LEX LOCALIS-JOURNAL OF LOCAL SELF-GOVERNMENT
ISSN:1581-5374 E-ISSN:1855-363X
VOL. 23, NO. S5 (2025)

HYBRID INTRUSION DETECTION WITH DEEP FEATURE EXTRACTION AND ML CLASSIFICATION

Madini O. Alassafi[1]

[1]Department of Information Technology, Faculty of Computing and Information Technology, King Abdulaziz University, Jeddah, Saudi Arabia,

malasafi@kau.edu.sa[1]

Abstract: With the advancement of the digital network, there has been a constant increase in the number of instances of intrusions. In today's world of cybersecurity, intrusion detection is alarmingly of great concern due to the serious consequences it can present. In the past few years, diverse "Machine Learning" (ML) and "Deep Learning" (DL) approaches have been employed for intrusion detection. However, regarding accuracy, both types of learning techniques have their limitations. A combined approach, however, presents a situation whereby much more research has to be conducted in order to determine its effectiveness in the detection of intrusions. In this work, a model was proposed that incorporates DL feature extraction with ML classification. It is, therefore, with high precision that this method recorded an astounding 96 percent accuracy in detecting intrusion. This paper has provided evidence that the model proposed in this paper outperforms the previous models, including DL and ML, in intrusion detection. These are likely to provide significant contributions to the improvement of cybersecurity.

Keywords: intrusion detection, Machine Learning, Deep Learning, feature extraction, classification, dataset, accuracy

1 Introduction

The increasing trend of cyber intrusions calls for the necessity of having good and efficient intrusion detection systems. As the number and complexity of intrusions rise, there is a growing requirement to identify and counteract them properly. To overcome this challenge, many researchers have used ML and DL techniques to identify intrusion detection.

Although Ml and Dl techniques have greatly improved, the current approaches have shortcomings, which stem from the bias to attain high accuracy. This means that new methods are needed to find the best approach that will enable the integration of the two in intrusion detection.

This paper proposes a comprehensive model to employ DL techniques for feature extraction and then ML methods for classification. This work combines the profound learning ability to learn complex patterns with the ML ability to perform classification to propose a general framework for intrusion detection that is not confined to specific methods.

The primary contributions of our research can be highlighted as follows:

Advanced Hybrid Model: This work proposes a new model architecture that integrates DL to extract features and ML for classification so that all the detection abilities can be placed in one model.

Model Performance: The proposed model demonstrates its effectiveness by achieving a remarkable accuracy rate of 96% in detecting intrusions. This is even greater than the latest milestones in both the existing DL and ML methods.

Strengthened Cybersecurity Posture: The results support the usefulness of the hybrid approach to significantly improving cyber security defences by detecting and neutralizing intrusions.

The rest of the paper is organised as follows: In Section 2, the current literature and prior works related to intrusion detection and cybersecurity are reviewed. In Section 3, the dataset employed

in the current study is presented, and the methodology of the proposed intrusion detection is outlined with regards to feature selection, model architecture, training, and assessment criteria. The experimental results, performance indicators, and comparisons obtained during the study are presented in Section 4. Last, Section 5 offers a conclusion and presents the main findings of the paper, the implications of the findings for the field of cybersecurity, and possible future research directions.

2 Literature Review

In today's world, network security Intrusion detection systems (IDS) are vital in identifying threats that could be posed to any organisation. IDS are mainly based on two popular methods: these are the signature-based and anomaly-based detection. The signature-based IDS depends on predefined signatures to quickly identify known threats, while anomaly-based IDS utilises specific patterns to distinguish between normal and abnormal system behaviour. ML-inspired techniques are widely incorporated for their efficacy, which becomes integral to intrusion detection, providing a sophisticated and extensively utilised approach to recognising and mitigating security risks. The comprehensive analysis of relevant literature is presented next.

Nagaraja et al. [1] introduced the concept of membership function for precise detection of low-frequency data breaches, particularly R2L and U2R attacks. Their method includes a new similarity measure for high-dimensional data, which is very effective in reducing noise and accurately identifying intrusions after the mandatory data reduction and transformation process.

Zoppi et al. [2] focused on building an IDS using unsupervised anomaly detection algorithms for zero-day attack detection. To find low classification errors, the authors performed experiments with different algorithms, such as Self-Organizing Maps and Isolation Forests. However, they advocate for clustering algorithms due to performance constraints, emphasising their stability in detecting diverse attacks, including malware and spam.

Li et al. [3] addressed the dynamic network access control challenge by developing an anomaly-based IDS for Software-Defined Networking (SDN). The proposed technique incorporates a recurrent neural networks (RNNs) architecture, which supports the proposed IDS in dynamically acquiring network anomaly knowledge for creating access control policies.

Zoppi et al. [4] outlined some essential design aspects for improving SoS architecture and provided guidelines for observing anomaly detection frameworks. The authors, as expected, defined SoS challenges on top of anomaly detection solutions, in line with the manoeuvrability of anomaly-based algorithms into managing assorted components that could be third-party possessed or off-the-shelf in SoS.

Pacheco et al. [5] proposed an IDS using anomaly behaviour analysis (ABA-IDS) to safeguard IoT nodes from cyber hreats. The ABA-IDS consists of honeypot and Markov models for identifying anomalies and compromised machines accurately. This also provides protection and assurances of the IoT nodes within a smart structure environment.

Sonmez et al. [6] focused on the comprehensive analysis of phishing website classification by employing the tool known as Extreme Learning Machine (ELM). The performance was classified by the authors by employing a broad range of categorisation techniques, including naïve Bayes, SVM, and artificial neural networks. In this paper, six activation functions brought the overall accurate percentage of the ELM approach to 95.34%.

