several classifiers to produce a final prediction, improving robustness and overall detection performance [13]. Ensemble

where high accuracy and low false alarm rates are critical [14]. By exploiting the complementary strengths of different classifiers, ensemble-based IDS can enhance detection capability and reduce misclassification costs [15].

In this research, an ensemble-based Cybersecurity & IDS (C-IDS) is proposed using lightweight and computationally efficient ML models, including QDA, NB, SGD, and DT classifiers. These classifiers are integrated using ensemble learning strategies to improve detection accuracy and reliability. To further enhance system efficiency, a feature selection technique is applied to retain only the most relevant features, thereby reducing computational improving classification performance. The proposed system is evaluated using a binary classification framework to distinguish between normal and malicious network traffic.

The main objectives of this research are as follows:

- To develop an ensemble ML–based Cybersecurity intrusion detection system capable of accurately identifying malicious network activities.
- To evaluate and compare the performance of lightweight ML classifiers, including QDA, NB, SGD, and DT, both individually and within an ensemble framework.
- To improve IDs accuracy and reliability through the integration of ensemble learning techniques and feature selection methods.
- To design a robust and efficient IDS framework suitable for handling imbalanced datasets and evolving cyber threats in network environments.

The paper is organized as follows. Section 2 shows a review of related work in IDS and ensemble ML. Section 3 describes the methodology and system architecture. Section 4 discusses the performance evaluation results. Finally, Section 5 shows the conclusion and future work. methods are especially well suited for IDs systems

LITERATURE REVIEW

This section highlights the benefits and drawbacks of previous research on ML and intrusion detection systems [16]. This provides background information on the study paper's suggested ensemble-based ML method [17]. Over the years, researchers have developed a variety of ML-based model techniques

[18]. We examined fairly recent models with their advantages and disadvantages in order to better understand the importance of our suggested ensemble-based method to detect intrusion and guarantee network security [19]. IDS are a fundamental component of Cybersecurity, designed to detect and prevent unauthorized or malicious activities within computer networks [20]. The significance of ensemble learning in enhancing IDS has been thoroughly studied in earlier research. A noteworthy survey [23] thoroughly examined advancements in ensemble-based IDS. Using datasets like KDD'99 and AWID, the study looked at traditional ensemble techniques like bagging, boosting, stacking, and majority voting. A wide range of learning techniques, including neural networks (NNs), SVMs, DT, fuzzy clustering, and RBF, were also presented. This thorough analysis highlighted the advantages and disadvantages of the ensemble approaches already in use, offering potential avenues for future development. An IDS architecture that includes multiple classifiers, such as NB, LR, and DT, was proposed in [21] for two-class IDS. Stochastic gradient descent was used to integrate these models using the CICIDS-2017 datasets. The input space was refined using chi-square-based feature selection [28]. The authors recommend investigating data augmentation and alternative ensemble approaches for performance improvement because the model suffered from imbalanced data despite its efficacy [22]. Other approaches include protocol-based IDS and statistical IDS that rely on protocol analysis and statistical modeling to detect abnormal behavior. ML techniques have been widely applied to enhance IDS performance by enabling automatic learning from network traffic data. Algorithms such as NB, SVM, KNN, and DT have been investigated for IDS [24]. Another contribution was a pipeline for binary anomaly detection created by [26] using ensemble learners like RF, AdaBoost, and GB that were aggregated by soft voting. Similarly, classifiers like LR, DT, and SVM were used in an ensemble configuration in [27] to improve detection accuracy on the NSL-KDD and UNSW-NB15 datasets. Additionally, this work experimented with feature selection, highlighting the potential of unsupervised learning and the need for more realistic datasets. Another study [29] used weighted voting across SVM, auto-encoder, and random forest to integrate real-world data, such Palo Alto logs, into an IDS ensemble framework. Its scalability and voting method diversity limits were observed, notwithstanding its effectiveness in lowering false positives. One study [28] presented an ensemble IDS based on the TON-IoT dataset in the context of IoT security. Base models including DT, and LR were stacked and voted over as part of the process. Despite the framework's performance, it was not evaluated on other datasets and did not include other ensemble techniques like bagging [29]. Another study [30], on the other hand, concentrated on real-time detection using a group of auto-encoders, providing a special method appropriate for streaming data. They investigated the use of NB, and LR in ensemble model selection to reduce over-fitting in tiny binary datasets. However, its computationally demanding validation processes and dependence on non-IDS datasets were serious disadvantages.

Table 1. Comparison between our work and prior ensemble learning approaches for IDS systems.

Study	Dataset(s)	Ensemble Method(s)	Base Classifiers	Limitations
Al-a'araji et al. [21]	KDD'99	stacking, voting	NN, DT, SVM, RBF	Broad survey
Thockom et al. [22]	CICIDS-2017, KDD'99	SGD-based ensemble	GNB, LR, DT	Imbalanced data
Alotaibi et al. [23]	TON-IoT	Stacking, voting	RF, KNN, DT, LR	No other datasets
Mirsky et al. [24]	Custom streaming data	Auto-encoder ensemble	Autoencoders	Streaming-focused
Lazzarini et al. [25]	CICIDS-2017, ToN-IoT	Stacking	DNN, CNN, RNN, LSTM	High computational cost
Mahfouz et al. [26]	GTCS	Majority voting	J48, MLP, IBK	Limited model diversity

To overcome these limitations, ensemble learning techniques have been introduced for IDS. Ensemble methods, such as RF and stacking-based models, combine multiple classifiers to improve accuracy, robustness, and reliability. By leveraging the complementary strengths of different ML models, ensemble-based IDS have shown improved detection performance compared to single-classifier approaches. Despite these advancements, challenges related to computational efficiency and adaptability remain, highlighting the need for lightweight and effective ensemble-based IDS frameworks, which this research aims to address. The second section is the detailed methodology, as a result of which this proposed method is supported.

PROPOSED FRAMEWORK

The proposed methodology for the Cybersecurity intrusion detection system (C-IDS) involves a structured approach to data collection, preprocessing, feature refinement, model training, and evaluation. The main objective is to develop an accurate and reliable ensemble-based IDS capable of detecting malicious network activities. The steps are as follows:

1. The study uses a Cybersecurity dataset obtained from Google Drive.
2. Several preprocessing steps were applied to ensure the quality and consistency of the dataset. Missing or inconsistent values were either imputed or removed. Feature scaling using MinMaxScaler and Standard scalar was performed to bring all variables into a common range, followed by normalization to improve numerical stability and comparability across features.
3. Agglomerative Hierarchical Clustering (AHC), an unsupervised learning technique, was applied to identify natural clusters within the data. This step enhanced classification metrics by grouping similar instances and reducing noise in the dataset.
4. The processed dataset was divided into training 75% and testing sets 25%.
5. The SMOTE was applied exclusively to the training set to address class imbalance. By generating synthetic samples for the minority class, SMOTE ensures fair representation and reduces model bias toward the majority class.
6. Multiple ML techniques were employed to develop and evaluate the IDS, including Agglomerative Hierarchical Clustering (AHC), Stacking Ensemble Classifier (SEC), Logistic Regression (LR), Quadratic Discriminant Analysis (QDA), Naive Bayes (NB), Stochastic Gradient Descent (SGD), and Decision Tree (DT). This diverse selection of algorithms was chosen to improve prediction accuracy, enhance reliability, and capture complex patterns within the dataset.

