SAVUNMA BİLİMLERİ DERGİSİ
The Journal of Defense Sciences

https://dergipark.org.tr/tr/pub/khosbd

A High-Precision Cybersecurity Model Based on Stacked Ensemble Learning

Yığınlanmış Topluluk Öğrenmesi ile Yüksek Hassasiyetli Bir Siber Güvenlik Modeli

Faruk AYATA [1*]

[1]*Van Yüzüncü Yıl Üniversitesi, Başkale Meslek Yüksekokulu, Bilgisayar Teknolojileri Bölümü, Van, Türkiye*

Makale Bilgisi

Araştırma makalesi
Başvuru: 30.12.2025
Düzeltme: 11.02.2026
Kabul: 02.03.2026

Keywords

Stacked ensemble learning
K-Fold Cross-Validation
Cybersecurity
Attack classification

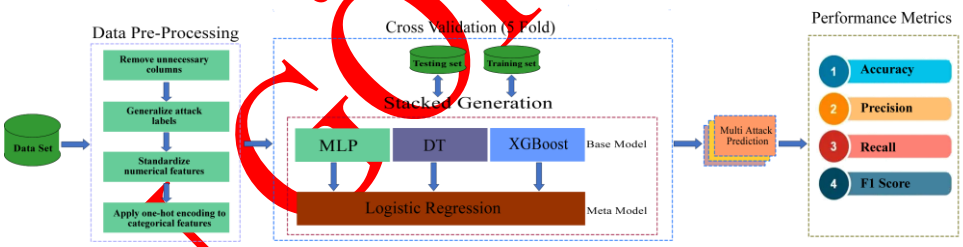
Anahtar Kelimeler

Yığınlanmış Topluluk Öğrenimi
K-Katlı Çapraz Doğrulama
Siber Güvenlik
Saldırı Sınıflandırma

Highlights

This study presents a stacking ensemble model for cyber attack detection using MLP, DT, and XGBoost with LR as the meta-learner. After feature selection and encoding-based preprocessing, the model achieves high multi-attack classification performance in terms of accuracy, precision, recall, and F1-score.

Graphical Abstract



Abstract

The success of Artificial Intelligence (AI) techniques across various domains has significantly increased interest in their application to cyber security. Although Machine Learning (ML) methods are effective in detecting malicious activities, certain challenges may negatively affect performance accuracy. Developing an effective Intrusion Detection System (IDS) requires careful selection of appropriate techniques and features. In this study, a Stacking Ensemble Learning (SEL) model was developed to classify network attacks using Decision Tree (DT), XGBoost, and Multilayer Perceptron (MLP) as base learners, with Logistic Regression (LR) employed as the meta-learner. To enhance generalization capability and prevent overfitting, K-fold cross-validation was applied. The proposed SEL model achieved an accuracy of 99.65% on the NSL-KDD dataset and 99.50% on the CICIDS2017 dataset. These findings indicate that combining SEL with K-fold cross-validation can provide high accuracy in network attack detection and effectively identify cyber threats.

Özet

Yapay Zekâ (YZ) tekniklerinin farklı alanlardaki başarısı, siber güvenlikte kullanımına olan ilgiyi artırmıştır. Makine Öğrenmesi (ML) yöntemleri kötü amaçlı faaliyetlerin tespitinde etkili olsa da bazı zorluklar performansı olumsuz etkileyebilir. Etkili bir Saldırı Tespit Sistemi (İSS) geliştirmek, uygun teknik ve özelliklerin dikkatli seçimini gerektirir. Bu çalışmada, ağ saldırılarını sınıflandırmak amacıyla Karar Ağacı (DT), XGBoost ve Çok Katmanlı Algılayıcı (MLP) temel modelleri ile Lojistik Regresyon (LR) meta modeli kullanılarak Yığınlı Topluluk Öğrenmesi (SEL) modeli geliştirilmiştir. Modelin genelleme yeteneğini artırmak ve aşırı uyumu önlemek için K-Katlı çapraz doğrulama uygulanmıştır. SEL modeli, NSL-KDD veri kümesinde %99,65 ve CICIDS2017 veri kümesinde %99,50 doğruluk elde etmiştir. Bulgular, SEL modelinin K-katlı çapraz doğrulama ile birlikte ağ saldırılarının tespitinde yüksek doğruluk sağladığını ve siber tehditlerin belirlenmesinde etkili olduğunu göstermektedir.

* Corresponding author, e-mail: farukayata@yyu.edu.tr

DOI:10.17134/khosbd.1852311

Elektronik/Online ISSN: 2148-1776
Basılı/Printed ISSN: 1303-6831

1. INTRODUCTION

In today's world where technology is advancing very rapidly, it is becoming very difficult for users to hide and protect their important information [1,2]. The development of technology increases cyber attacks and the development of the technological structure behind these attacks. The increase in the variety and complexity of cyber attacks makes traditional security solutions inadequate [3,4]. Cyber attacks can cause data loss, damage to critical infrastructures and threaten the security of institutions, causing great socioeconomic damage. For this reason, cyber security measures need to be strengthened and their continuity ensured in every field [5,6].

Cyber attacks adapt very quickly to technological developments and develop rapidly. Every day, a new phishing attack, denial of service (DoS) attack and data breach emerge. Such attacks cause large amounts of data loss and financial losses [7,8]. Traditional IDSs (Intrusion Detection Systems) are inadequate in the face of these complex and renewed attacks [9].

With the use of ML methods in the field of cyber security, great successes are achieved in preventing many attacks. ML methods offer the ability to uncover hidden patterns in large data sets, which helps in detecting attacks and decision-making processes in network traffic activities [10]. Although the use of ML methods in IDS is effective in detecting malicious activities, some performance problems can be experienced due to reasons such as data quality, algorithm complexity and real-time detection requirements [11-14]. In order for ML methods to be used effectively in cyber security systems, continuous research and development should be carried out and these methods should be strengthened by supporting them with different algorithms and integrated models. These approaches can help in determining the challenges related to data quality and the appropriate ML techniques in the attack detection process. Some of the studies in this field are given below:

In recent years, machine learning (ML) approaches have increasingly been adopted to improve the detection and classification of cyber threats. Early studies focused primarily on optimizing individual ML algorithms to enhance detection performance and adaptability under varying data conditions. For instance, Dovbysh et al. introduced an extreme information-based ML approach that optimizes cyberattack detection by adapting to different learning matrix conditions and employing a modified Kullback information measure to refine model parameters. Their simulation results demonstrated high accuracy (ACC), highlighting the importance of parameter optimization and adaptive recognition strategies in dynamic cyber environments [15]. Similarly, Reddy et al. comparatively evaluated multiple ML algorithms, including Support Vector Machine (SVM), Convolutional Neural Network (CNN), Random Forest (RF), and Artificial Neural Network (ANN), using the CICIDS2017 dataset. Their findings indicated that RF outperformed other models in terms of both accuracy and efficiency, emphasizing the strength of tree-based ensemble approaches in intrusion detection tasks [17].