Ram et al. [7] proposed a ML approach to the detection of phishing incidences that requires the use of 16 predictors and six models. The lowest error rate in all experiments is that of the SVM, while a biased SVM and artificial neural networks both have an error rate of 2.02%.
Kamal et al. [8] suggested that ML and URL-based features should be used for the identification of phishing as reported by the increased number of such attacks in the year 2014. Using the Naïve Bayes algorithm on the Weka platform with ensemble methods incorporating Decision Tree and Random Forest, the classification accuracy was 97.08%.
Priyanka et al. [9] suggested using phishing detection using ML feature extraction, using Adaline and backpropagation algorithms with SVM. Adaline learnt to elucidate better performance with a 99.14% detection rate as compared to SVM with less processing time as compared to the backpropagation network.
Kaytan Mustafa et al. [10] proposed an ELM classification for identifying the phishing websites based on features such as "Request URL" and "Website Forwarding." The classification merged an average accuracy of 95.05% and the highest accuracy of 95.93% using 10-fold cross-validation for assessment.
Weina Niu et al. [11] presented the Cuckoo-Scan SVM (CS-SVM) model for implementing an effective email phishing detection with high classification accuracy. In the present study, CS-SVM was used to select 23 features that were combined with Cuckoo Search (CS) and SVM. This paper used the hybrid classifier with Radial Basis Function (RBF) parameter tuning to generate 99.52% accuracy, which was better than the SVM Classifier.
Ishant et al. [12] suggested different approaches of ML used in the URL features for the detection of phishing. To accomplish this, the authors identified thirty features of phishing sites using Python and then used GLM and GAM for the analysis of the data. Improve the result: the random forest was used, and the accuracy was 98.4%.
Jain et al. [13] presented an anti-phishing framework based on URL features for a ML classification model. To distinguish a phishing site from the regular one, the framework employs fourteen URL features. In the experiments, the system was trained with 33,000 phishing and valid URLs using SVM and Naïve Bayes classifiers; the accuracy of identifying phishing websites was 90% using the SVM classifier.
Pujara et al. [14] addressed phishing fraud, a prevalent cybercrime, by conducting a literature survey and proposing a novel detection approach integrating feature extraction and ML algorithms. The paper explores various methodologies, including the Blacklist method, heuristic method, visual similarity, and ML. The Blacklist method involves storing a list of phishing URLs in a database, flagging URLs in the database as phishing and triggering alerts, while unrecognised URLs are considered legitimate. The heuristic approach extends the Blacklist method, detecting new attacks by extracting features from phishing sites. The visual similarity strategy deceives users by removing images from valid websites. The ML approach proves effective in handling large datasets.
Rathore et al. [15] conducted a comprehensive survey of security and privacy risks targeting social networking site (SNS) users. Social Network Services (SNS) create virtual connections between individuals with similar interests. As the number of SNS users worldwide steadily grows, the paper focuses on various risks associated with multimedia content sharing within social networking platforms. The identified threat categories include risks to multimedia content, conventional threats, and social threats.

Zaman and Karray [16] Proposes a novel lightweight Intrusion Detection System (IDS) featuring a Fuzzy ESVDF-based feature selection method and an IDS classification scheme (application layer, transport layer, network layer, and link layer). This innovative approach significantly improves system efficiency, addressing the limitations of traditional IDS reliant on predefined attack signatures in the context of modern cybersecurity challenges.

In their work, Louvieris et al. [17] presented a narrative anomaly detection approach that employs k-means clustering, Naive Bayes feature extraction, and C4.5 decision tree classification to identify previously unrecognised intrusion attempts. This impactful feature identification algorithm enhances cyber network operators' depth perception and exhibits high precision in locating cyber-attacks.

Alotaibi and Eileithyia [18] introduced a novel wireless intrusion detection system voting technique to establish a rogue Wireless Local Area Network Intrusion Detection System (WIDS) and identify critical areas in WLAN MAC-layer frames. Their evaluation identifies Extra Trees, Random Forests, and Bagging as top-performing algorithms. Bagging and the proposed technique achieve enhanced efficiency through a customised voting approach, with accuracy levels reaching 96.25 % and 96.32 %, respectively. Utilising data-mining techniques, the study employs the Additional Trees ensemble method to pinpoint the most influential features in the Aegean WiFi Intrusion Dataset (AWID). The work has identified that Extra Trees and the suggested voting method are the best classifiers by accuracy using the twenty most important characteristics.

As pointed out in [19], a proposed independently developed normalised gain-based IDS for MAC intrusions may help in enhancing IDS performance. The NMI included two crucial components: OFSNP and DCMI, employing the methodology for MAC 802.11 intrusion detection and categorisation based on an SVM classifier. Feature ranking in OFSNP is done differently through normalised gain (NG) and selection of the best features from the semi-supervised clustering (SSC) involving particle swarm optimisation (PSO). For proper classification of attacks, DCMI uses the selected features with the SVM learning method. In assessment, the NMI achieved equal accuracy rate and learning time needed.

Palmieri et al. [20] proposed an adaptive anomaly detection model using two ML models incorporating BSS strategies with rule-based classifiers. This framework was envisaged to help identify 0-day attacks that are borne out of changes in traffic volumes and flow dynamics. The method is favourable against new and relatively unknown threats by being conscious of suspicious-looking network onnections. These results demonstrated our ability to withstand evasion, and the work also exhibits promising performance in separating suspicious traffic from ordinary network traffic. The findings were analysed using real traffic data tests, and it also looks fairly effective at the same time as setting good detection rates with a low false positive rate.

Mehrotra et al. [21] proposed a modification to the decision tree ID3 algorithm for better attribute selection by assigning higher weight values to the important attributes. The authors explored techniques of classification, clustering, association rule mining, and sequential pattern mining on the "KDD99" dataset with five attributes per packet. The results show higher accuracy through the modified ID3 (95%) compared to the standard ID3 (92%).

Ashoor et al. [22] delineated the functionalities of IDS and intrusion prevention systems (IPS) during an attack, emphasising the role of signature-based traffic analysis. They compared IDS and IPS across stability, performance, and accuracy metrics, focusing on network stability, accuracy (false positives and negatives), and data log analysis. Their analysis highlighted the distinct

objectives of IDS and IPS in tracking and preventing network attacks through practical log analysis.

The work presented in [23] proposed an IDS including a honeypot with Real real-time rule Accession (RTRA). The authors used the Apriori method for association rule mining on log files to construct Snort IDS rules for effective intrusion detection. The proposed approach has proven very effective in identifying new attacks.

Mishra et al. [24] highlighted the growing popularity of cloud computing and the significance of security within this domain. They introduce Snort as an intrusion detection system, mainly designed for rule-based detection to analyse and detect various attacks. The Snort architecture includes a packet sniffer, preprocessor, detection engine, logging and alerting system, and rules. Their conclusion highlights Snort's efficacy in addressing security concerns within the cloud computing environment.

In Brahmi et al. [25], authors addressed the challenge of detecting distributed attacks and proposed a novel distributed IDS named MAD-IDS (Mobile Agent using Data mining-based Intrusion Detection System). This system employs mobile agents in collaboration with data mining techniques to identify both known and unknown attacks. The distributed system's Sniffer, Filter, Misuse, Anomaly, Rule Mining, and Reporter Agents may move between stations. Their experiment, conducted with DARPA traffic data, concludes that their approach is effective for detecting attacks in distributed systems.

Wang [26] compared DL algorithms for intrusion detection to determine the weakness of neural networks under attack and analysed the role of features in creating attack instances. The evaluation utilises DL models, particularly the multilayer perceptron (MLP), and then attacks the models with FGSM, JSMA, DEEPFOOL, and CW Attack.

In their research work, Zhong et al. [27] developed a big data-based hierarchical DL system (BDHLS) to improve the IDS's effectiveness. The dataset is categorised into several levels, and a DL model is developed for each level. The decision values of these models are then aggregated to make a final decision as to the type of attack. The system uses both the behavioural and content analysis of the attack samples. Nevertheless, the computationally demanding nature of the method is a drawback in order to obtain the most optimal results.

Khan et al. [28] suggested a two-stage neural network model (TSDL) for detection of intrusion that employs a soft max classifier with stacking denoising automatic encoder in the first stage. The first stage has a simple binary decision of whether the traffic is normal or abnormal, and the values obtained in the first stage are used as an extra feature in the second stage to make the final decision on the network traffic. The proposed model achieves a high true detection rate, with accuracies of 99.996% on the KDD99 and 89.134% on the UNSW-NB15 datasets.