The overall methodological framework and task breakdown of the proposed C-IDS system are illustrated in Figure 1, providing a visual representation of the steps from data preprocessing to model evaluation.

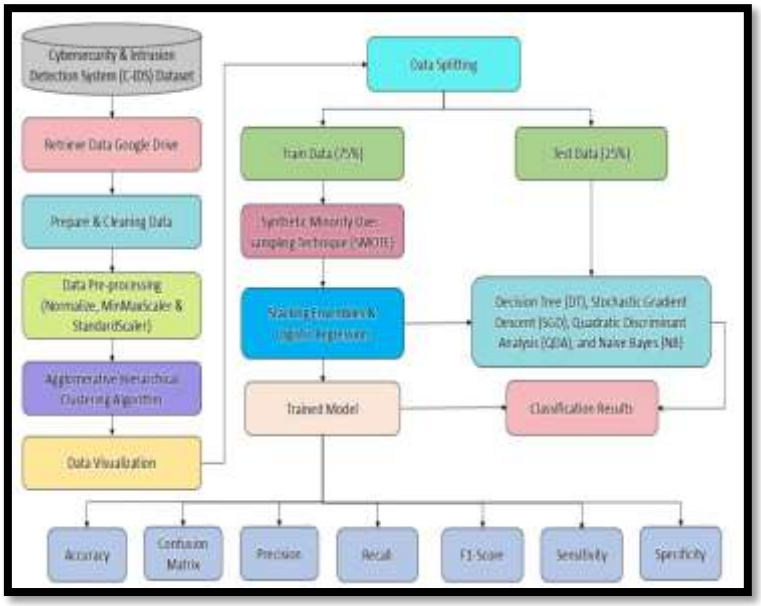


Figure 1: Proposed Methodology of the Cybersecurity & IDS (C-IDS)

The final processed dataset was split into 75% for training and 25% for testing to ensure comprehensive validation of the proposed ensemble-based IDS model. Initially, the raw data was cleaned and processed following strict preprocessing standards to guarantee data consistency, integrity, and quality, as illustrated in the accompanying workflow diagram. The two main stages of this process, data preparation and classification, are critical in determining the overall performance of the model. This systematic approach enhances the robustness, generalizability, and detection accuracy of the C-IDS framework across different network environments.

PREPROCESSING

Effective ML-based IDs requires high-quality input data, making preprocessing a vital step in the methodology. The preprocessing stage ensures that the dataset is clean, consistent, and suitable for training ensemble learning models, directly impacting the accuracy and reliability of the final IDS. It involves detecting and correcting inconsistencies, normalizing feature values, encoding categorical variables, and preparing the dataset for classification. In this study, preprocessing steps include removing duplicate records, handling missing or infinite values, separating numerical and categorical features, normalizing numerical columns, encoding categorical variables, and converting the target variable into a discrete binary label (benign or malicious). Duplicate rows were identified using the duplicated function in the panda's library and removed with drop duplicates. Numerical features were normalized using Standard Scalar from the sk-learn library to ensure zero mean and unit variance, improving model stability. Categorical features were encoded using Label Encoder, converting each unique value into an integer to make them compatible with ML models. These preprocessing steps prepare the dataset for the subsequent clustering and ensemble classification stages, enabling the C-IDS to learn underlying patterns effectively and perform accurate IDS.

Data Collection: The dataset used in this research was obtained from the publicly available Cybersecurity Intrusion Detection Dataset on Kaggle, which contains labeled network traffic records suitable for IDS analysis. This dataset comprises 9,537 total instances with 11 features, representing a mix of network behavior indicators and class labels that distinguish between benign and malicious activities. The dataset thus provides a balanced foundation for evaluating ML models for Cybersecurity IDS. The binary target variable Attack Detected shows that out of the total instances, 5,273 records are normal (benign) and 4,264 are malicious, indicating a significant presence of both classes for effective supervised learning is shown in Figure 2.

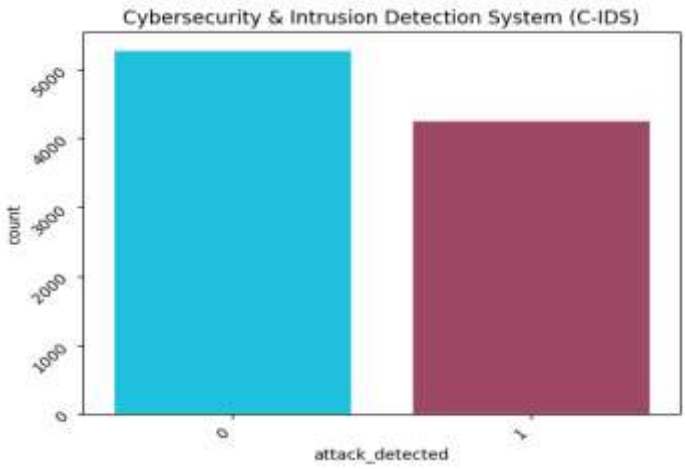


Figure 2: Number Of Cybersecurity & Intrusion

The dataset also includes categorical attributes such as Protocol Type and Browser Type, which characterize network traffic protocols and client software. For example, the most frequent protocol in the dataset is TCP with 6,624 instances, followed by UDP (2,406 instances) and ICMP (507 instances) is shown in Figure 3.

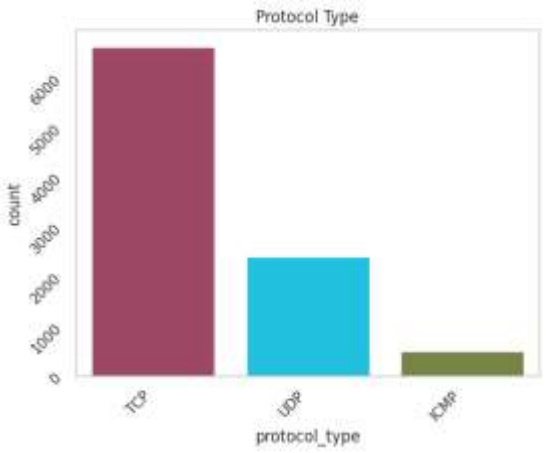


Figure 3: Number Of Protocol Type

Browser Type distribution shows Chrome as the most common (5,137), followed by Firefox (1,944), Edge (1,469), Safari (485), and Unknown (502) categories is shown in Figure 4.