Beyond single-model optimization, hybrid and meta-heuristic strategies have been explored to

further enhance classification performance. Albakri et al. proposed a blockchain-supported ML framework that integrates a hybrid meta-heuristic feature selection mechanism with a semi-recurrent neural network classifier. By combining blockchain security features with ML-based detection, their approach aimed to improve both robustness and classification success, reporting high performance in their experimental analysis [16]. These studies demonstrate a shift from purely algorithm-centric optimization toward integrated and security-aware learning architectures.

As the limitations of individual models became more apparent—particularly regarding generalization capability and stability across datasets—ensemble learning techniques emerged as a powerful alternative. Stacked models, in particular, have been widely applied in different domains to improve predictive accuracy by combining multiple base learners. Lu et al. proposed a novel stacking-based model for daily flow prediction, integrating RF, AdaBoost, and XGBoost algorithms. Their results showed that the stacked structure outperformed both baseline and traditional ensemble models, demonstrating the effectiveness of multi-learner aggregation strategies [18]. Similar improvements were observed in healthcare and biomedical prediction tasks. Ghasemieh et al. developed a Stacking Ensemble Learner (SEL) model for predicting emergency readmission in heart disease patients, achieving an 88% success rate [20]. Zhu et al. combined RF, XGBoost, and Extra Trees algorithms in a stacked configuration to predict olanzapine blood levels, reporting an ACC rate of 63.4% [21].

Stacked ensemble models have also shown significant performance gains in classification problems beyond cybersecurity. Jaiyeoba et al. applied stacking techniques for skin disease classification using base learners such as Naïve Bayes (NB), SVM, Decision Tree (DT), RF, and Gradient Boosting. While individual model accuracies ranged between 85.41% and 98.61%, the stacked model improved the overall ACC to 99.30%, illustrating the synergistic effect of combining diverse classifiers [19]. Likewise, Fernandes et al. employed SVM, KNN, and DT as base learners with Logistic Regression (LR) as a meta-learner to predict student performance, achieving an ACC of 84.15% under 10-fold cross-validation [22].

Collectively, these studies reveal a clear methodological evolution: from optimizing single ML models, to integrating hybrid and meta-heuristic mechanisms, and ultimately to leveraging stacked ensemble architectures to enhance predictive robustness and accuracy. However, although stacked ensemble approaches have demonstrated strong performance across various domains, their structured and performance-oriented adaptation to cybersecurity detection systems remains an area that warrants further systematic investigation.

As understood from the literature, hybrid models in AI are effective in detecting cyber threats. For this reason, hybrid models are employed in this study. In summary, the contributions of this study can be listed as follows:

✓ The use of AI in cybersecurity: This study effectively demonstrates the application of AI and ML techniques in cybersecurity, offering a reliable method for detecting malicious activities.

✓ High performance with SEL: By using DT, XGBoost, and MLP algorithms together in the SEL model, higher success rates are achieved, yielding effective results in detecting cyberattacks.

✓ High success rate: The combination of SEL and K-Fold CV has achieved ACC rates of 99.65% on the NSL-KDD dataset and 99.50% on the CICIDS2017 dataset. These success rates demonstrate the effectiveness of SEL in detecting cyber threats.

✓ Improved generalization ability: The K-Fold CV method has increased the generalization ability of SEL, preventing overfitting and producing effective results on different datasets.

✓ An effective tool for cybersecurity: The study's findings, with high ACC rates, reveal that SEL is an effective tool for detecting cyber threats, making a significant contribution to the development of IDS.

✓ Practical value: The study emphasizes how critical it is to select the right techniques and features in IDS development for detecting cyberattacks, guiding future cyber security studies.

✓ The rest of the study is organized as follows. In Section 2, the methodology of the developed system, datasets, and a comprehensive explanation of the methods used are presented. Section 3 evaluates and discusses the findings of the study. The final section includes conclusions and suggestions for the future.

2. MATERIALS AND METHODS

2.1 Datasets

Producing high-quality datasets required for the training of ML methods is quite challenging. Two commonly used datasets in the IDS field are NSL-KDD and CICIDS2017. These datasets were also chosen for this study.

NSL-KDD: This dataset is an improved and cleaned version of the KDD'99 dataset from the UNB. During the creation of the KDD dataset, extensive network traffic data was collected [23]. This dataset contains 42 different features, 22 different attack types, and 148,517 instances. Some of this information is presented in Table 1.

Table 1: NSL-KDD dataset descriptions.

Main Attacks	22 Attacks Classes
DoS	back, land, neptune, pod, smurf, teardrop
R2L	ftp write, guess passwd, imap, multhop, phf, spy, warezclient, warezmaster
U2R	buffer overflow, perl, loadmodule, rootkit
Probe	ipsweep, nmap, portsweep, satan

CICIDS2017: Created by the Canadian Institute for Cyber security (CICIDS) for network attack detection research, this dataset contains labeled data samples, covering both benign and attack traffic collected over five days. It includes 2,830,743 records and 79 features. In this study, 20% of the CICIDS2017 dataset was used. While the dataset provides a rich source of information for understanding network traffic and cyberattacks, it is crucial for modeling real-world cyber security scenarios. Some of the dataset information is presented in Table 2.

Table 2: CICIDS2017 dataset descriptions.

Category		Total
BENIGN	BENIGN	2273097
DoS	DDoS	128027
	DoS slowloris	5796

	DoS SlowHTTPTest	5499
	DoS Hulk	231073
	DoS GoldenEye	10293
	Heartbleed	11
Port Scan	Port Scan	158930
Bot	Bot	1966
Brute Force	FTP-P	7938
	SSH-P	5897
Web Attacks	WA-Brute Force	1507
	WA-XSS	652
	WA-SQL Injection	21
Infiltration	Infiltration	36

2.2 K-Fold CV

It is a method frequently used in data mining. This method is a cross-validation method applied to achieve high ACC in dividing the data into training and testing. K-Fold CV divides randomly selected sample data into K groups and K trials are performed. In each trial, the Kth group is used as test data, while the remaining parts are considered as training data. The process continues until the K value is completed, and thus the performance of the model is evaluated more accurately [24].

2.3 ML methods

DT; A nonlinear predictive modeling tool used in statistics, data mining, and ML. DT intuitively mimic the human decision-making process better than other algorithms, making them easier to understand and interpret. The trees divide the dataset into branches, which are then further subdivided into smaller subsets based on different attributes. This splitting continues until the algorithm determines that further subdivision adds no value or meets a predefined stopping criterion [27].

XGBoost; In the field of ML, XGBoost is a gradient boosting library that offers high performance and speed, especially for large datasets. XGBoost improves efficiency through innovative features such as parallel processing, pre-pruning of tree branches, and storage structures optimized for irregular data. These features provide regularization against overfitting, making XGBoost a popular choice in various data science competitions and real-world applications, producing effective results [28].

MLP; A widely used neural network model for classification problems. MLP consists of multiple layers of neurons: an input layer for the initial data, one or more hidden layers to discover complex patterns, and an output layer to produce the final result [29, 30]. Neurons in each layer contribute to the model by applying specific weights to their inputs through activation functions that introduce nonlinearity. Data is processed step by step, ACC is calculated, and then weight adjustments are made [31]. To reduce the difference between predicted and actual results, the model goes through an iterative process involving optimization techniques, weight, and parameter adjustments. Over time, this repeated learning process allows MLP to improve its ability to classify data correctly [32].