Al-Qatf et al. [29] presented another DL model, namely Self-Taught Learning (STL)-IDS with SVM. The model is built based on sparse autoencoders. In comparison with other algorithms, the proposed model gives higher accuracy and lower training and testing time on the NSL-KDD dataset.

Yen et al. [30] proposed an RNN-based IDS known as RNNIDS for intrusion detection. The performance was compared with binary and multiclass classification with different state-of-the-art ML algorithms. The proposed model achieved a higher intrusion detection capability and improved accuracy than other approaches in the NSL-KDD dataset.

In [31], Xu et al. used a multilayer perceptron neural network with a SoftMax layer and one hidden layer of gated recurrent units for intrusion detection. Based on the KDD 99 and NSL-KDD datasets, the GRU outperforms the LSTM in intrusion detection.
Vijayanand et al. [32] outlined a hierarchical DL-based system for analysing network traffic in smart electric meters. This system prioritises algorithms based on the severity of attacks, with each layer determining if an attack belongs to its class or passes it to the next layer. Experiments show that the proposed system accurately detects attacks and performs well on the CICIDS 2017 dataset.
Wasi et al. [33] suggested an intrusion detection method consisting of statistical analysis and DL approaches. The approach involves a backpropagation cross-entropy artificial neural network algorithm parallel with statistical analysis. The proposed model demonstrates the simplicity and high detection accuracy of 99.23% on the KDDCup99 dataset.
Chockwanich and Visoottiviseth [34] suggested a DL-based intrusion detection system that operates without human intervention, avoiding the need for signatures and preset rules. Three DL models (RNN, Stacked RNN, and CNN) are implemented. Evaluation of the MAWILab'2017 dataset shows that all models outperform Snort IDS in detecting attacks.
Zhao et al. [35] proposed a DL approach to convert the raw data into low-dimensional data, constructing a model using a probabilistic neural network (PNN). The number of hidden layers is optimised using the particle swarm optimisation (PSO) algorithm. Experiments on the KDD-CUP99 dataset demonstrate that the combination of (PSO) and (PNN) addresses challenges intrusion detection researchers face, such as large amounts of data and long training times.

3 Material and Methods

3.1 Dataset

The aim of the model uses the image of the SIDD dataset. The Segmented Intrusion Detection Dataset (SIDD) or the Large Scale Network Intrusion Image Dataset is the first intrusion detection system that was introduced by Sun et al. [36]. It has a collection of image-based network traffic samples designed explicitly for intrusion detection applications.
However, the SIDD dataset is unique in many ways with its uniqueness of design and scope. It contains images created by network traffic protocol communication over 15 geographical locations within different Asian countries. Each photographic image in the dataset, measuring 48×48, is a time frame in minutes of 128, showing the overtime of the several communicating protocols [26]. This demonstrates that the cover of the network traffic is also rich in detail to the level of understanding the nature of threats in the intrusions.
The SIDD dataset's versatility allows it to be employed for various tasks. In particular, it is helpful for network intrusion detection in ML approaches and federated learning [37]. The elaborate image data set based on knowledge of the SIDD dataset can very well address the problems of these research areas.
Moreover, regarding the data set, one can freely employ their imagination in enhancing the current methods of IDS. It becomes possible in order to seek fresh ideas to solve problems in the connected subject areas. In this context, the SIDD dataset can be further valuable for the development of innovative concepts and ideas but also for the refinement of already used concepts and ideas. In conclusion, the Segmented Intrusion Detection Dataset offered is a major improvement in the IDS research field. Due to its unique approaches to construction, usability, and variety of information, it is a highly valuable tool for scientists and professionals. Further, it supports the detection of

intrusions in a network and the improvement of the resolution in this area of computer security as well.

3.2 Preprocessing

First, the dataset is preprocessed for ML and DL by loading images from two subdirectories into one subdirectory in the most convenient way using the built-in function. Every picture is tagged with a normal score of 0 or an attack score of 1. Also, it underlines image augments such as rotations, scaling in both width and height, and flippings in terms of width and height. The dataset split into a ratio of 80:20. Feature selection is then performed using the chi-squared statistical test through the SelectKBest method from scikit-learn. A specific number of top features (k=96) are chosen based on their relevance to the target variable. The images are flattened into one-dimensional arrays, and feature selection is applied to retain only the desired features.

3.3 Propose modelling

The proposed model uses Convolutional Neural Networks (CNNs) for feature extraction and employs ML techniques for classification, as illustrated in Figure 1. Different ML techniques, including LightGBM (Light Gradient Boosting Machine), Random Forest (RF), Decision Trees (DT), SVM, K-Nearest Neighbors (KNN), and Multi-Layer Perceptron (MLP), have been selected based on their respective capabilities in classification.

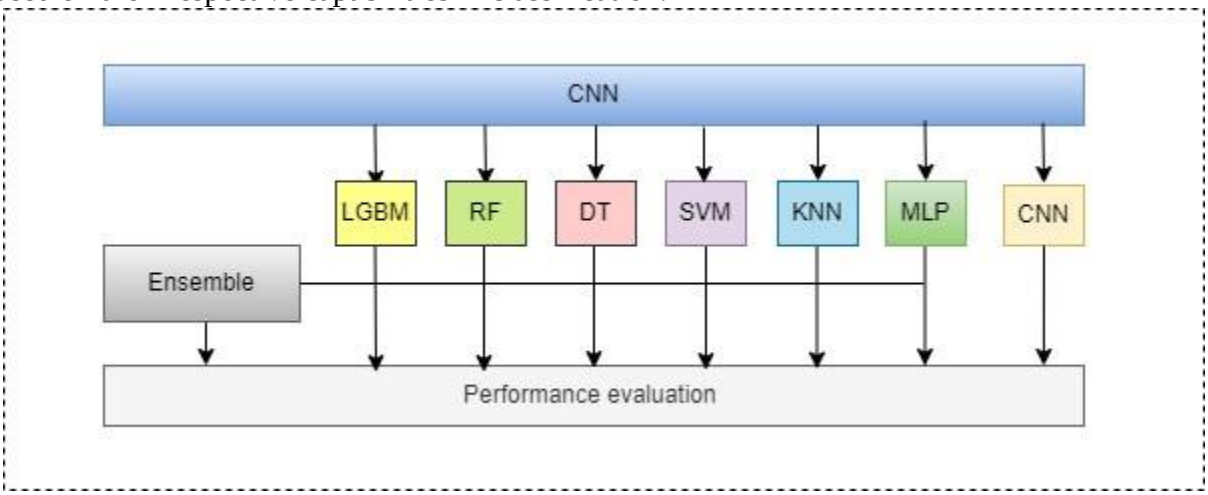

Figure 1: Propose model based on deep and ML

3.4 Implementation

The proposed model was implemented on an Intel(R) Core(TM) i7-6500U CPU @ 2.50GHz, equipped with two cores and four logical processors. TensorFlow, in conjunction with essential Python libraries, was employed for implementation. Numerical operations were executed using NumPy, while visualization tasks were carried out using matplotlib. pyplot for general plotting and matplotlib. Image for image visualization. Additionally, Seaborn was utilized to enhance data visualization. Neural network architecture was constructed using Keras, incorporating optimizers for network optimization [38]-[47].