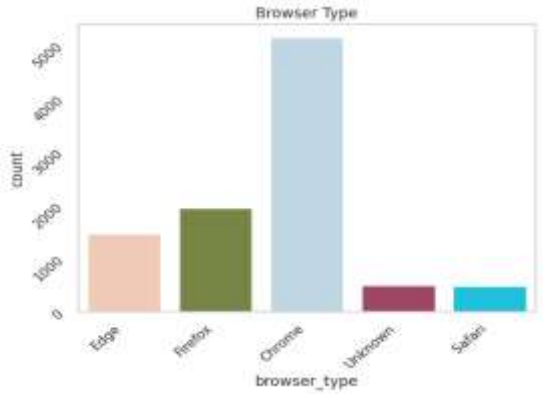


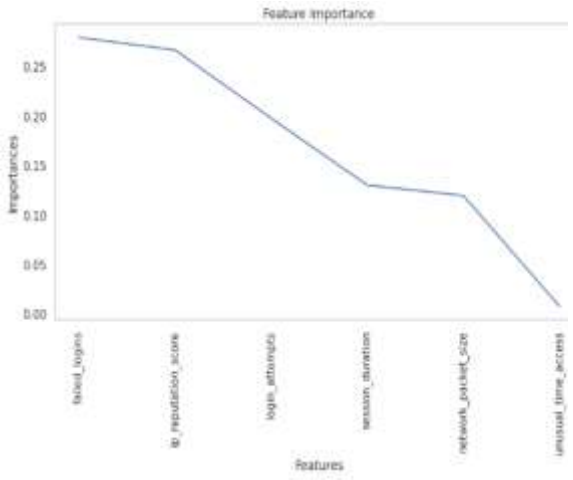
Figure 4: Number of Browser Type

FEATURE IMPORTANCE. Feature importance evaluates the contribution of each input variable to the predictions of the model. Features with higher importance scores play a more significant role in classification, while less important features can be removed to reduce dimensionality, improve computational efficiency, and enhance predictive accuracy. In this study, Recursive Feature Elimination (RFE) was employed as the primary method for feature selection. RFE is an iterative algorithm that ranks features based on their importance and sequentially eliminates the least significant ones to identify the optimal subset of features. To further improve performance, a hybrid version of RFE was implemented by integrating multiple ML models, including RRF, SVM, and XGBoost. This ensemble-based approach allows for more robust feature evaluation by combining the strengths of diverse algorithms, ensuring that the most informative and predictive features are retained. The resulting feature set enhances model robustness, increases classification accuracy, and ensures optimal performance of the proposed C-IDS framework.

The selected feature importance scores for the model are summarized in Table 2. In addition, Figure 5 provides a visual representation of the feature contributions, offering a dual-perspective analysis that combines quantitative evaluation with graphical clarity. This approach not only guides model optimization but also informs future improvements in IDs research and ensemble-based cybersecurity frameworks.

Table 2: Importance of Dataset Features

Sr. no.	Feature	Importance of Score Feature
1	failed_logins	0.279102
2	ip_reputation_score	0.266489
3	login_attempts	0.197595
4	session_duration	0.129948
5	network_packet_size	0.119412
6	unusual_time_access	0.007453



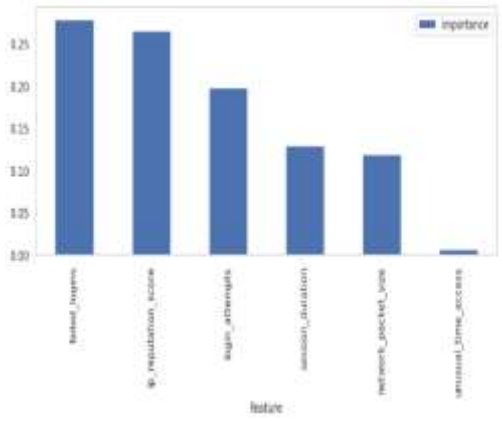


Figure 5: Feature Importance Dataset

This section provides a systematic analysis of the dataset and its preparation for the proposed C-IDS framework, emphasizing key stages such as data normalization, preprocessing, clustering, and feature organization. It also highlights the relationships between variables, evaluation measures, and overall data readiness for model development. The process begins with the collection of raw network traffic and system activity data, which is then cleaned, standardized, and normalized to ensure consistency and suitability for ML models. Table 3 summarizes the processed dataset, providing a clear reference for subsequent analysis and model training. To better understand complex patterns and relationships within the data, Agglomerative Hierarchical Clustering (AHC) was applied. Figure 6 illustrates these interrelationships, where each point represents a network instance, and clusters highlight similar feature patterns. This visualization facilitates the identification of correlated features, detection of anomalies, and understanding of critical interactions within the dataset, thereby supporting improved feature selection and enhanced performance of the ensemble-based IDS model. The black circle is the data point X of this representation, and the bright pink circle is the data point Y. This enables the separation of feature patterns with ease and enables one to understand sensitive data relations.

Table 3: shows the dataset for Cybersecurity & IDS (C-IDS)

```
07-Dimension
[[ 0.52060362 -0.1034006 -0.09427052 ... -0.3466666 -0.3160479
   0.64494773]
 [-1.15932966 -0.71080433 0.98139904 ... -0.85980569 -0.3160479
  -0.62260865]
 [ 1.2899728 -0.18043982 -1.82356574 ...  0.39704085 -0.3160479
   0.92972275]
 ...
 [ 1.30959052 0.32986603 -2.04290256 ... -0.26113137 -0.3160479
  -0.62260865]
 [ 1.23762716 0.71765783 -1.52158502 ...  0.0990545   3.26493171
  -0.62260865]
 [ 1.20537421 2.08516904 -1.39191745 ...  3.67680634 3.919584
   2.17897656]]
```

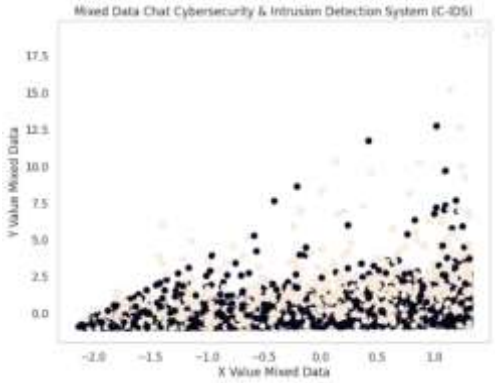


Figure 6: Mixed Data Chat Cybersecurity & IDS (C-IDS)

AGGLOMERATIVE HIERARCHICAL CLUSTERING METHOD: Agglomerative Hierarchical Clustering (AHC) is a widely used unsupervised learning technique that groups similar data points without requiring predefined labels. Also known as AGNES (Agglomerative Nesting), AHC follows a bottom-up approach, where each data point initially forms its own cluster. The algorithm then recursively merges the two most similar clusters at each step until all points are combined into a single cluster. This hierarchical process is commonly visualized using a dendrogram, which illustrates the order of cluster merges and the structure of the data. Each resulting cluster is assigned a unique identifier, allowing researchers to control the level of granularity and extract meaningful patterns from complex datasets [28, 29, and 30].

The AHC process involves the following steps:

- Compute the Proximity Matrix: Calculate pairwise similarities between data points using a distance metric, such as Euclidean or Manhattan distance.
- Build the Dendrogram: Apply a linkage criterion to the proximity matrix to construct the hierarchical tree.
- Merge Closest Clusters: Iteratively combine the two clusters or points with the highest similarity.
- Iterative Merging: Continue merging until all points are included in a single root cluster.
- Completion: Assign final cluster identifiers and prepare the data for further analysis.

In this study, AHC is applied to a dataset with 35 features and binary class labels, making it suitable for detecting and clustering Cybersecurity threats. The method transforms raw and unstructured data into organized clusters, enhancing interpretability and highlighting key patterns. Each data point is assigned a probability score indicating its likelihood of belonging to a particular cluster, resulting in a relationship matrix that captures the strength of associations between samples and cluster centroids.