2.4 Performance Measurements

The ACC of a classification can be assessed by calculating the number of correctly classified class correctly identified non-class examples (true negatives), along with the number of examples incorrectly classified as part of the class (false positives) or missed as class examples

(false negatives) [33]. All the performance measurement formulas used in the study are presented in Table 3.

Table 3: Classification performance measurement formulas.

Measure	Formula
ACC	$\frac{tp + tn}{tp + fn + fp + tn}$
PRC	$\frac{tp}{tp + fp}$
Recall	$\frac{tp}{tp + fn}$
F1 score	$\frac{2 * (precision * Sensitivity)}{precision + Sensitivity}$

3. EXPERIMENTAL STUDIES

The system's operational architecture is shown in Figure 1. This architecture consists of three main phases for the network attack detection system: data preprocessing, model training, and performance evaluation. In the first phase, unnecessary columns are removed from the dataset, attack labels are generalized, and numerical features are standardized. Then, categorical data is transformed using the one-hot encoding method, enabling algorithms to work more efficiently with such data. During the model training phase, the data is split into 80% training and 20% test sets. In the base model section of the SEL structure, DT, MLP and XGBoost are used. DT visually and understandably segments the data to make decisions, increasing the model's interpretability and capturing important data features easily. MLP; is an artificial neural network that can learn complex relationships. It also has deep learning capacity. XGBoost; on the other hand; is high-performance and fast, which makes it stand out among ML algorithms. It is quite strong in feature extraction and model optimization. These three models are combined as the basic model in the flood structure and their outputs are combined with LR, which is selected as the meta model. This combination, which is done with the stacking method, produces more accurate predictions by taking advantage of the strengths of different models. In addition, K-Fold CV is used to prevent the model from over-learning and to increase its generalization ability. In the last stage, the performance of the model is analyzed. The success of the model in attack detection is evaluated with success metrics such as ACC, PRC, recall and F1 score.

3.1 Data Preprocessing

Table 4 shows the preprocessing module for the dataset used in the study. This algorithm processes the raw dataset (H), making it suitable for modeling. First, duplicate records and examples with missing values are removed from the dataset (Hdup and Hmiss). Then, features with zero variance (F) are eliminated from the dataset, as such columns carry no useful information. Next, predetermined unnecessary columns (G) are removed from the dataset. Finally, all remaining features are standardized using z-score normalization, ensuring that the dataset becomes more consistent and balanced. Once these steps are completed, the cleaned and normalized dataset (H') is ready for the modeling process.

Table 4: Pre-Processing Algorithm of the System.

Algorithm 1: Preprocessing Algorithm
Input: Raw dataset H

Output: Preprocessed dataset H'

1. Remove duplicates and missing values:
   - Let Hdup = set of duplicate instances in H
   - Let Hmiss = set of instances with missing values in H
   - $H' = H \setminus (H\text{dup} \cup H\text{miss})$
2. Remove columns with zero variance:
   - Let F = set of features with zero variance in H'
   - $H' = H' \setminus F$
3. Remove specified columns:
   - Let G = set of specified columns to be removed from H'
   - $H' = H' \setminus G$
4. Standardize the dataset:
   - For each feature in H':
     - Apply z-score normalization to the feature
5. Return preprocessed dataset H

stacking, known as multi-stacking, uses multiple stacking layers to provide more complex and deeper learning processes. In this method, the outputs of the learners in each layer are fed into the next layer, resulting in stronger predictive models. Thus, the multi-stacking method enhances prediction performance by combining the strengths of different model combinations and produces better results, especially in more complex datasets. The algorithm for the SEL model used in this study is presented in Table 5.

Figure 1: Working architecture of the system.

3.2 Stacking Ensemble Learning (SEL)

Stacking, a parallel ensemble strategy, allows for the combination of outputs from multiple learners in a multi-layered learning structure [25]. In this approach, a two-layered learning framework is utilized, with the learners in the first layer termed base learners, and those in the second layer called meta-learners [26]. A more advanced version of

Table 5: SEL of the System.

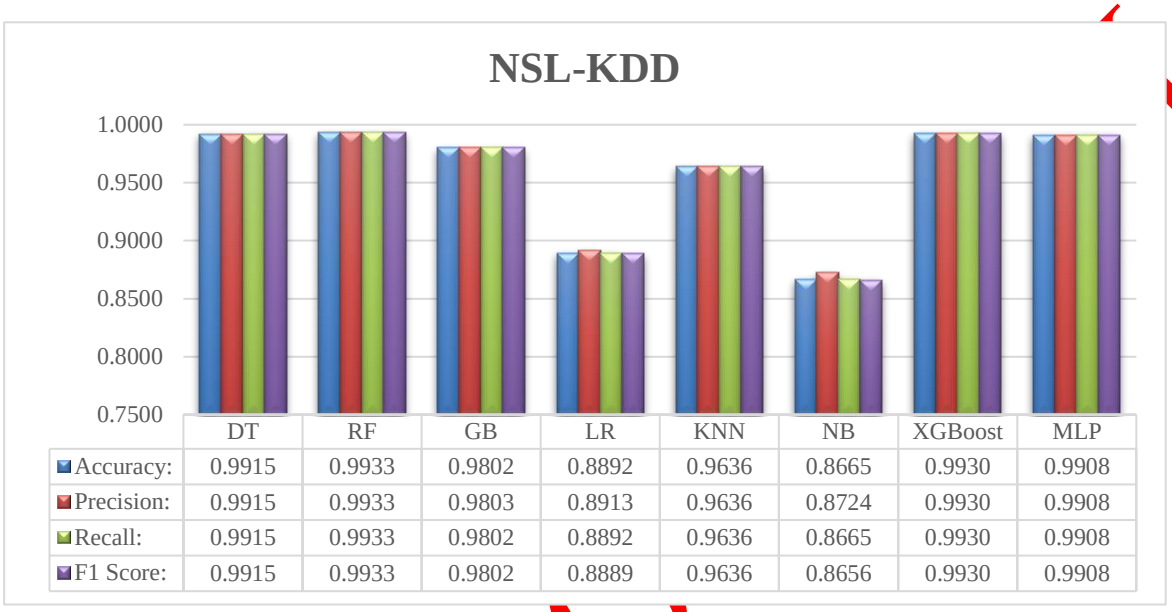
Algorithm 2: Stacked Classifier Training Algorithm

Input: Preprocessed dataset H', Target variable Y
Output: Trained stacked classifier C

1. Split H' into K folds for cross-validation;
2. For each fold k (k = 1, 2, ..., K):
   - The available data were divided into two disjoint subsets: a training set (D′_train) and a validation set (H′_val);
   - Train DT, XGBoost, and MLP on the training set H'train ;