a. Implementation phase 1 starts with implementing several independent ML models. The LightGBM classifier model, renowned for its efficiency and scalability, was initially instantiated and prepared for training on the preprocessed and feature-selected dataset using Keras' built-in 'fit' method. Then, the Random Forest classifier was presented as 'rf' due to its ability as an ensemble model and its ability to work with complex data relationships. The SVM implemented as 'self' using the built-in Keras model API and the K-nearest neighbours (KNN) model, which classify

instances based on the majority class of the cases within a given proximity, were instantiated as 'kin'. Moreover, a Multilayer Perceptron classifier was designed and instantiated as a 'map' using the Keras model API.
b. In the last sub-phase, 1, an ensemble model was strategically introduced to improve the overall predictive power of the other models mentioned. This study employs the VotingClassifier from the scikit-learn libraries and uses the 'soft' voting strategy, which means that the predicted probabilities of each classifier are averaged to increase the model's ability to identify different patterns within the dataset.
C. phase 2 introduces a hybrid pipelining approach integrating a pre-trained VGG16 model for feature extraction, followed by feature selection and subsequent classification using ML models. The pre-trained VGG16 model is loaded and configured to exclude its top layers responsible for classification, serving solely as a feature extractor. An intermediate model is defined as one that takes inputs from the base model and produces the features extracted. The feature extractor extracts and selects optimal features using the chi-squared test, employing the SelectKBest class from the sklearn feature_selection module.
d. Finally, each ML model is called to compare the hybrid model's performance against the ensemble models during training and assessment.

3.5 Training and Testing

The training and testing phase involves constructing and evaluating a hybrid image classification model. This model combines Convolutional Neural Network (CNN) feature extraction with various traditional ML classifiers.
First, we load and preprocess the images. Images from two classes (regular and attack) are loaded and resized to a uniform size of 64x64 pixels. They are then converted to a numpy array. The model is evaluated on unobserved data by dividing the data set into train and test.
Second, to capture features from the images, we use a pre-trained VGG16 model. Using DL and transfer learning, we can get features without having to train a network from scratch. We do not include the last layer of the VGG16 model in our study.
To address the issue of the CNN features' high dimensionality, we resort to the 'chi2' feature selection. This approach chooses variables for classification based on the chi-squared statistic to determine which variables are most relevant. It simply keeps only the top k features of the data.
We proceed to train the following classifiers based on their capabilities in classification, using the selected features:

1. Light Gradient Boosting Machine (LGBM): LGBM is an algorithm based on the gradient boosting methodology suitable for big and high-dimensional data and computationally efficient.
2. Random Forest (RF): Multiple decision trees in a supervised learning system reduce overtraining and keep it stable.
3. Decision Tree (DT): The decision tree model makes decisions using a graph or a tree-like structure. It is simple, effective, and quick to comprehend.
4. SVM: A supervised learning algorithm that performs classification by finding a hyperplane works excellently in high-dimensional space, where the high dimensionality tends to exceed the number of samples.
5. K-Nearest Neighbors (KNN): A classification and regression type of a non-parametric ML algorithm where the most popular class is chosen among k-closest instances.

6. Multilayer Perceptron (MLP): This is a feedforward class of artificial neural networks that can learn non-linear models.

3.6 Performance evaluation

Classification accuracy, precision, recall, and f1 score quantify model performance in categorising normal pictures versus attack images. In contrast, the Confusion Matrix, for example, contains information on how well the model performed in classifying the images, measuring the number of correct classifications and the types of errors the model made.

Accuracy = (TN+TP)/ (TP+FP+TN+FN)

Where:

- True Positives (TP): Accurate prediction of attack
- False Positives (FP): Inaccurate prediction of attack.
- True Negatives (TN): Accurate prediction of normal.
- False Negatives (FN): Inaccurate prediction of normal.

Accuracy is also measured as an amount of the ratio of successfully anticipated positive outcomes to total positive values. The calculation is:

Precision = TP/ (TP+FP)

Detection Rate, or Recall, is the ratio of correctly predicted positive values to the total number of positive values that exist. It is calculated as:

Recall = TP/ (TP+FN)

F1 Score balances accuracy and recall to assess model performance. The weighted average of recall and accuracy is obtained using the formula:

F1-Score = 2× (Recall × Precision)/ (Recall + Precision)

4 Results

In this section, a detailed evaluation and interpretation of the data that was obtained from the application of the different selected models is given. The evaluation of each model is done by using a number of parameters such as accuracy, precision, F1 score, AUC, and ROC analysis as described in the previous section.

Table 1 allows for the evaluation of the model accuracy so as to show the performance of various models. The CNN+MLP model outperforms the other models, which brings together Convolutional Neural Networks and Multi-Layer Perceptron. Furthermore, the ensemble model, which is a composition of a number of base models, also outperforms other models.

Table 1: Comparative analysis of model accuracy

Model	**Accuracy**
CNN+LGBM	0.92
CNN+RF	0.83
CNN+DT	0.75
CNN+SVM	0.88
CNN+KNN	0.92
CNN+MLP	0.96
CNN	0.92
Ensemble	0.96

Figure 2 illustrates the graphical representation of accuracy comparison across various models. Both the ensemble model and the hybrid model based on CNN+MLP exhibit superior performance among other models.

