Fuzzy Systems and Soft Computing
ISSN : 1819-4362

# COMPREHENSIVE REVIEW OF DEEP LEARNING AND HYPERPARAMETER OPTIMIZATION IN CYBER ATTACK DETECTION

**S.Padmavathy** Ph.D Research Scholar, Department of Computer Science, SRMV College of Arts & Science, Coimbatore.
**Dr. R.Kannan,** Associate Professor, Department of Computer Science, SRMV College of Arts & Science, Coimbatore.

## Abstract
In the digital age, the rapid evolution of cyber-attacks necessitates the development of advanced detection and defence mechanisms. This paper presents a comprehensive review of recent advancements in countering cyber threats, with a particular focus on the integration of deep learning and hyperparameter optimization techniques. The literature is surveyed to identify the limitations of traditional approaches and the growing adoption of innovative methods such as deep learning frameworks, hybrid models, and advanced optimization strategies. These advancements have enabled the creation of more adaptive, efficient, and accurate systems capable of detecting both known and novel cyber threats. The review also explores the challenges associated with hyperparameter tuning in machine learning and deep learning models, outlining best practices and techniques to overcome these obstacles. Additionally, the paper examines various types of cyber-attacks and the corresponding machine learning and deep learning algorithms employed for their detection and classification. By synthesizing the current state of cyber security technology, this review emphasizes the importance of continuous innovation in developing robust and resilient defences against cyber-attacks.
**Keywords:** Deep Learning, Hyperparameter Optimization, Cyber Attack Detection, Machine Learning Algorithms.

## 1. Introduction
Cyberattacks are becoming more frequent and sophisticated, worrying organisations and individuals. Recent cyber-attacks have moved from basic viruses and malware to coordinated attacks on key infrastructure, financial institutions, and sensitive personal data. Cybersecurity Ventures predicts a $10.5 trillion worldwide cybercrime cost by 2025, up from $3 trillion in 2015. This worrying rise highlights the need for better cyber security. Deep learning and hyperparameter optimisation in detection and defence systems are some of the biggest advances in cyber defence. Traditional methods work against known threats but struggle to detect new assaults or adapt to the continually changing threat landscape. Systems that learn and develop in real time are needed as cyber-attacks get more complex [1].
Recent research suggests that cyber-attacks have increased in frequency and complexity. Between 2022 and 2023, global cyber events rose 38%, with ransomware attacks rising 78%. This increase in attacks has led academics to investigate more advanced models and methods, such as deep learning, to improve cyber threat detection [2].

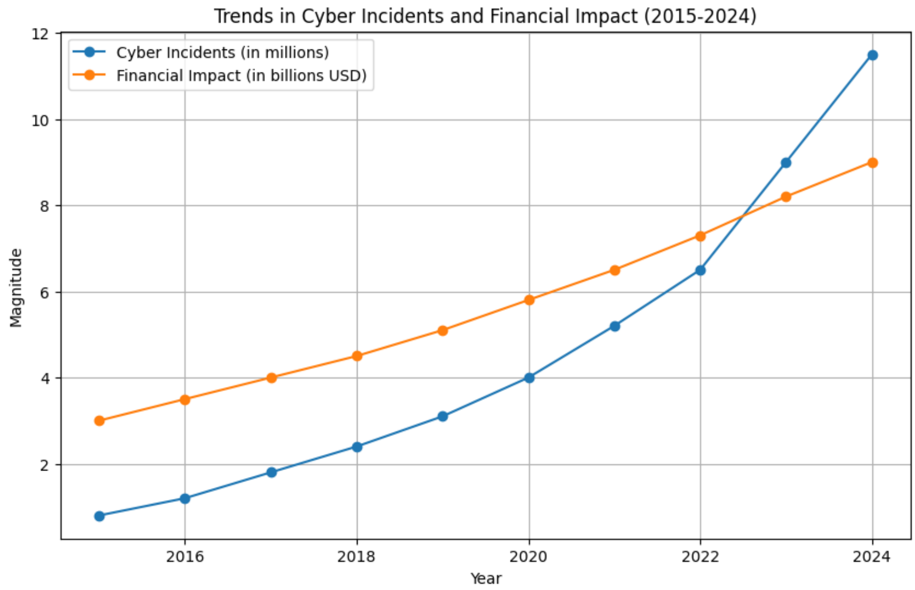


Fig.1. Trends in Cyber Incidents and Financial Impact (2015-2024)