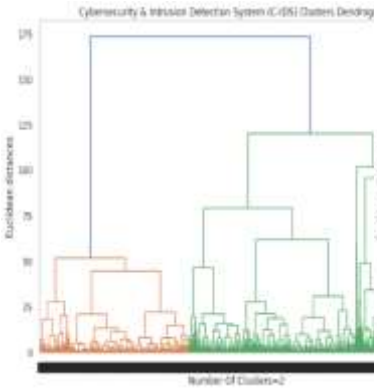


Figure 7: Cybersecurity & IDS (C-IDS) Clusters Dendrograms

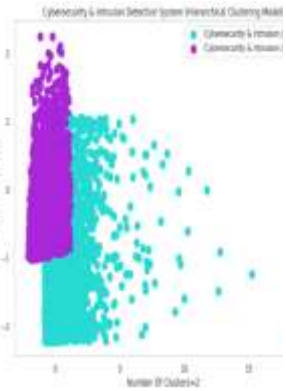


Figure 8: Cybersecurity & IDS (Hierarchical Clustering Model)

The graphical output of AHC, including dendrograms (Figures 7 and 8), provides a clear visualization of cluster structures, facilitating the identification of anomalous or suspicious network behaviors. Cluster-based feature identification is evaluated using metrics such as F-measure, accuracy, and recall, which assess the system's ability to correctly detect and classify relevant data points. Additionally, a "worry score" is computed to classify data points as normal, uncertain, or anomalous based on deviations from expected patterns. This approach strengthens the model's capacity to detect intrusions, organize network traffic effectively, and improve decision-making within the proposed ensemble-based C-IDS framework.

3.2 CLASSIFICATION

Classification is a core process in ML that assigns data points to predefined classes based on patterns learned from labeled training data. In the context of Cybersecurity IDS, classification algorithms are used to distinguish between normal and malicious network activities. By leveraging supervised learning techniques, classifiers construct decision boundaries from historical data to predict the class labels of new, unseen instances. Rigorous implementation and evaluation of multiple classification algorithms allow researchers to determine the most accurate and reliable models for a given dataset, ensuring both high detection rates and robust generalization to varied network conditions. In this study, classification is applied to the processed C-IDS dataset to identify the most effective algorithms for IDS.

SYNTHETIC MINORITY OVER-SAMPLING TECHNIQUE (SMOTE). Imbalanced datasets, where one class (e.g., malicious traffic) is significantly underrepresented compared to another (e.g., normal traffic), can negatively impact classifier performance. This enhances the ability of classifiers to learn meaningful patterns from underrepresented classes, thereby improving predictive accuracy and reliability on imbalanced cybersecurity datasets [31].

ENSEMBLE MACHINE LEARNING (EML). Ensemble learning combines multiple base models to produce a more accurate and stable prediction than individual classifiers. By integrating strategies such as bagging,

boosting, and stacking, ensemble methods reduce both bias and variance, enhancing the generalization of models. In cybersecurity IDS applications, where network traffic can be noisy or highly variable, ensemble learning provides more robust predictions and reduces the likelihood of false positives and false negatives, ensuring reliable threat detection in dynamic environments [32].

STACKING ENSEMBLE CLASSIFIER (SEC). The SEC is an advanced ensemble approach that leverages the complementary strengths of multiple base classifiers. In this study, algorithms such as K-NN, SGD, NB, and QDA are independently trained on the same dataset. Their predictions are then combined by a meta-learner, typically LR, to generate the final classification output. This multi-layered structure improves prediction accuracy, reduces overfitting, and enhances robustness by exploiting the unique capabilities of each base classifier. The SEC framework is particularly effective for detecting complex intrusion patterns and can adapt to high-dimensional, high-stakes cybersecurity data [33].

LOGISTIC REGRESSION (LR). LR is a widely used statistical method for binary classification tasks, modeling the probability of an event occurring based on predictor variables. In the context of IDS, LR estimates the likelihood of a network instance being normal or malicious. By applying the logit function, LR transforms probabilities into a linear combination of input features, providing interpretable results and computational efficiency. Its simplicity, transparency, and reliability make it a standard baseline in cybersecurity and other domains requiring clear decision-making [34].

QUADRATIC DISCRIMINANT ANALYSIS (QDA) ALGORITHM. QDA is a powerful supervised classification technique that assumes each class in a dataset follows its own Gaussian distribution. Unlike Linear Discriminant Analysis (LDA), which assumes a shared covariance matrix across classes, QDA models class-specific covariance matrices, enabling flexible, curved decision boundaries. This flexibility allows QDA to capture complex patterns in the data, making it particularly effective for datasets with imbalanced classes or smaller training samples. However, since QDA requires an independent covariance estimate for each class, it can become computationally intensive with high-dimensional data and is susceptible to overfitting when the number of features exceeds the number of training samples [35].

The QDA algorithm follows a systematic procedure to classify new instances:

1. Compute Class-Specific Statistics: Calculate the mean vector and covariance matrix for each class to capture its unique distribution.
2. Apply the Quadratic Discriminant Function: Evaluate new data points using a quadratic decision rule based on each class's covariance structure.
3. Assign Class Labels: Assign each data point to the class with the highest discriminant score for optimal classification.
4. Use Non-Linear Decision Boundaries: Leverage the curved decision surfaces of QDA to separate classes with overlapping or complex patterns effectively.

For the proposed C-IDS framework, QDA provides robust probabilistic classification, especially useful for complex network traffic with overlapping features. Its class-specific covariance modeling allows accurate detection of intrusions and reduces misclassification errors. Figure 9 illustrates the confusion matrix generated by the QDA model, offering a visual representation of its performance in classifying network activity as normal or malicious.

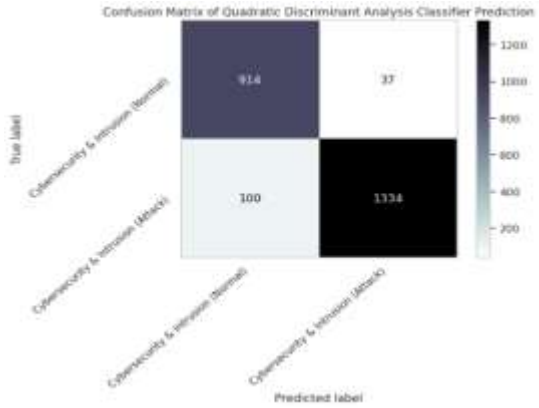


Figure 9: Confusion Matrix Quadratic Discriminant Analysis (QDA) Algorithm

Naive Bayes (NB) ALGORITHM: NB is a widely used supervised learning algorithm for classification tasks, grounded in the principles of Bayes' theorem. Its simplicity, computational efficiency, and probabilistic foundation make it highly suitable for applications involving large-scale or high-dimensional datasets, such as network traffic analysis in cybersecurity. Despite its assumption of feature independence, NB can generate accurate predictions quickly and with minimal computational overhead, making it an effective component in real-time IDS [35]:

The Naive Bayes classifier follows a structured sequence of steps to determine the class of a new instance:

- Compute Conditional Probabilities: Determine the probability of each feature value occurring within each class.
- Assign Class Label: Assign the instance to the class with the highest posterior probability.
- Terminate Classification: Once the most probable class is determined, the process concludes for that instance.

In the context of the proposed C-IDS framework, NB is used to classify network activity as normal or malicious. Its lightweight nature ensures efficient handling of large volumes of network traffic while maintaining high predictive accuracy. The probabilistic approach of NB also complements ensemble methods, contributing to the diversity and robustness of the combined classifier. The performance of the Naive Bayes classifier is evaluated using a confusion matrix, which provides a detailed assessment of its predictive accuracy by comparing predicted class labels against actual labels. Figure 10 illustrates the confusion matrix for NB, demonstrating its capability to classify network events accurately and efficiently within the C-IDS model.