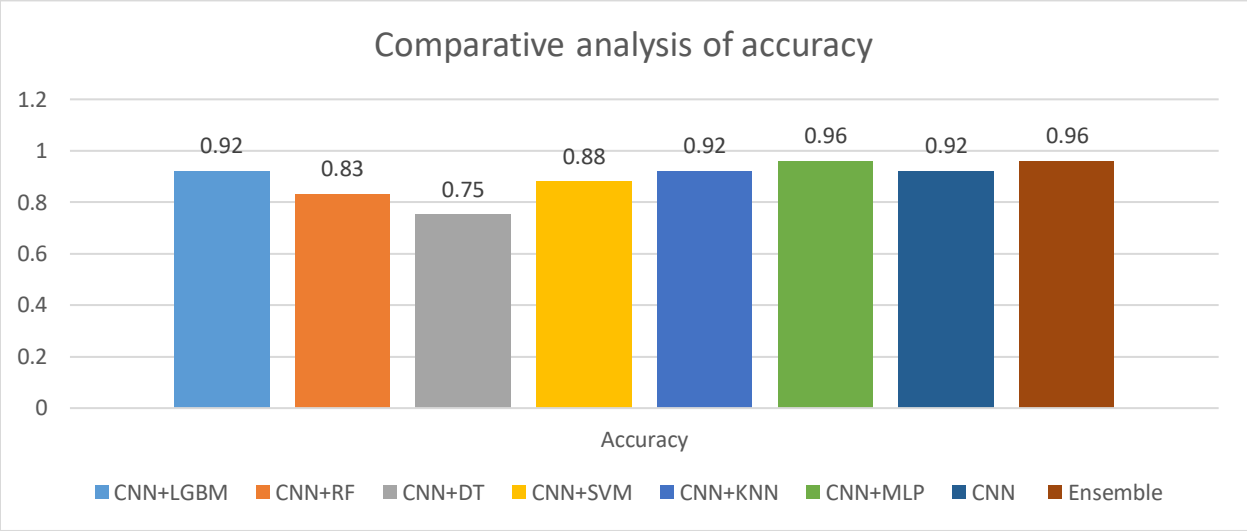

Figure 2: Comparative analysis of accuracy of different models

Each model corresponds to a specific model configuration, and its precision is presented in Table 2. The model CNN+MLP and the ensemble model show 0.92 and 0.96 precision. This indicates that these models are more effective in identifying intrusion than the other models evaluated.

Table 2: Precision comparison of various models

Model	**Precision**
CNN+LGBM	0.85
CNN+RF	0.77
CNN+DT	0.69
CNN+SVM	0.79
CNN+KNN	0.91
CNN+MLP	0.92
CNN	0.85
Ensemble	0.96

In Figure 3, the precision scores of various models are compared. This visual analysis provides insights into the performance of different models in terms of precision. The precision comparison proves that the CNN+MLP and ensemble models are significantly better than all other models.

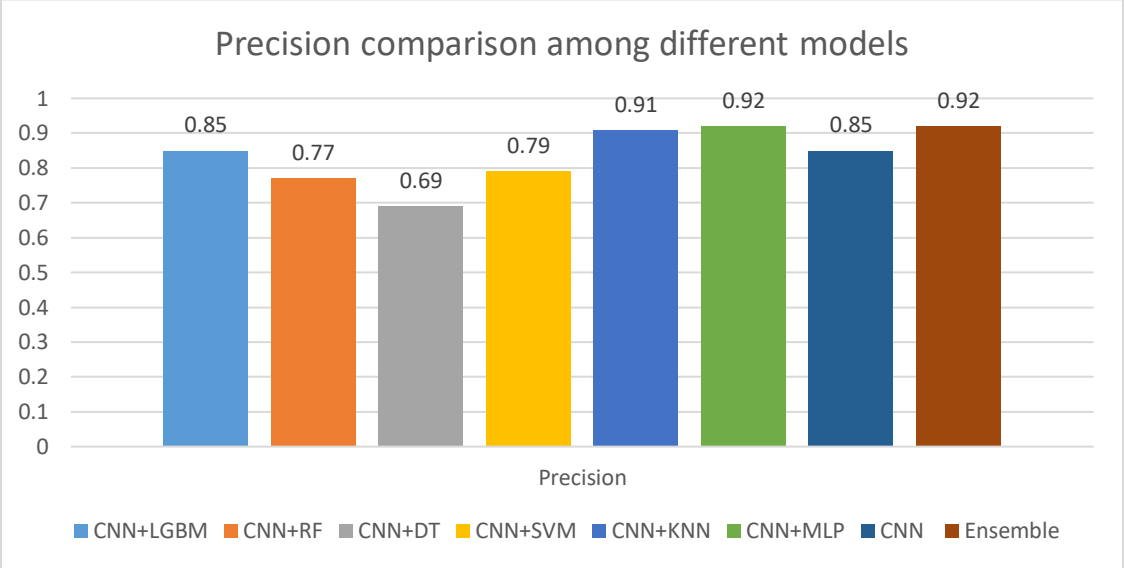

Figure 3: Precision comparison among different models

Performance comparison of different models is also shown in Table 3, where F1 scores are compared. The CNN+MLP model with 0.92 performance and the ensemble model with 0.96 performance are proved to be slightly better than the rest of the models.

Table 3: Comparative F1 scores across different models

Model	**F1 score**
CNN+LGBM	0.85
CNN+RF	0.77
CNN+DT	0.69
CNN+SVM	0.79
CNN+KNN	0.91
CNN+MLP	0.92
CNN	0.85
Ensemble	0.96

Table 4 presents a comparison of F1 scores of the three models. Every model configuration is shown, and the F1 score associated with it is indicated beside it. As it has been ascertained, the performance of this ensemble model is higher compared to other models.

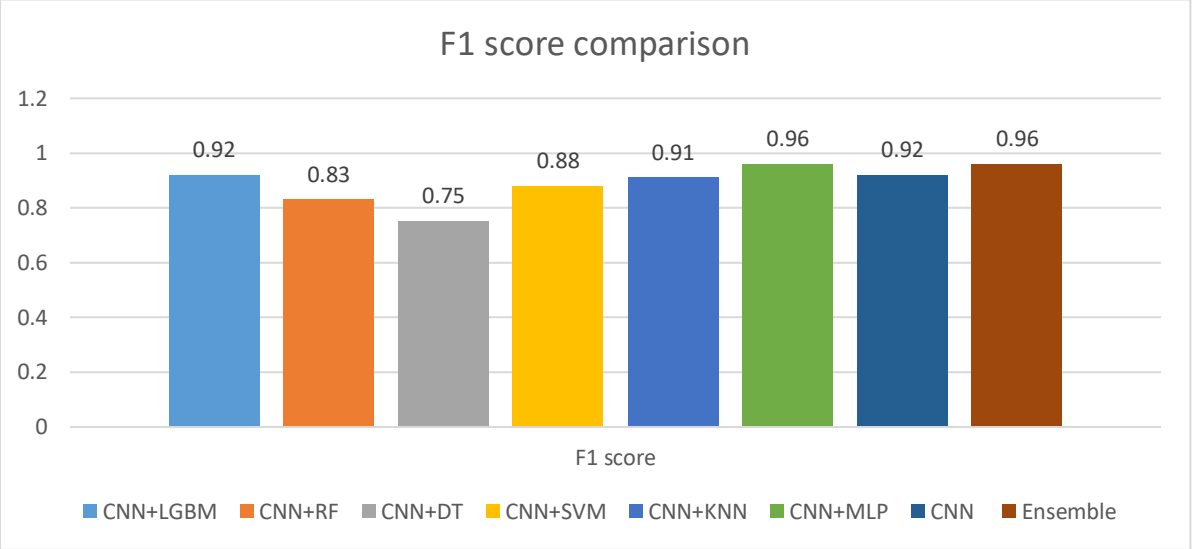

Figure 4: F1 score comparison

Table 4 also shows the difference in area under the curve (AUC) for various models. For the CNN+MLP model, AUC is equal to 0.92, and for the ensemble model, the AUC is equal to 0.96.