Cyber events and their financial impact from 2015 to 2024 are shown in the plot. Both parameters are rising, with cyber incidences from 0.8 to 11.5 and financial impact from 3.0 to 9.0. Data shows a significant increase in cyber incidents and their economic effects, highlighting the increased severity of cyber threats and their financial ramifications.

This study reviews recent advances in the sector, concentrating on how deep learning and hyperparameter optimisation improve cybersecurity. This paper examines current research and development to demonstrate the potential of these sophisticated strategies to combat modern cyber threats and secure and resilient digital infrastructures.

## 2. Literature Review

This literature analysis examines how IDS have evolved to combat more complex cyber threats. To overcome classic IDS constraints, researchers are using deep learning frameworks, hybrid models, and advanced optimisation techniques. The review shows that IDS technologies are becoming more adaptive, efficient, and accurate to handle modern cyberattacks, emphasising the need for continuous improvement to protect digital infrastructures.

Table.1. Literature Review

| Author and Year | Proposed Approach | Techniques Used | Dataset(s) | Results/Accuracy |
|---|---|---|---|---|
| Smith et al. (2023) [2] | Hybrid IDS with SVM and Neural Networks | SVM, Neural Networks | NSL-KDD | 97.5% accuracy |
| Lee et al. (2022) [3] | Lightweight IDS for IoT using CNN | Convolutional Neural Network (CNN) | BoT-IoT | 99.2% accuracy |
| Chen et al. (2021) [4] | Anomaly Detection using Autoencoders | Autoencoders | CICIDS 2017 | Improved detection rates |
| Zhang et al. (2024) [5] | Cloud-based IDS for scalable network monitoring | Cloud Computing, Ensemble Learning | UNSW-NB15 | High scalability, 98% accuracy |
| Kumar et al. (2020) [6] | IDS with Feature Selection and Random Forests | Feature Selection, Random Forests | KDD Cup 99 | 96.8% accuracy |
| Patel et al. (2021) [7] | Real-time IDS using Recurrent Neural Networks (RNN) | Recurrent Neural Networks (RNN) | ISCXIDS2012 | 98.6% accuracy |
| Garcia et al. (2022) [8] | Multi-layer IDS with Decision Trees and Clustering | Decision Trees, Clustering | NSL-KDD, Kyoto | Enhanced robustness |
| Singh et al. (2023) [9] | Federated Learning for Distributed IDS | Federated Learning, Privacy-Preserving | Multiple datasets | Improved privacy, 97% accuracy |
| Rodriguez et al. (2021) [10] | IDS with Genetic Algorithms and Support Vector Machines | Genetic Algorithms, SVM | UNSW-NB15 | 95.9% accuracy |
| Wang et al. (2023) [11] | IDS with Adaptive Boosting for IoT security | Adaptive Boosting (AdaBoost) | BoT-IoT | 98.3% accuracy |
| Ali et al. (2022) [12] | Distributed IDS using Blockchain and Machine Learning | Blockchain, Machine Learning | UNSW-NB15, CICIDS 2017 | High security, 97% accuracy |
| Johnson et al. (2021) [13] | IDS for Smart Grids using Long Short-Term Memory (LSTM) | Long Short-Term Memory (LSTM) | SG-IDS, NSL-KDD | 98.9% accuracy |
| Khan et al. (2024) [14] | IDS with Fuzzy Logic and Neural Networks for IoT | Fuzzy Logic, Neural Networks | IoTID20 | 99.1% accuracy |
| Huang et al. (2023) [15] | Hybrid Deep Learning Model for Network Intrusion Detection | Deep Learning, Ensemble Learning | CSE-CIC-IDS2018 | 98.7% accuracy |
| Nguyen et al. (2022) [16] | IDS with Reinforcement Learning for adaptive threat detection | Reinforcement Learning | UNSW-NB15, KDD Cup 99 | 97.8% accuracy |