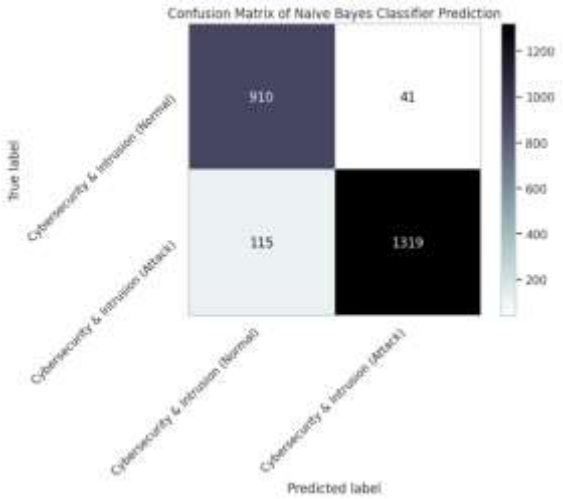


Figure 10: Confusion Matrix Naive Bayes (NB) Algorithm

STOCHASTIC GRADIENT DESCENT (SGD) ALGORITHM. SGD is a highly efficient optimization and classification method, particularly well-suited for large-scale datasets like network traffic data in cybersecurity applications. Unlike traditional batch gradient methods that update model parameters after evaluating the entire dataset, SGD performs incremental updates for each data point. This approach not only reduces computational cost but also accelerates convergence. Additionally, the inherent randomness in SGD helps the model avoid local minima and better optimize complex, non-convex functions. SGD is commonly employed in models such as Support Vector Machines (SVM) and Logistic Regression due to its speed, scalability, and robustness.

The SGD algorithm follows a structured procedure for model training and classification:

Compute Gradients: Calculate the gradient of the loss function for each feature and data point, incorporating necessary adjustments for model parameters.

Initialize Parameters: Randomly assign initial values to model parameters. Split the dataset into independent variables (features) and the dependent variable (target) to define predictions for each instance.

Update Gradients: Iteratively revise gradient

computations using the current parameter values to ensure accurate, incremental updates.

Determine Step Size: Multiply the gradient by the learning rate to compute the step size for updating parameters.

Update Parameters: Apply the formula new parameter = old parameter + (step length × gradient) to adjust each parameter iteratively.

Convergence: Repeat the gradient updates until the parameter changes become minimal, indicating that the model has reached a stable optimum.

Finalize Model: Once the parameters converge and further updates no longer improve the loss function, training concludes, resulting in an optimized classifier.

In the context of the proposed C-IDS framework, SGD is used to classify network activities as normal or malicious by continuously adjusting the model to capture patterns in network traffic. This incremental and adaptive approach allows SGD to handle high-dimensional and noisy datasets effectively, ensuring improved prediction accuracy and generalization. The performance of the SGD model is evaluated using a confusion matrix, which compares predicted class labels against actual labels to assess accuracy, precision, recall, and overall effectiveness. Figure 11 illustrates the confusion matrix for the SGD classifier, demonstrating its capability to classify network traffic efficiently and contribute to the ensemble learning approach in C-IDS.

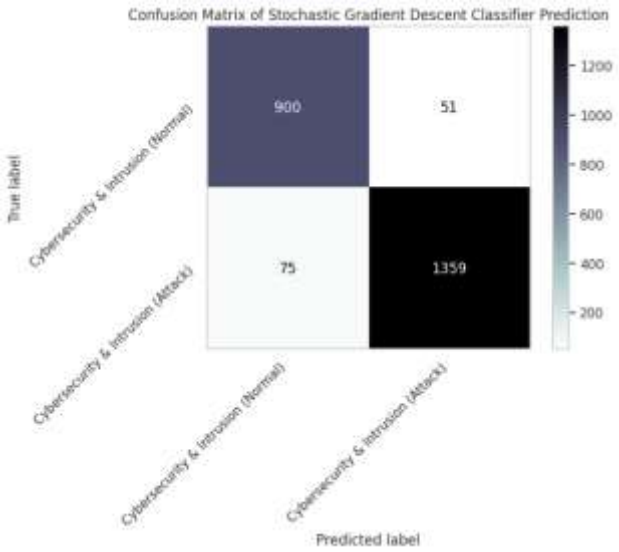


Figure 11: Confusion Matrix Multi-Layer Perceptron Algorithm

Decision Tree (DT) Algorithm: Decision Tree (DT) is a widely used supervised learning algorithm for classification tasks, offering a clear and interpretable model structure. It works by recursively partitioning the dataset into subsets based on feature values, forming a tree-like structure where each internal node represents a decision on a feature, each branch represents an outcome of that decision, and each leaf node represents a class label. This hierarchical decision-making process makes DT particularly suitable for identifying complex patterns in network traffic and detecting anomalies in Cybersecurity applications. In the proposed C-IDS framework, DT is used to classify network activities as normal or malicious. Its ability to handle both categorical and numerical features, as well as its interpretability, makes it a powerful component in ensemble learning. By combining multiple classifiers, including DT, the ensemble approach leverages the strengths of each model while minimizing individual weaknesses, resulting in enhanced detection accuracy and reliability. The effectiveness of the Decision Tree classifier is evaluated using a confusion matrix, which measures the model's predictive performance by comparing predicted labels with actual labels. Figure 12 presents the confusion matrix for the DT model, highlighting its capability to accurately classify network traffic and contribute to the overall robustness of the ensemble-based C-IDS.

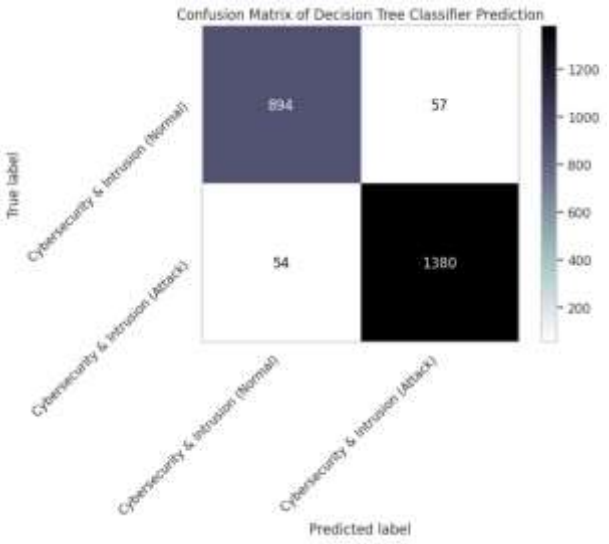


Figure 12: Confusion Matrix Decision Tree (DT)

4. RESULTS AND DISCUSSION

To evaluate the effectiveness of the proposed ensemble-based (C-IDS), a comprehensive analysis was conducted using multiple hybrid model combinations. Each approach combined AHC with a SEC and a variety of base classifiers to improve the detection and classification of normal versus malicious network activity. The goal was to maximize accuracy, reduce false positives, and enhance the system's overall reliability in detecting cyber threats.