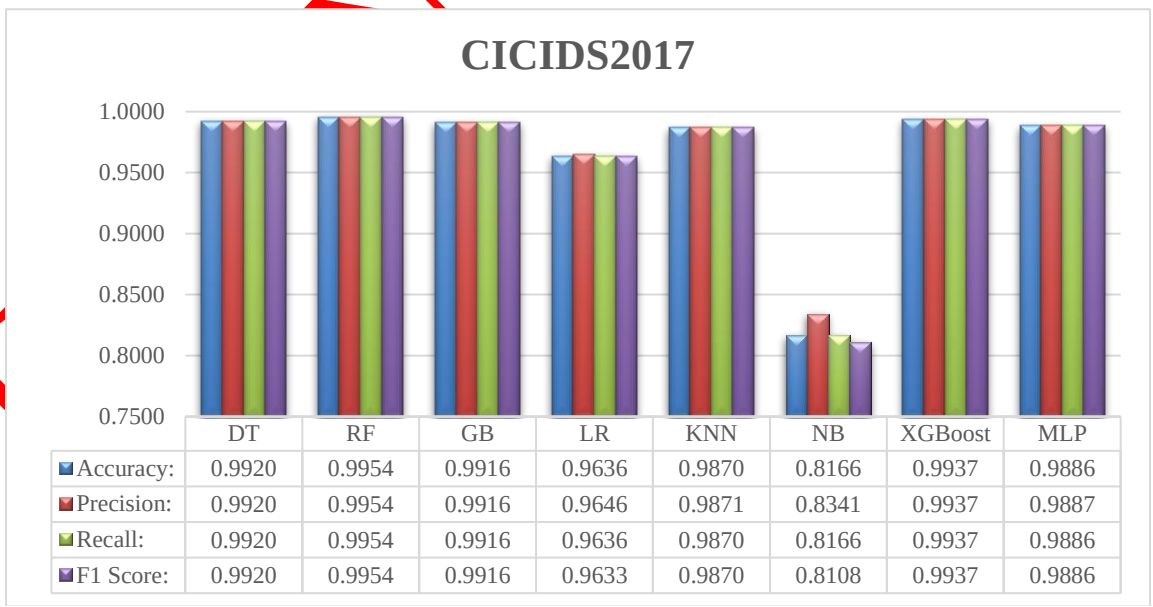
- The base models were employed to produce out-of-sample predictions on H′_val;
- Out-of-sample predictions were utilized to train the LR-based meta-model;

3. After cross-validation, combine the models trained in each fold to create the final base models;
4. Train the final LR meta-model using the combined out-of-sample predictions from all folds;
5. Use the trained base models to make predictions on the testing set;
6. Combine predictions using the trained meta-model (LR);
7. Compute performance metrics on the testing set;
8. return Trained stacked classifier C;



	DT	RF	GB	LR	KNN	NB	XGBoost	MLP
Accuracy:	0.9915	0.9933	0.9802	0.8892	0.9636	0.8665	0.9930	0.9908
Precision:	0.9915	0.9933	0.9803	0.8913	0.9636	0.8724	0.9930	0.9908
Recall:	0.9915	0.9933	0.9802	0.8892	0.9636	0.8665	0.9930	0.9908
F1 Score:	0.9915	0.9933	0.9802	0.8889	0.9636	0.8656	0.9930	0.9908

Figure 2: Metric results of various machine learning methods with NSL-KDD dataset (without SEL+K-Fold).



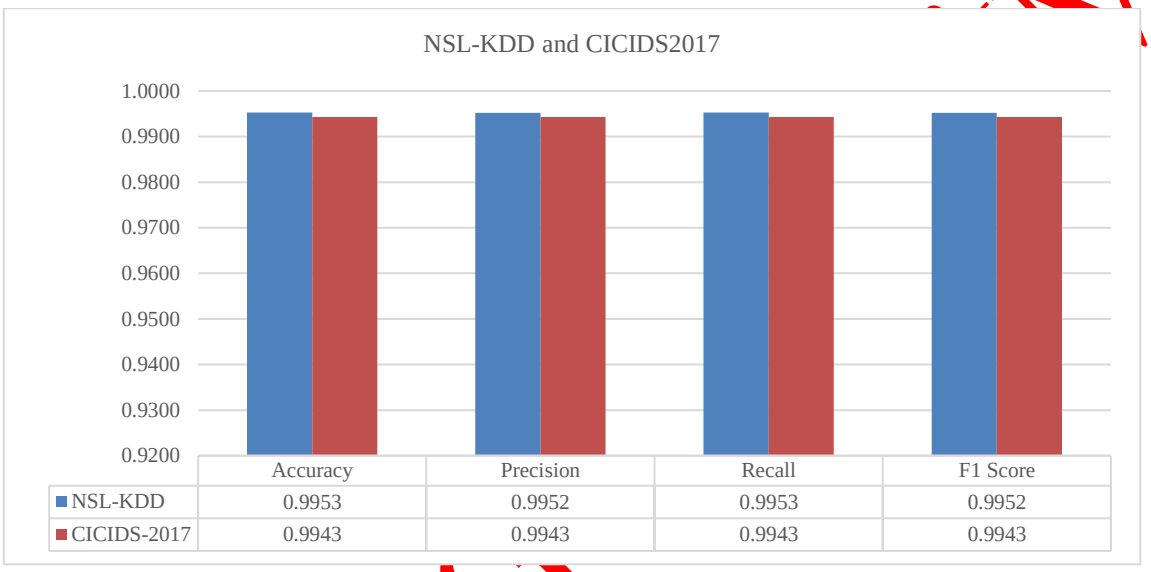
	DT	RF	GB	LR	KNN	NB	XGBoost	MLP
Accuracy:	0.9920	0.9954	0.9916	0.9636	0.9870	0.8166	0.9937	0.9886
Precision:	0.9920	0.9954	0.9916	0.9646	0.9871	0.8341	0.9937	0.9887
Recall:	0.9920	0.9954	0.9916	0.9636	0.9870	0.8166	0.9937	0.9886
F1 Score:	0.9920	0.9954	0.9916	0.9633	0.9870	0.8108	0.9937	0.9886

Figure 3: Metric results of various machine learning methods with CICIDS2017 dataset (without SEL Fold).

4. RESULTS AND DISCUSSION

The ACC of multiple classifiers is assessed for the CICIDS2017 and NSL-KDD datasets used in the SEL model. Figures 2 and 3 present the ACC-based comparison of different ML algorithms used in IDS for the NSL-KDD and CICIDS2017 datasets. The findings are as follows: DT (99.15%-99.20%), RF (99.33%-99.54%), GB (98.02%-99.16%), LR (88.92%-96.36%),

KNN (99.36%-98.70%), NB (86.65%-81.16%), XGBoost (99.30%-99.37%), and MLP (99.08%-98.86%). Several popular algorithms were tested for the SEL model, and the top three algorithms with the highest ACC, XGBoost, DT, and MLP, were selected for further integration.



	Accuracy	Precision	Recall	F1 Score
NSL-KDD	0.9953	0.9952	0.9953	0.9952
CICIDS-2017	0.9943	0.9943	0.9943	0.9943

Figure 4: Comparison of success metrics of NSL-KDD and CICIDS2017 datasets applied with SEL model (without K-Fold).