Table 4: AUC comparison among different models

Model	**AUC**
CNN+LGBM	0.85
CNN+RF	0.77
CNN+DT	0.69
CNN+SVM	0.79
CNN+KNN	0.91
CNN+MLP	0.92
CNN	0.85
Ensemble	0.96

Figure 5 shows a breakdown of the Area Under the Curve (AUC) between models. The results show that the ensemble model has achieved a better AUC score than other models.

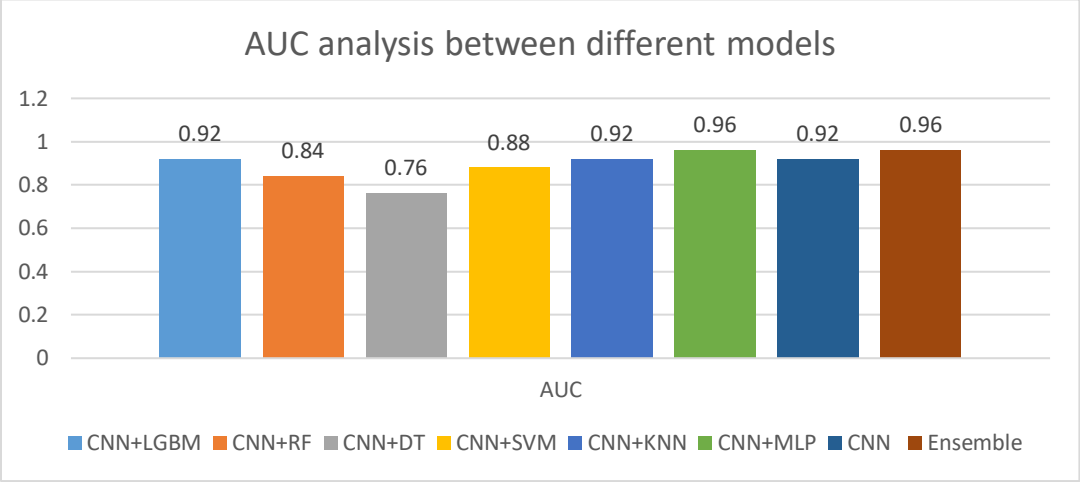

Figure 5: AUC analysis between different models

Figure 6 illustrates the analysis of Receiver Operating Characteristic (ROC) curves across different models. Each model configuration is depicted, and their respective ROC curves are compared. This visual representation offers insights into the comparative performance of various models based on their ROC curves.

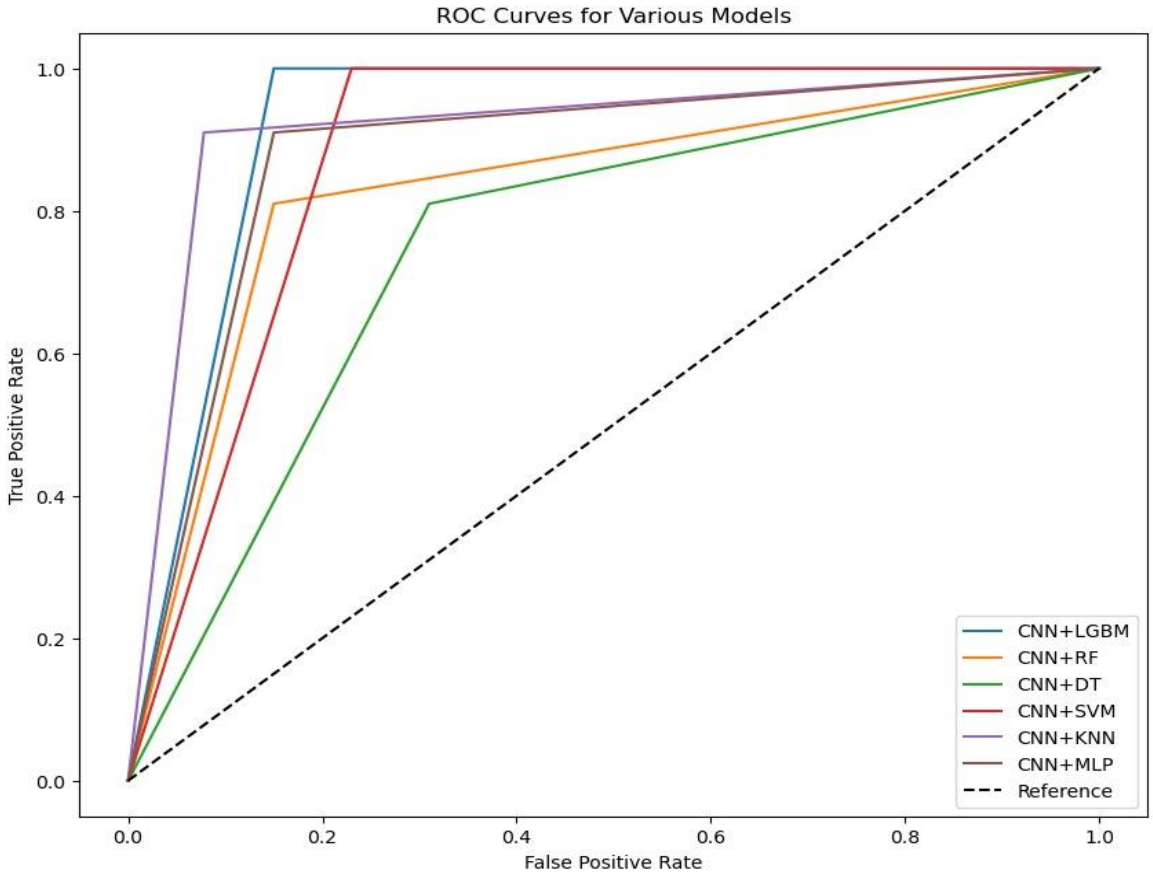

Figure 6: ROC curve analysis across different models

5 Conclusion

Various methods involving ML (ML) and DL (DL) have been applied to detect intrusions. However, it is clear from the results that every learning method has a range of accuracy. Deeper exploration is needed to assess the effectiveness of a hybrid approach in the context of the issue. Therefore, both DL for feature extraction and ML for classification purposes are proposed in this research. The hybrid DL-ML method proposes several model configuration methods, including CNN+LGBM, CNN+RF, CNN+DT, CNN+SVM, CNN+KNN, CNN+MLP, CNN and Ensemble. All these models were thoroughly tested and validated using the SIDD dataset, a standard for malware verification. Notably, the CNN+MLP model, which consists of Convolutional Neural Networks (CNN) coupled MLP (Multi-Layer Perceptron), is best within the class of models regarding a mix of features and classification processes. Also, the ensemble model, composed of algorithms of several individual models, is very effective.

In conclusion, this work's results further emphasize the importance of hybrid DL-ML approaches for intrusion detection. The CNN+MLP and Ensemble models offer promising solutions to these intrusion detection challenges. These models could be enhanced in the future, and other hybrid architectures could be embraced.

6 Acknowledgement

This Project was funded by the Deanship of Scientific Research (DSR) at King Abdulaziz University, Jeddah, under grant no.(IPP: 417-611-2025). The authors, therefore, acknowledge DSR’s technical and financial support with thanks.