Table 4: Comparative Accuracy of Hybrid Models of (C-IDS)

Hybrid Methods	Accuracy of Methods
Agglomerative Hierarchical, Stacking Ensemble & Logistic Regression, Decision Tree (DT) **Proposed Method**	95.3459 %
Agglomerative Hierarchical, Stacking Ensemble & Logistic Regression, Stochastic Gradient Descent (SGD) **Proposed Method**	94.7169 %
Agglomerative Hierarchical, Stacking Ensemble & Logistic Regression, Naive Bayes (NB) **Proposed Method**	93.4591 %
Agglomerative Hierarchical, Stacking Ensemble & Logistic Regression, Quadratic Discriminant Analysis (QDA) **Proposed Method**	94.2557 %

The performance of each hybrid model was assessed based on accuracy, which provides a clear measure of the system's ability to correctly classify network traffic. Table 4 summarizes the accuracy results for the different hybrid model configurations. Among the evaluated models, the combination of AHC, SEC, LR, and DT achieved the highest accuracy of 95.35%, indicating that this model was the most effective in capturing complex patterns in the network traffic and accurately distinguishing between normal and malicious activity. Other combinations also demonstrated strong performance, with the AHC + SEC + LR + SGD model achieving 94.72% accuracy, followed by the AHC + SEC + LR + QDA model at 94.26%, and the AHC + SEC + LR + NB model at 93.46%. These results confirm that ensemble-based approaches outperform single classifiers by leveraging the complementary strengths of multiple algorithms, enhancing both robustness and predictive performance in IDS tasks.

Table 5: Performance Scores of Hybrid Models of (C-IDS)

Table 5 presents a detailed performance analysis of the proposed ensemble-based C-IDS using key evaluation metrics, including precision, recall, F1-score, sensitivity, and specificity. These metrics provide a comprehensive view of the system's effectiveness in detecting malicious network activity while minimizing false alarms. Among the hybrid models, the combination of AHC, SEC, LR, and DT achieved the highest precision of 0.9517,

S/No.	Parameter Score	Agglomerative Hierarchical, Stacking Ensemble & Logistic Regression, Decision Tree (DT) Algorithm	Agglomerative Hierarchical, Stacking Ensemble & Logistic Regression, Stochastic Gradient Descent (SGD) Algorithm	Agglomerative Hierarchical, Stacking Ensemble & Logistic Regression, Naive Bayes (NB) Algorithm	Agglomerative Hierarchical, Stacking Ensemble & Logistic Regression, Quadratic Discriminant Analysis (QDA) Algorithm
1	Precision	0.95168600	0.94345333	0.92882890	0.93719653
2	Recall	0.95120309	0.94703549	0.93834611	0.94567928
3	F1-Score	0.95144284	0.94513782	0.93260935	0.94071927
4	Sensitivity	0.94006309	0.94637223	0.95688748	0.96109358
5	Specificity	0.96234309	0.94769874	0.91980474	0.93026499

indicating that the model accurately identifies malicious instances with minimal misclassification. The same model also demonstrated strong recall (0.9512) and F1-score (0.9514), reflecting balanced performance in correctly detecting both malicious and normal network traffic. Its sensitivity (0.9401) and specificity (0.9623) further highlight the model's ability to correctly classify positive and negative instances, making it the most robust and reliable approach among the evaluated configurations. The AHC + SEC + LR + Stochastic Gradient Descent (SGD) model also performed well, with precision, recall, and F1-score values of 0.9435, 0.9470, and 0.9451, respectively. Its sensitivity (0.9464) and specificity (0.9477) demonstrate a good balance between detecting true intrusions and avoiding false positives, making it suitable for environments where rapid detection is critical. The AHC + SEC + LR + NB model achieved precision (0.9288) and recall (0.9383) slightly lower than DT and SGD but exhibited the highest sensitivity (0.9569), indicating strong capability in detecting positive cases, though with slightly lower specificity (0.9198) compared to other models. Finally, the AHC + SEC + LR + QDA model showed a precision of 0.9372, recall of 0.9457, and F1-score of 0.9407, with sensitivity (0.9611) and specificity (0.9303) demonstrating robust detection performance, particularly in identifying true intrusions, though with slightly reduced overall balance compared to the DT-based model. Overall, these results confirm that the ensemble-based C-IDS framework benefits significantly from integrating clustering with multiple classification techniques. The DT-based hybrid model emerges as the most accurate and reliable, while other combinations also provide strong performance, emphasizing the effectiveness of ensemble learning in cybersecurity IDS.

The findings highlight the advantages of integrating clustering and ensemble learning techniques in cybersecurity applications. AHC effectively groups similar network traffic patterns, which allows the subsequent ensemble classifier to make more informed and precise predictions. Moreover, the stacking ensemble methodology combines multiple classifiers' outputs, reducing bias and variance, and further improving overall detection performance. The high accuracy achieved by the proposed C-IDS demonstrates its potential for practical implementation in real-world network environments, offering reliable, scalable, and adaptive cybersecurity protection.

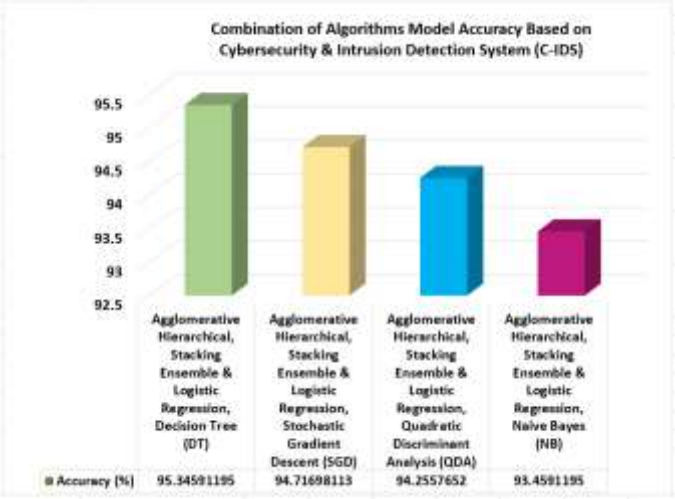


Figure 13: Combination of Algorithms Model Accuracy of (C-IDS)

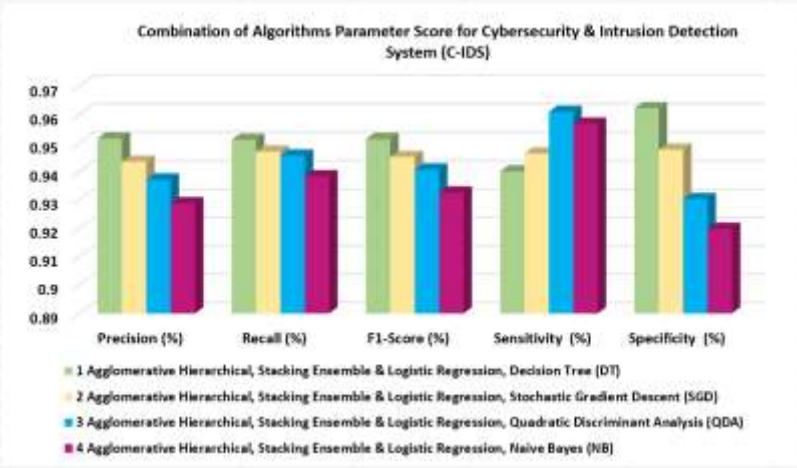


Figure 14: Combination of Algorithms Parameter Score (C-IDS)

The predictive performance of the hybrid model combinations has reached its peak, as reflected in the

accuracy comparison graph, which highlights the highest level of detection precision achieved by the models. The system's balanced predictive capability is further demonstrated by consistently strong values across key evaluation metrics, including precision, F1-score, recall, sensitivity, and specificity. While the ensemble-based C-IDS already demonstrates exceptional performance in detecting and classifying malicious network activity, these metrics can be further optimized or fine-tuned to meet specific cybersecurity requirements, thereby enhancing the overall robustness and effectiveness of the IDS.