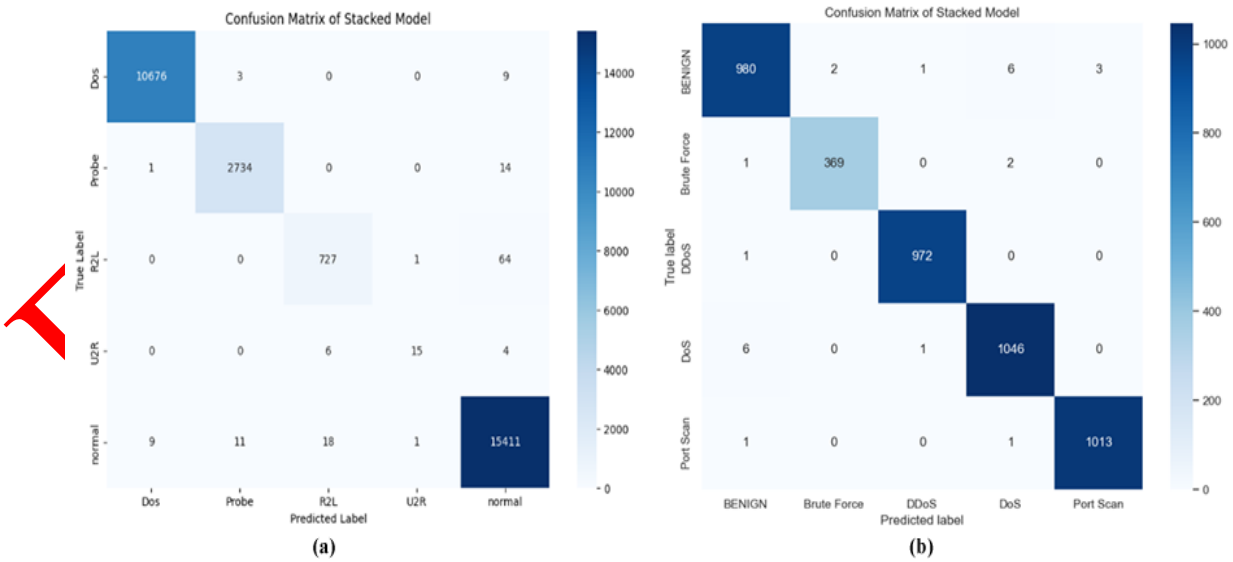
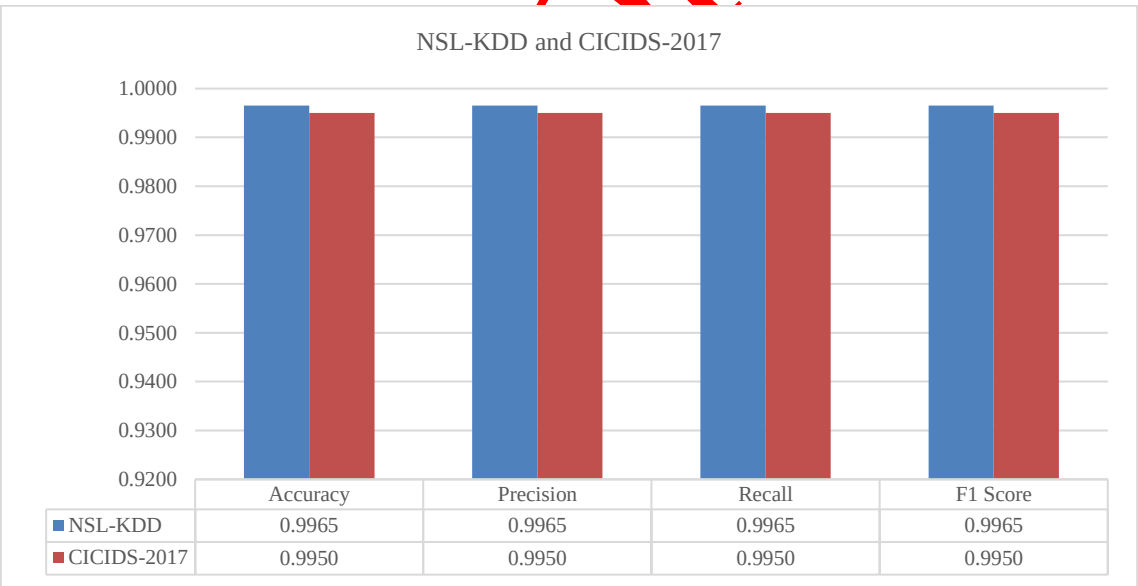


Figure 5: Comparison of confusion matrix of NSL-KDD and CICIDS2017 datasets applied with SEL model (without K-Fold).

In Figure 4, the performance results of the SEL model on the NSL-KDD and CICIDS2017 datasets are compared in detail. For key metrics such as ACC, precision, recall, and F1 score, a success rate of over 99% was achieved for both datasets. These high success rates demonstrate that the SEL model not only makes accurate predictions but also operates reliably in critical areas such as cybersecurity, where error tolerance is very low. This confirms the model's effectiveness in detecting cyber threats with high PRCand minimal errors.

In Figure 5 (a), the confusion matrix for the SEL model applied to the NSL-KDD dataset is shown. DoS attacks were classified with high ACC, with 10,676 correct predictions. Probe attacks were also successfully classified, with 2,734 correct predictions. However, the model performed poorly in R2L and U2R attacks, with only 727 correct predictions for R2L and just 6 correct predictions for U2R. Normal traffic achieved near-perfect results with 15,411 correct predictions, highlighting the model's strength in distinguishing normal traffic from attacks.

In Figure 5 (b), the confusion matrix for the CICIDS2017 dataset shows that BENIGN, DoS, DDoS, Brute Force, and Port Scan attacks were also classified with high ACC. Particularly, Port Scan was detected with near-perfect ACC, with 1,013 correct predictions. These results demonstrate that the SEL model generally delivers strong performance across different attack types.



	Accuracy	Precision	Recall	F1 Score
NSL-KDD	0.9965	0.9965	0.9965	0.9965
CICIDS-2017	0.9950	0.9950	0.9950	0.9950

Figure 6: Comparison of success metrics of NSL-KDD and CICIDS2017 datasets applied with SEL+ K-Fold model.

In Figure 6, the comparison of the performance metrics achieved by applying the SEL + K-Fold model to the NSL-KDD and CICIDS2017 datasets is shown. For both datasets, the values for all metrics exceeded 99.50%, demonstrating the strength and reliability of the SEL + K-Fold model in detecting network attacks. The contribution of the K-Fold CV method to the model's success is significant, as it reduces the risk of overfitting by training the model on different data subsets and enhances the model's generalization ability. Achieving such high

results across all metrics clearly highlights the model's effectiveness and reliability in identifying network attacks.

In Figure 7 (a), the confusion matrix for the SEL + K-Fold model applied to the NSL-KDD dataset is displayed. The DoS attacks were classified with very high ACC, achieving 10,685 correct predictions. Probe attacks were also highly successful, with 2,740 correct predictions. For R2L attacks, 747 correct predictions were made, though some misclassifications still occurred. U2R attacks had only 18 correct predictions. Normal traffic showed near-perfect ACC, with 15,412 correct predictions.

Figure 7: Comparison of confusion matrix of NSL-KDD and CICIDS2017 datasets applied with SEL + K-Fold model.