References

[1]. A. Nagaraja, V. S. Kiran, P. H. S, and N. Rajasekhar, "A membership function for intrusion and anomaly detection of low-frequency attacks," in Proceedings of the First International Conference on Data Science, E-learning and Information Systems, ACM, Madrid, Spain, 2018, pp. 1–6. https://doi.org/10.1145/3279996.3280031.

[2]. T. Zoppi, A. Ceccarelli, T. Capecchi, and A. Bondavalli, "Unsupervised Anomaly Detectors to Detect Intrusions in the Current Threat Landscape," ACM/IMS Transactions on Data Science, vol. 2, no. 2, pp. 1-26, Apr. 2021. https://doi.org/10.1145/3441140.

[3]. H. Li, F. Wei, and H. Hu, "Enabling Dynamic Network Access Control with Anomaly-based IDS and SDN," in Proceedings of the ACM International Workshop on Security in Software Defined Networks & Network Function Virtualization, Scottsdale, Arizona, USA, 2019, pp. 13-16. DOI: 10.1145/3317549.3323403.

[4]. T. Zoppi, A. Ceccarelli, and A. Bondavalli, "Exploring anomaly detection in systems of systems," in Proceedings of the Symposium on Applied Computing, ACM, 2017, pp. 1139–1146. DOI: 10.1145/3019612.3019711.

[5]. J. Pacheco, V. Benitez, and L. Félix, "Anomaly behavior analysis for IoT network nodes," in Proc. 3rd Int. Conf. Future Netw. Distrib. Syst., Jul. 2019, pp. 1–6. DOI: 10.1109/ICFNDT.2019.8859388.

[6]. Y. Sonmez, T. Tuncer, H. Gokal, and E. Avci, "Phishing Web Sites Features Classification Based on Extreme ML," in 6th International Symposium on Digital Forensic and Security (ISDFS), 2018.

[7]. R. Basnet, S. Mukkamala, and A. H. Sung, "Detection of Phishing Attacks: A ML Approach," Studies in Fuzziness and Soft Computing, pp. 373–383. DOI: 10.1007/978-3-319-05333-8_27.
[8]. G. Kamal and M. Manna, "Detection of Phishing Websites Using Naive Bayes Algorithm," Proceedings of the International Journal of Recent Research and Review, Vol. XI, Issue 4, December 2018, ISSN 2277-8322. DOI: 10.30953/ijrrr.11.4.02.
[9]. P. Singh, Y. P. S. Maravi, and S. Sharma, "Phishing websites detection through supervised learning networks," in 2015 International Conference on Computing and Communications Technologies (ICCCT).
[10]. M. Kaytan and D. Hanbay, "Effective classification of Phishing Webpages Based on New Rules by Using Extreme ML," Anatolian Journal of Computer Sciences, AJCS 17, pp: 15-36, ISSN: 2548- 1304, 2017.
[11]. W. Niu, X. Zhang, G. Yang, Z. Ma, and Z. Zhuo, "Phishing Emails Detection Using CS-SVM," in 2017 IEEE International Symposium on Parallel and Distributed Processing with Applications and 2017 IEEE International Conference on Ubiquitous Computing and Communications (ISPA/IUCC).
[12]. I. Tyagi, J. Shad, S. Sharma, S. Gaur, and G. Kaur, "A Novel ML Approach to Detect Phishing Websites," in 5th International Conference on Signal Processing and Integrated Networks (SPIN), 2018. DOI: 10.1109/SPIN.2018.8474287.
[13]. A. Jain and B. B. Gupta, "PHISH-SAFE: URL features based phishing detection system using ML," in Cyber Security, 2018, pp. 467-474. DOI: 10.1007/978-981-13-1716-9_41.
[14]. P. Pujara and M. B. Chaudhari, "Phishing Website Detection using ML: A Review."
[15]. S. Rathore, P. K. Sharma, V. Loia, Y. S. Jeong, and J. H. Park, "Social network security: Issues, challenges, threats, and solutions," Information Sciences, vol. 421, pp. 43-69, 2017. DOI: 10.1016/j.ins.2017.08.031.
[16]. S. Zaman and F. Karray, "Lightweight IDS based on features selection and IDS classification scheme," in 2009 International Conference on Computational Science and Engineering, vol. 3, 2009. DOI: 10.1109/CSE.2009.363.
[17]. P. Louvieris, N. Clewley, and X. Liu, "Effects-based feature identification for network intrusion detection," Neurocomputing, vol. 121, pp. 265-273, 2013. DOI: 10.1016/j.neucom.2012.11.034.
[18]. B. Alotaibi and K. Elleithy, "A majority voting technique for wireless intrusion detection systems," in 2016 IEEE Long Island Systems, Applications and Technology Conference (LISAT). DOI: 10.1109/LISAT.2016.7478809.
[19]. Usha, M., and P. J. W. N. Kavitha. "Anomaly based intrusion detection for 802.11 networks with optimal features using SVM classifier." Wireless Networks 23.8 (2017): 2431-2446.
[20]. Palmieri, Francesco, Ugo Fiore, and Aniello Castiglione. "A distributed approach to network anomaly detection based on independent component analysis." Concurrency and Computation: Practice and Experience 26.5 (2014): 1113-1129.
[21]. Mehrotra L., Saxena P.S., Doohan N.V. (2018) A Data Classification Model: For Effective Classification of Intrusion in an Intrusion Detection System Based on Decision Tree Learning Algorithm. In: Mishra D., Nayak M., Joshi A. (eds) Information and

Communication Technology for Sustainable Development. Lecture Notes in Networks and Systems, vol 9. Springer, Singapore. DOI: 10.1007/978-981-10-3932-4_7.

[22]. Ashoor A.S., Gore S. (2011) Difference between Intrusion Detection System (IDS) and Intrusion Prevention System (IPS). In: Wyld D.C., Wozniak M., Chaki N., Meghanathan N., Nagamalai D. (eds) Advances in Network Security and Applications. CNSA 2011. Communications in Computer and Information Science, vol 196. Springer, Berlin, Heidelberg. DOI: 10.1007/978-3-642-22540-6_48.

[23]. Singh A.N., Kumar S., Joshi R.C. (2011) Intrusion Detection System Based on Real Time Rule Accession and Honeypot. In: Wyld D.C., Wozniak M., Chaki N., Meghanathan N., Nagamalai D. (eds) Advances in Network Security and Applications. CNSA 2011. Communications in Computer and Information Science, vol 196. Springer, Berlin, Heidelberg. DOI: 10.1007/978-3-642-22540-6_29.

[24]. Mishra V., Vijay V.K., Tazi S. (2016) Intrusion Detection System with Snort in Cloud Computing: Advanced IDS. In: Satapathy S., Joshi A., Modi N., Pathak N. (eds) Proceedings of International Conference on ICT for Sustainable Development. Advances in Intelligent Systems and Computing, vol 408. Springer, Singapore. DOI: 10.1007/978-981-10-0129-1_48.