5. COMPARATIVE ANALYSIS

A comparative analysis was conducted to evaluate the performance of the proposed hybrid ensemble-based Cybersecurity & IDS (C-IDS) against existing methods reported in recent literature. Table 6 summarizes the accuracy achieved by different hybrid techniques, including both classical and advanced ML models. Previous studies by Farzaan et al. (2024) employed a combination of RF and SVM models, achieving AUC scores of 90% and 91%, respectively. Sundaramurthy et al. (2025) implemented a hybrid of RF, Gradient Boosting, and LR, with corresponding AUC values of 88%, 86%, and 91%. Hossain et al. (2025) utilized a combination of Random Forest and XGBoost, resulting in AUC scores of 92% and 84%. While these methods demonstrated reasonable performance, their predictive accuracy remained limited by the constraints of individual classifiers and single-layer ensemble strategies.

In contrast, the proposed methodology, whichof malicious activity is essential for maintaining the integrates AHC with a SEC and a range of MLintegrity, confidentiality, and availability of network models, achieves significantly higher accuracy.systems. To address this critical need, this study has Specifically, the combination of AHC + SEC-LR +developed an ensemble-based Cybersecurity & IDS DT achieved the highest accuracy of 95.35%,(C-IDS) that leverages a hybrid framework combining outperforming all previous studies. Similarly, theAgglomerative Hierarchical Clustering (AHC), AHC + SEC-LR + SGD and AHC + SEC-LR + QDA combinations reached accuracies of 94.72% and 94.26%, respectively, while the AHC + SEC-LR + NB model achieved 93.46%. This comparative evaluation demonstrates that the proposed ensemble-based C-IDS is more effective in detecting and classifying network intrusions compared to existing hybrid models. The superior performance can be attributed to the integration of hierarchical clustering for feature grouping, the stacking ensemble for leveraging multiple base classifiers, and the inclusion of diverse ML models to capture complex patterns in network traffic. Overall, these results confirm that the proposed methodology enhances detection capability, reduces misclassification, and provides a robust framework for real-time cybersecurity threat monitoring.

Table 6: Comparison of hybrid Methods for (C-IDS)

Study / Authors	Hybrid Method	Accuracy
Farzaan et al. (2024) [9]	RF & SVM	AUC = 90%, 91%
Sundaramurthy et al. (2025) [10]	RF &Gradient Boosting & LR	AUC =88%, 86%,91%.
Hossain et al. (2025) [11].	RF & XGBoost	AUC = 92%, 84%
Proposed Method (AHC + SEC-LR + DT))	Agglomerative Hierarchical Clustering + Stacking Ensemble + DT	95.3459 %
Proposed Method (AHC + SEC-LR + SGD)	Agglomerative Hierarchical Clustering + Stacking Ensemble + Stochastic Gradient Descent	94.7169 %
Proposed Method (AHC + SEC-LR + NB)	Agglomerative Hierarchical Clustering + Stacking Ensemble + Naive Bayes	93.4591 %
Proposed Method (AHC + SEC-LR + QDA)	Agglomerative Hierarchical Clustering + Stacking Ensemble + Quadratic Discriminant Analysis	94.2557 %

Cybersecurity threats continue to pose significant risks to organizations worldwide, with intrusions leading to financial losses, data breaches, and operational disruption. Early and accurate detection

Stacking Ensemble Classifiers (SEC), and multiple ML models, including DT, SGD, NB, and QDA. The proposed system demonstrates high predictive accuracy and robustness by integrating clustering for feature structuring, ensemble learning for combining the strengths of multiple classifiers, and sophisticated preprocessing techniques, including feature selection and SMOTE for balancing datasets. Among the evaluated combinations, AHC + SEC-LR + DT achieved the highest accuracy of 95.35%, followed by AHC + SEC-LR + SGD (94.72%), AHC + SEC-LR + QDA (94.26%), and AHC + SEC-LR + NB (93.46%). Performance metrics such as precision, recall, F1-score, sensitivity, and specificity further validate the model's balanced and reliable detection capability, highlighting its effectiveness in real-time IDS scenarios. The significance of this research lies in its ability to enhance network security

CONCLUSION AND FUTURE DIRECTIONS

through intelligent, automated, and adaptive detection mechanisms. By accurately identifying both known and unknown threats, the ensemble-based C-IDS reduces false alarms, improves threat response times, and provides actionable insights for cybersecurity professionals. The hybrid framework also offers flexibility for future extensions, such as incorporating additional ML models, deploying deep learning techniques, or integrating real-time monitoring systems to improve scalability and adaptability. Overall, the proposed C-IDS represents a substantial advancement in IDS, providing a comprehensive, efficient, and high-performing solution for modern cybersecurity challenges. Its implementation has the potential to strengthen organizational defenses, minimize risks, and ensure reliable protection of critical network infrastructure, ultimately contributing to safer and more resilient digital environments. Future work may focus on integrating deep learning techniques, expanding datasets across diverse network environments, and developing explainable and real-time response capabilities to enhance scalability, adaptability, and interpretability. These improvements could make the system a comprehensive solution for modern Cybersecurity challenges.

REFERENCE

[1] Perez, S.I.; Criado, R. Increasing the Effectiveness of Network Intrusion Detection Systems (NIDSs) by Using Multiplex Networks and Visibility Graphs. Mathematics 2022, 11, 107.

[2] Ahmad, Z.; Khan, A.S.; Shiang, C.W.; Abdullah, J.; Ahmad, F. Network intrusion detection system: A systematic study of machine learning and deep learning approaches. Trans. Emerg. Telecommun. Technol. 2021, 32, e4150.

[3] Ahmad, Z.; Khan, A.S.; Shiang, C.W.; Abdullah, J.; Ahmad, F. Network intrusion detection system: A systematic study of machine learning and deep learning approaches. Trans. Emerg. Telecommun. Technol. 2021, 32, e4150.

[4] Mahmood, S., Mohsin, S. M., & Akber, S. M. A. (2020). Network security issues of data link layer: An overview. In Proceedings of the 3rd International Conference on Computing, Mathematics and Engineering Technologies (iCoMET) (pp. 1–6).

[5] Muqaddas, M., Majeed, S., Hira, S., & Mumtaz, G. (2024). A Systematic Literature Review on Performance Evaluation of SQL and NoSQL Database Architectures. Journal of Compu-ting & Biomedical Informatics, 7(02). https://jcbi.org/index.php/Main/article/view/548/502

[6] Salahuddin, M., Zaman, F. U., Mumtaz, G., Khan, M. Z., Kainat, M., Hira, S., Parveen, F., & Mahmood, R. (2025). Integrating network intrusion detection with machine learning techniques for enhanced network security. Spectrum of Engineering Sciences, 3(4), 612–625. https://sesjournal.com/index.php/1/article/view/286

[7] Zahra, W. U., Zaman, F. U., Mumtaz, G., Salahuddin, M., Khan, M. Z., Sultan, S. A., Hira, S., & Parveen, F. (2025). Approaches to predict cardiovascular issue using machine learning method. Spectrum of Engineering Sciences, 3(4), 417–429.

[8] Parveen, F., Iqbal, S., Mumtaz, G., & Salahuddin, M. (2024). Real-time intrusion detection with deep learning: Analyzing the UNR intrusion detection dataset. Journal of Computing & Biomedical Informatics, 7(02). https://jcbi.org/index.php/Main/article/view/554

[9] Farzaan, M. A. M., Ghanem, M. C., El-Hajjar, A., & Ratnayake, D. N. (2024). AI-Enabled System for Efficient and Effective Cyber Incident Detection and Response in Cloud Environments. IEEE / preprint (arXiv).