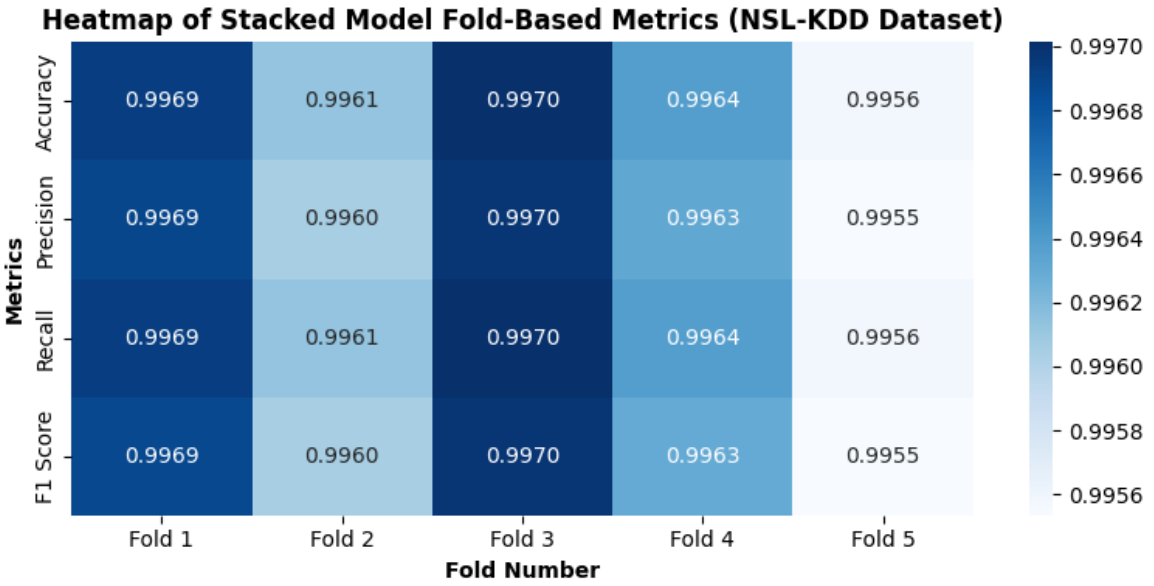


Figure 8: Success metrics of each fold of the SEL+K-fold model applied to the NSL-KDD dataset.

In Figure 7 (b), the performance of the SEL + K-Fold model on the CICIDS2017 dataset is illustrated. BENIGN traffic was classified with high ACC, achieving 982 correct predictions. Brute Force, DoS, and DDoS attacks were also detected with high ACC. Port Scan attacks were detected nearly perfectly. These results indicate that the SEL + K-Fold model performed successfully on both datasets, demonstrating its effectiveness in identifying various types of network traffic and attacks.

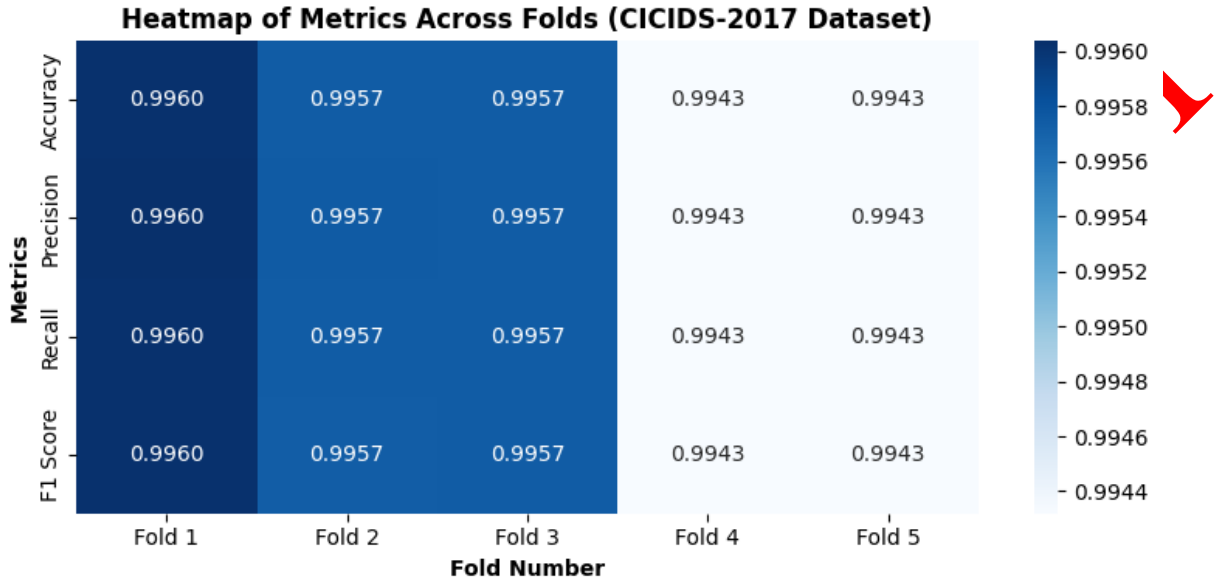


Figure 9: Success metrics of each fold of the SEL+K-fold model applied to the CICIDS2017 dataset.

Figure 8 shows the performance metrics of the SEL + K-Fold model applied to the NSL-KDD dataset for each fold. Across the five different folds, all metrics range between 99.55% and 99.70%. This indicates that the model demonstrates consistent and high performance in each fold. Notably, the third fold achieves the highest performance, with 99.70% across all metrics. The K-Fold method has enhanced the model's generalization capacity, making the results more reliable. The consistently high metrics highlight the model's success in both detecting attacks and classifying normal traffic, underscoring its effectiveness.

Figure 9 shows the success metrics (ACC, Precision, Recall, F1 Score) of the SEL + K-Fold model applied to the CICIDS2017 dataset for each fold. In the tests performed on five different folds, all metrics vary between 99.43% and 99.60%. Especially in the first three folds, the success rate reaches 99.57%. However, it is observed that these values drop to around 99.43% in the 4th and 5th folds. This shows that the SEL + K-Fold model generally exhibits a very high performance, but there may be slight variations between different folds. The fact that the model gives such high and consistent results reveals that it is effective and reliable on the CICIDS2017 dataset. In the study, the integration of multiple ML algorithms (DT, XGBoost and MPL) and K-Fold cross validation into the SEL model represents a significant advance in cyber attack detection methods. In addition, the success of the

system is evaluated using popular data sets such as CICIDS2017 and NSL-KDD.

Table 6 shows that the K-Fold cross validation applied together with the SEL model increases the performance in both data sets. The ACC value increased from 0.9953 to 0.9965 in the NSL-KDD data set, and from 0.9943 to 0.9950 in the CICIDS-2017 data set. This situation reveals that K-Fold cross validation increases the overall ACC and reliability of the proposed model. It also enables the model to better learn the general structure and patterns in the data and increase the reliability of its predictions. This is a finding that supports the model to be more robust and effective in practical applications.

Table 6: Comparative success metrics of models.

	NSL-KDD		CICIDS-2017	
	SEL Model	**SEL Model + K-Fold**	**SEL Model**	**SEL Model + K-Fold**
Accuracy	0.9953	0.9965	0.9943	0.9950
Precision	0.9952	0.9965	0.9943	0.9950
Recall	0.9953	0.9965	0.9943	0.9950
F1 Score	0.9952	0.9965	0.9943	0.9950

Table 7: Comparative time values of models.