[25]. Brahmi I., Yahia S.B., Poncelet P. (2010) MAD-IDS: Novel Intrusion Detection System Using Mobile Agents and Data Mining Approaches. In: Chen H., Chau M., Li S., Urs S., Srinivasa S., Wang G.A. (eds) Intelligence and Security Informatics. PAISI 2010. Lecture Notes in Computer Science, vol 6122. Springer, Berlin, Heidelberg. DOI: 10.1007/978-3-642-13601-6_9.

[26]. Z. Wang, "Deep Learning-Based Intrusion Detection with Adversaries," in IEEE Access, vol. 6, pp. 38367-38384, 2018. DOI: 10.1109/ACCESS.2018.2854599.

[27]. W. Zhong, N. Yu and C. Ai, "Applying big data based DL system to intrusion detection," in Big Data Mining and Analytics, vol. 3, no. 3, pp. 181-195, Sept. 2020. DOI: 10.26599/BDMA.2020.9020003.

[28]. F. A. Khan, A. Gumaei, A. Derhab and A. Hussain, "A Novel Two-Stage DL Model for Efficient Network Intrusion Detection," in IEEE Access, vol. 7, pp. 30373-30385, 2019. DOI: 10.1109/ACCESS.2019.2899721.

[29]. M. Al-Qatf, Y. Lasheng, M. Al-Habib, and K. Al-Sabahi, "DL Approach Combining Sparse Autoencoder With SVM for Network Intrusion Detection," in IEEE Access, vol. 6, pp. 5284-52856, 2018, doi: 10.1109/ACCESS.2018.2869577.

[30]. C. Yin, Y. Zhu, J. Fei, and X. He, "A DL Approach for Intrusion Detection Using Recurrent Neural Networks," in IEEE Access, vol. 5, pp. 21954-21961, 2017, doi: 10.1109/ACCESS.2017.2762418.

[31]. C. Xu, J. Shen, X. Du, and F. Zhang, "An Intrusion Detection System Using a Deep Neural Network With Gated Recurrent Units," in IEEE Access, vol. 6, pp. 48697-48707, 2018, doi: 10.1109/ACCESS.2018.2867564.

[32]. R. Vijayanand, D. Devaraj, and B. Kannapiran, "A Novel DL Based Intrusion Detection System for Smart Meter Communication Network," in 2019 IEEE International Conference on Intelligent Techniques in Control, Optimization and Signal Processing (INCOS), Tamilnadu, India, 2019, pp. 1-3, doi: 10.1109/INCOS45849.2019.8951344.

[33]. S. Wasi, S. Shams, S. Nasim, and A. Shafiq, "Intrusion Detection Using DL and Statistical Data Analysis," in 2019 4th International Conference on Emerging Trends in

Engineering, Sciences and Technology (ICEEST), Karachi, Pakistan, 2019, pp. 1-5, doi: 10.1109/ICEEST48626.2019.8981688.

[34]. N. Chockwanich and V. Visoottiviseth, "Intrusion Detection by DL with TensorFlow," in 2019 21st International Conference on Advanced Communication Technology (ICACT), PyeongChang Kwangwoon_Do, Korea (South), 2019, pp. 654-659, doi: 10.23919/ICACT.2019.8701969.

[35]. G. Zhao, C. Zhang, and L. Zheng, "Intrusion Detection Using Deep Belief Network and Probabilistic Neural Network," in 2017 IEEE International Conference on Computational Science and Engineering (CSE) and IEEE International Conference on Embedded and Ubiquitous Computing (EUC), Guangzhou, 2017, pp. 639-642, doi: 10.1109/CSE-EUC.2017.119.

[36]. Sun, Y., Esaki, H., & Ochiai, H. (2020). Adaptive intrusion detection in the networking of large-scale lans with segmented federated learning. IEEE Open Journal of the Communications Society, 2, 102-112.

[37]. Agrawal, S., Sarkar, S., Aouedi, O., Yenduri, G., Piamrat, K., Alazab, M., ... & Gadekallu, T. R. (2022). Federated learning for intrusion detection system: Concepts, challenges and future directions. Computer Communications.

[38]. Haruna Chiroma, Ahmad Shukri Mohd Noor, Sameem Abdulkareem, Adamu I. Abubakar, Arief Hermawan, Hongwu Qin, Mukhtar Fatihu Hamza, Tutut Herawan, Neural Networks Optimization through Genetic Algorithm Searches: A Review, Applied Mathematics & Information Sciences, Volume 11, No. 6 PP: 1543-1564 (2017) doi:10.18576/amis/110602

[39]. C. Ashwini, V. Sellam, Corn Disease Detection based on Deep Neural Network for Substantiating the Crop Yield, Applied Mathematics & Information Sciences, Volume 16, No. 3 PP: 423-433 (2022) doi:10.18576/amis/160304

[40]. Hamdy H El-Sayed, Shereen K. Refaay, Samia A. Ali, Moumen T. El-Melegy, Chain based Leader Selection using Neural Network in Wireless Sensor Networks protocols, Applied Mathematics & Information Sciences, Volume 16, No. 4 PP: 643-653 (2022) doi:10.18576/amis/160418

[41]. T. M. Shahwan, A Comparison of Bayesian Methods and Artificial Neural Networks for Forecasting Chaotic Financial Time Series, Journal of Statistics Applications & Probability, Vol. 1, No. 2 PP: 89-100 (2012)

[42]. Maksat Kanatov, Lyazzat Atymtayeva, Deep Convolutional Neural Network based Person Detection and People Counting System, Advanced Engineering Technology and Application, Vol. 7, No. 3 PP: 21-25 (2018) doi:10.18576/aeta/070301

[43]. Samat Bukenov, Askar Akshabayev, Using Neural Networks to Improve Emotional State of Person, Advanced Engineering Technology and Application, Vol. 5, No. 3 PP: 65-68 (2016) doi:10.18576/aeta/050303

[44]. M. E. Karar, M. F. Al-Rasheed, A. F. Al-Rasheed, Omar Reyad, IoT and Neural Network-Based Water Pumping Control System For Smart Irrigation, Information Sciences Letters, Vol. 9, No. 2 PP: 107-112 (2020) doi:10.18576/isl/090207

[45]. Meshal Mohammed Al Anazi, Osama R. Shahin, A Machine Learning Model for the Identification of the Holy Quran Reciter Utilizing K-Nearest Neighbor and Artificial Neural Networks, Information Sciences Letters, Vol. 11, No. 04 (2022) PP: 1093-1102 doi:10.18576/isl/110410

[46]. Marghny H. Mohammed, Botheina H. Ali, Ahmed I. Taloba, Self-adaptive DNA-based Steganography Using Neural Networks, Information Sciences Letters, Vol. 8, No. 1 PP: 15-23 (2019) doi:10.18576/isl/080102

[47]. K. Thilagavathi, A. Vasuki, A Novel Hyperspectral Image Classification Technique Using Deep Multi-Dimensional Recurrent Neural Network, Applied Mathematics & Information Sciences, Volume 13, No. 6 PP: 955-963 (2019) doi:10.18576/amis/130608