[10] S. K. Sundaramurthy, N. Ravichandran, A. C. Inaganti, and R. Muppalaneni, "AI-Driven Threat Detection: Leveraging Machine Learning for Real-Time Cybersecurity in Cloud Environments," Artificial Intelligence and Machine Learning Review, vol. 6, no. 1, pp. 23–43, Jan. 2025.

[11] Z. Hossain, M. Emran Hossain, N. Ahmed, M. Farhad Kabir & I. S. Hossain, "AI-Powered Framework for Real-time Threat Detection

and Response in Cloud Infrastructure," Formosa Journal of Multidisciplinary Research (FJMR), vol. 4, no. 4, pp. 1859-1874, Apr. 2025. DOI:10.55927/fjmr.v4i4.168

[12] Afzal, M., Salahuddin, M., Hira, S., Sultan, M. F., Ahmad, S. Z., & Iqbal, M. W. (2024). A systematic literature review of understanding the human-computer interaction collaboration with user experience design. Bulletin of Business and Economics, 13(2), 723–729. https://doi.org/10.61506/01.00386

[13] Mahmood, R., Mustafa, S., Asif, H., Raza, A., & Salahuddin, M. (n.d.). Leveraging artificial intelligence to optimize software project management: Enhancing efficiency, risk mitigation, and decision-making. Contemporary Journal. https://doi.org/10.12345/f42z1z57

[14] Zaman, F.U., Khan, M.Z., Imroz, A., Khan, A.A., Salahuddin, M. and Kainat, M., 2024, December. Student Performance Analysis in Higher Education Using Integrated Approach of Machine Learning Techniques. In 2024 International Conference on Sustainable Technology and Engineering (i-COSTE) (pp. 1-6). IEEE

[15] Abbas, A., Salahuddin, M., Khan, M. Z., & others. (2025). Machine learning-based hybrid technique to enhance cyber-attack perspective. Journal of Cloud Computing, 14, Article 57. https://doi.org/10.1186/s13677-025-00782-5

[16] Khan, M. Z., Shaikh, S. A., Khan, A. A., Imroz, A., Salahuddin, M., Bhatti, P., & Bhatti, S. D. (2025). Optimizing heart disease forecasting: Bridging gaps in interpretability, efficiency, and scalability using machine learning. Biomedical Materials & Devices. https://doi.org/10.1007/s44174 025 00504 0

[17] Khan, M. Z., Shaikh, S. A., Khan, A. A., Imroz, A., Salahuddin, M., Bhatti, P., & Bhatti, S. D. (2025). Artificial intelligence in dermatology: Current applications and future innovations. [PDF]. ResearchGate.https://www.researchgate.net/publication/396020302_ARTIFICIAL_INTELLIGENCE_IN_DERMATOLOGY_CURRENT_APPLICATIONS_AND_FUTURE_INNOVATIONS

[18] H. Ahmad, G. Mumtaz, and M. Salahuddin, "A Hybrid Deep Learning Model for High-Accuracy Brain Tumor," Al-Aasar, vol. 2, no. 3, pp. 1–15, Aug. 2025, doi: 10.63878/aaj737.

[19] Moeez, M., Mahmood, R., Asif, H., Iqbal, M. W., Hamid, K., Ali, U., & Khan, N. (2024). Comprehensive Analysis of DevOps: Integration, Automation, Collaboration, and Continuous Delivery. Bulletin of Business and Economics (BBE), 13(1).

[20] Amna Ilyas, Fahim Uz Zaman, Muqaddas Salahuddin, Faraz Ahmad Zia, Muhammad Zohaib Khan, Sammia Hira, & Muhammad Ather Ameen. (2025). AI Driven Intrusion Detection Using Machine Learning: An Anomaly Based Analysis of Network Traffic. Spectrum of Engineering Sciences, 3(12), 664–679. Retrieved from https://thesesjournal.com/index.php/1/article/view/1708/1288.

[21] Al-A'araji, N.H.; Al-Mamory, S.O.; Al-Shakarchi, A.H. Classification and clustering based ensemble techniques for intrusion detection systems: A survey. J. Physics Conf. Ser. 2021, 1818, 012106.

[22] Thockchom, N.; Singh, M.; Nandi, U. A novel ensemble learning-based model for network intrusion detection. Complex Intell. Syst. 2023, 9, 5693–5714.

[23] Alotaibi, Y.; Ilyas, M. Ensemble-learning framework for intrusion detection to enhance internet of things' devices security. Sensors 2023, 23, 5568.

[24] Mirsky, Y.; Doitshman, T.; Elovici, Y.; Shabtai, A. Kitsune: An ensemble of autoencoders for online network intrusion detection. arXiv 2018, arXiv:1802.09089.

[25] Lazzarini, R.; Tianfield, H.; Charissis, V. A stacking ensemble of deep learning models for iot intrusion detection. Knowl.-Based Syst. 2023, 279, 110941.

[26] Mahfouz, A.; Abuhussein, A.; Venugopal, D.; Shiva, S. Ensemble classifiers for network intrusion detection using a novel network attack dataset. Future Internet 2020, 12, 180.

[27] Kumar, D. Cybersecurity Intrusion Detection Dataset (Dataset). Kaggle. https://www.kaggle.com/datasets/dnkumars/cybersecurity-intrusion-detection-dataset

[28] Elzaridi, K., & Kurnaz, S. (n.d.). Integration between network intrusion detection and machine learning techniques to optimize network security. Altinbas University, Istanbul, Turkey.

[29] Built In. (2011). Feature importance. https://builtin.com/data-science/feature-importance

[30] Khan, M. Z., Shaikh, S. A., Shaikh, M. A., Khatri, K. K., Rauf, M. A., Kalhoro, A., & Adnan, M. (2023). Performance analysis of machine learning algorithms for credit card fraud detection. International Journal of Online and Biomedical Engineering (iJOE), 19(3), 82–98. https://doi.org/10.3991/ijoe.v19i03.35331

[31] TutorialsPoint. (n.d.). Stochastic gradient descent (SGD). https://www.tutorialspoint.com/scikit_learn/scikit_learn_stochastic_gradient_descent.htm

[32] Baig, M. A., Shaikh, S. A., Khatri, K. K., Shaikh, M. A., Khan, M. Z., & Rauf, A. (2023). Prediction of students' performance level using an integrated approach of ML algorithms. International Journal of Emerging Technologies in Learning (iJET), 18(1), 216–234. https://doi.org/10.3991/ijet.v18i01.35339

[33] Towards Data Science. (n.d.). Introduction to logistic regression. https://towardsdatascience.com/introduction-to- logistic-regression-66248243c148

[34] Khan, M. Z., Khan, A. A., Laghari, A. A., Shaikh, Z. A., Khani, M. A. K., Morkovkin, D., Gavel, O., Shkodinsky, .,Makar, S., & Taburov, D. (2022). Comparative case study: An evaluation of performance computation between support vector machine, k-nearest neighbors, k-means, and principal component analysis. Journal of Tianjin University Science and Technology.

[35] Siddiqui, M., Kalwar, H. A., Khan, M. Z., Khan, M A., Imroz, A., Kalwar, M. A., & Marri, H. B. (2023) Performance analysis for the diagnosis of COVID-19 prediction by mathematical modeling and simulation. International Journal of Artificial Intelligence & Mathematical Sciences http://ijaims.smiu.edu.pk/ijaims/index.php/AIMS/article/view/47