	NSL-KDD		CICIDS-2017	
	SEL Model	**SEL Model + K-Fold**	**SEL Model**	**SEL Model + K-Fold**
Training Time	394.7138	5149.4919	65.6485	297.3936
Prediction Time	0.8539	0.5381	0.0166	0.0172

In Table 7, the time values for training and prediction for the models are presented. While K-Fold CV enhances the model's generalization ability, it significantly increases the training time. This is because cross-validation requires retraining the model for each fold, which greatly impacts the length of the training process. However, the prediction time remains generally short and nearly unchanged, as it operates independently of the training process once the model is trained. This minimal variation in prediction time highlights that K-Fold CV, while offering more accurate and reliable results, comes with the trade-off of increased training time.

In Table 8, it is clearly evident that the proposed model outperforms existing methods. These results emphasize the model's ability to accurately and effectively address various types of attacks. Particularly, the critical role of K-Fold CV in enhancing the success of SEL models in attack detection tasks is highlighted. These findings demonstrate that K-Fold CV strengthens the model's generalization capability and optimizes its performance. Thus, our research

makes a significant contribution to advancing cyber security by strengthening and enhancing defense mechanisms against the ever-evolving cyber threats.

Table 8: Comparing the performance of the proposed model with related works.

Ref.	Year	Dataset	ML Techniques or stacked	Best Accuracy
[34]	2020	NSL-KDD	KNN, MLP, RF, NB, LR, SVM	99.47
[35]	2020	NSL-KDD	XGBoost, MCC, AUC	97.00
[36]	2022	UNSW-NB15, NSL-KDD	Stacked	95.26
[37]	2024	CICIDS2017, NSL-KDD	Stacked	97.83
[38]	2020	CICIDS2017, NSL-KDD	AE+DNN	98.43
[39]	2024	NSL-KDD, N-BaIoT, CICIDS2017	hybrid (Stacked)	97.00
[40]	2023	UNSW-NB15, NSL-KDD	LR, KNN, RF, DT, MLP	98.60
[41]	2022	UNSW-NB15, NSL-KDD	EMBAM (Stacked)	99.50
Proposed Model		NSL-KDD, CICIDS2017	SEL + K-Fold	**99.65**

5. CONCLUSION AND SUGGESTIONS

Traditional intrusion detection systems that are not integrated with ML methods are insufficient. This study presents an innovative method for detecting cyber attacks. The proposed method includes a multiple SEL system. DT, XGBoost and MLP are used in the basic model section of the multiple SEL system, and LR is used in the meta model section. In addition, K-Fold CV is used to prevent over-learning of the model and to increase generalization ability. As a result of the analysis performed on NSL-KDD and CICIDS2017 data sets, 99.65% and 99.50% success rates are achieved, respectively. This innovative method, when compared to traditional methods, reveals how successful the model is in detecting different types of attacks.

The results indicate that the use of K-Fold CV alongside the SEL structure improves the model's ACC. This method has proven particularly effective in scenarios where individual models struggle to achieve high performance. Overall, the proposed SEL + K-Fold model demonstrates clear superiority over existing methods in terms of ACC, detection rates, and computational efficiency, greatly enhancing the ability to detect cyberattacks.

This study aims to enhance Cyberattack detection systems using a robust SEL classification approach, with a focus on generalization and scalability to foster future innovations in cyber security and advance threat detection and network security.

Future Recommendations:

- ✓ Real-Time Data Usage: The model can be integrated with real-time data to enhance its dynamic threat detection capabilities.
- ✓ Advanced Feature Engineering: Techniques such as feature selection and dimensionality reduction can be employed to enhance the model's efficiency.
- ✓ Adaptation to New Threats: Adapting the model to new types of cyberattacks is essential to maintain its effectiveness.

ACKNOWLEDGMENTS

This research received no external funding.

AUTHOR CONTRIBUTIONS

Faruk AYATA: Conceptualization, Data curation, Formal analysis, Investigation, Methodology, Project administration, Resources, Software, Supervision, Validation, Visualization, Writing – original draft, Writing – review & editing, Experimental studies.

CONFLICTS OF INTEREST

The author declares that there is no conflict of interests regarding the publication of this manuscript.

REFERENCES

[1] Al-Garadi MA, Mohamed A, Al-Ali A, et al. Nesnelerin İnterneti (IoT) güvenliği için makine ve derin öğrenme yöntemlerine ilişkin bir araştırma. arXiv.org, cilt. arXiv:1807.11023, 2018, s.1–42.

[2] Dasgupta, D. et al. ML in cybersecurity: a comprehensive survey, The Journal of Defense Modeling and Simulation, 2022, Vol. 19, No. 1, pp. 57-106.

[3] Tong, W. et al. A survey on intrusion detection system for advanced metering infrastructure, In: Sixth international conference on instrumentation & measurement, computer, communication and control (IMCCC), IEEE, Harbin, China, July 2016, pp. 33-37.

[4] Altın O. AB'nin Siber Güvenlik Alanındaki Politikalarının ve Uygulamalarının Etkinliği: Bir Siber Güvenlik Temsilcisi Olarak AB'nin Yeterliliği. Çankırı Karatekin Üniversitesi İktisadi ve İdari Bilimler Fakültesi Dergisi. 2023, 13(2):482-507.

[5] Willard, GN. Understanding the Co-Evolution of Cyber Defenses and Attacks to Achieve Enhanced Cybersecurity. Journal of Information Warfare, vol. 2015, 14, no. 2, pp. 16–30. JSTOR.

[6] Neciyev, S. and Pazarbaşı, B. Siber Güvenlik, Siber Savaş Alanında Seçili Anahtar Kelimeler ile İlgili Araştırmaların Bibliyometrik Analizi. Gazi Üniversitesi Fen Bilimleri Dergisi Part C: Tasarım ve Teknoloji. 2023.

[7] Alaca, Y., Celık, Y., & Goel, S. Anomaly Detection in Cyber Security with Graph-Based LSTM in Log Analysis. Chaos Theory and Applications. 2023, 5(3), 188-197.

[8] Yeniman Yıldırım, E. Bilişim Sistemlerine Yönelik Siber Saldırılar ve Siber

Güvenliğin Sağlanması. Mesleki Bilimler Dergisi (MBD). 2018: 7(2), 24-33.

[9] Halbouni, A. et al.: ML and Deep Learning Approaches for CyberSecuriy: A Review, IEEE Access, 2022, Vol. 10, pp. 19572-19585.

[10] Abdullahi, M. et al.: Detecting cyber security attacks in the internet of things using AI methods: A systematic literature review. Electronics, 2022, Vol. 11, No. 2, 198.

[11] Salih, A. A. and Abdulrazaq, M. B. Combining Best Features Selection Using Three Classifiers in Intrusion Detection System, 2019 International Conference on Advanced Science and Engineering (ICOASE), Zakho - Duhok, Iraq, 2019, pp. 94-99.

[12] Tokmak, M. Öğrencilerin Siber Güvenlik Farkındalık Düzeylerinin Makine Öğrenmesi Yöntemleri ile Belirlenmesi. Yüzüncü Yıl Üniversitesi Fen Bilimleri Enstitüsü Dergisi. 2023, 28(2), 451-466. https://doi.org/10.53433/yyufbed.1181694.

[13] İlgün E, Samet R. Veri setine uygulanan ön işlemler ile makine öğrenimi yöntemi kullanılarak geliştirilen saldırı tespit modellerinin performanslarının arttırılması. GUMMFD. 2023, 39(2):679-92.

[14] Tuğrul B, Ahmed ASA. Makine öğrenme yöntemleri ile ağ trafik analizi. NÖHÜ Mühendislik Bilimleri Dergisi. 2022, 11(4):862-70.

[15] Dovbysh, A, Liubchak, V, Shelehov, I, Simonovskiy, J, & Tenytska, A. Information-extreme ML of a cyber attack detection system. Radıoelectronıc and Computer Systems. 2022, 3, 121–131. doi: https://doi.org/10.32620/reks.2022.3.09.

[16] Albakri A, Alabdullah B, Alhayan F. Blockchain-assisted ML with hybrid metaheuristics-empowered cyber attack detection and classification model. 2023, Sustainability 15:13887. https://doi.org/10.3390/su151813887.

[17] Reddy, D, Sajjan, B.T, & Sadiq, S.J. Detection of Cyber Attack in Network using ML Techniques. 2021, Vol. 1 No. 2.

[18] Lu, M., Hou, Q., Qin, S., Zhou, L., Hua, D., Wang, X., & Cheng, L. (2023). A stacking ensemble model of various ML models for daily runoff forecasting. Water, 15(7), 1265.

[19] Jaiyeoba, O., Ogbuju, E., Yomi, O. T., & Oladipo, F. (2024). Development of a model to classify skin diseases using stacking ensemble ML techniques. Journal of Computing Theories and Applications, 2(1), 23–37. https://doi.org/10.62411/jcta.10488

[20] Ghasemieh, A., Lloyed, A., Bahrami, P., Vajar, P., & Kashef, R. (2023). A novel ML model with stacking ensemble learner for predicting emergency readmission of heart-disease patients. Decision Analytics Journal, 7, 100242. https://doi.org/10.1016/j.dajour.2023.100242

[21] Zhu, X., Hu, J., Xiao, T., Huang, S., Wen, Y., & Shang, D. (2022). An interpretable stacking ensemble learning framework based on multi-dimensional data for real-time prediction of drug concentration: The example of olanzapine. Frontiers in Pharmacology, 13, 975855. https://doi.org/10.3389/fphar.2022.975855.

[22] Fernandes, E., Holanda, M., Victorino, M., Borges, V., Carvalho, R., & Van Erven, T. (2019). A Stacking-Based Ensemble Learning Approach for Predicting Student Performance. IEEE Access, 7, 104746-104758. https://doi.org/10.1109/ACCESS.2019.2931811

[23] Tavallaee, M. et al: A detailed analysis of the KDD CUP 99 data set, 2009 IEEE Symposium on Computational Intelligence for Security and Defense Applications, Ottawa, ON, Canada, 2009, pp. 1-6

[24] Hafid, H. (2023). Penerapan K-Fold Cross Validation untuk Menganalisis Kinerja Algoritma K-Nearest Neighbor pada Data Kasus Covid-19 di Indonesia. Journal of Mathematics, 6(2), 161-168.

[25] Wu, W., Xia, Y., & Jin, W. (2020). Predicting bus passenger flow and prioritizing influential factors using multi-source data: Scaled stacking gradient boosting DTs. IEEE Transactions on Intelligent Transportation Systems, 22(4), 2510-2523.

[26] Li, Q., & Song, Z. (2023). Prediction of compressive strength of rice husk ash concrete based on stacking ensemble learning model. Journal of Cleaner Production, 382, 135279.

[27] Ali, J, Khan, R, Ahmad, N, & Maqsood, I. Random forests and DTs. International Journal of Computer Science Issues (IJCSI). 2012, 9(5), 272.

[28] Chen, T., & Guestrin, C. Xgboost: A scalable tree boosting system. In Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining. 2016, pp. 785-794.

[29] Pervaiz, M., Jalal, A., & Kim, K. (2021, January). Hybrid algorithm for multi people counting and tracking for smart surveillance. In 2021 International Bhurban conference on applied sciences and technologies (IBCAST) (pp. 530-535). IEEE.

[30] Naseer, A., & Jalal, A. (2024, February). Multimodal Objects Categorization by Fusing GMM and MLP. In 2024 5th International Conference on Advancements in Computational Sciences (ICACS) (pp. 1-7). IEEE.

[31] Arif, A., & Jalal, A. (2021, January). Automated body parts estimation and detection using salient maps and Gaussian matrix model. In 2021 International Bhurban Conference on Applied Sciences and Technologies (IBCAST) (pp. 667-672). IEEE.

[32] Shuja, A., & Jalal, A. (2023). Vehicle detection and tracking from Aerial imagery via YOLO and centroid tracking. IEEE ICACS, 151, 183-195.

[33] Orozco-Arias, S, Piña, J. S, Tabares-Soto, R, Castillo-Ossa, L. F, Guyot, R, & Isaza, G. Measuring performance metrics of ML algorithms for detecting and classifying transposable elements. Processes. 2020, 8(6), 638.

[34] Abrar, I, Ayub, Z, Masoodi, F, & Bamhdi, A. M. A ML approach for intrusion detection system on NSL-KDD dataset. In 2020 international conference on smart electronics and communication (ICOSEC). IEEE .2020, pp. 919-924.

[35] Liu, J, Kantarci, B, & Adams, C. ML-driven intrusion detection for Contiki-NG-based

IoT networks exposed to NSL-KDD dataset. In Proceedings of the 2nd ACM workshop on wireless security and ML. 2020, pp. 25-30.

[36] Rashid, M., Kamruzzaman, J., Imam, T., Wibowo, S., & Gordon, S. (2022). A tree-based stacking ensemble technique with feature selection for network intrusion detection. Applied Intelligence, 52(9), 9768-9781.

[37] Sneha, S., Roshni, A., & Padmavathi, G. (2024). A Stacked Ensemble Model to Detect Network Intrusions. Grenze International Journal of Engineering & Technology (GIJET), 10.

[38] Kasim, Ö.: An efficient and robust deep learning based network anomaly detection against distributed denial of service attacks. Computer Networks, 2020, Vol. 180, pp. 107390

[39] Jemili, F., Meddeb, R., & Korbaa, O. (2024). Intrusion detection based on ensemble learning for big data classification. Cluster Computing, 27(3), 3771-3798.

[40] Fuat, T. Analysis of intrusion detection systems in UNSW-NB15 and NSL-KDD datasets with ML algorithms. Bitlis Eren Üniversitesi Fen Bilimleri Dergisi. 2023, 12(2), 465-477.

[41] Alhabshy, A. A., Hameed, B. I., & Eldahshan, K. A. (2022). An ameliorated multiattack network anomaly detection in distributed big data system-based enhanced stacking multiple binary classifiers. IEEE Access, 10, 52724-52743.