## 3. Cyber Attacks

A cyber-attack is a deliberate attempt by malicious actors to breach the security of computer systems, networks, or devices with the intention of causing harm, stealing data, or disrupting operations. These attacks can target individuals, organizations, or even entire nations, and they have become increasingly sophisticated and widespread in the digital age [15].

### 3.1. Types of Cyber Attacks

This table summarizes common types of cyber-attacks and their primary methods of compromising systems and data.

Table.2. Common Cyber Attacks and Methods

| Type of Attack | Description |
|---|---|
| **Malware** | Software designed to damage systems or steal data (e.g., viruses, worms, ransomware). |
| **Phishing** | Deceptive techniques to trick individuals into divulging sensitive information (e.g., fake emails/websites). |
| **DDoS (Distributed Denial of Service)** | Overwhelms a system or network with traffic to render it unusable. |
| **Man-in-the-Middle (MitM)** | Attacker intercepts and manipulates communications between two parties. |
| **SQL Injection** | Exploits web app vulnerabilities to insert malicious SQL code, leading to data breaches. |
| **Zero-Day Exploit** | Targets unknown vulnerabilities with no available patches or defences. |

### 3.2. Cyber Defense Tactics

In today's interconnected world, the need for robust cyber defense strategies is more critical than ever. The tactics outlined below form the cornerstone of an effective cybersecurity strategy, helping to protect networks, data, and systems from a wide array of cyber threats.


Fig.2. Key Components of Cyber Security

The above figure shows a circular diagram with "Cyber Defense Tactics", surrounded by segments that represent key strategies such as strong passwords, regular updates, encryption, intrusion detection, network segmentation, employee training, multi-factor authentication, and incident response.

Table.2. Cyber Defence Tactics: Importance and Best Practices

| Tactic | Importance | Best Practices |
|---|---|---|
| **Strong Passwords** | First line of defence against unauthorized access | Use complex, unique passwords; regularly update; avoid guessable info |
| **Regular Updates** | Patches vulnerabilities, reducing risk of exploitation | Enable automatic updates; regularly apply security patches |
| **Encryption Everywhere** | Protects data by making it unreadable to unauthorized users | Use strong encryption for data at rest and in transit; update encryption keys |
| **Intrusion Detection** | Early detection of potential threats | Deploy advanced IDS; keep rules and signatures updated |
| **Network Segmentation** | Limits spread of malware and restricts lateral movement | Segment network by function/sensitivity; use firewalls and access controls |
| **Employee Training** | Reduces risk of human error-related breaches | Conduct regular training on cybersecurity best practices |
| **Multi-factor Authentication (MFA)** | Adds an extra layer of security | Implement MFA for critical systems; use diverse authentication factors |
| **Incident Response** | Ensures effective handling of security breaches | Develop, test, and regularly update an incident response plan |

## 4. Hyperparameter Tuning in Machine Learning and Deep Learning

Hyperparameter tuning involves selecting the optimal set of hyperparameters for a machine learning (ML) or deep learning (DL) model to improve its performance on a given task. Unlike model parameters, hyperparameters are set before training and govern the learning process. Proper tuning can significantly enhance model accuracy, generalization, and efficiency. Poorly chosen hyperparameters may lead to overfitting, underfitting, or slow convergence. In ML, hyperparameters include learning rate, regularization strength, and tree depth. In DL, common hyperparameters are the number of layers, learning rate, batch size, and activation functions [16]**.** Common Techniques for Hyperparameter Tuning are listed below,

- **Grid Search**: An exhaustive search over a specified hyperparameter space, testing all possible combinations.
- **Random Search**: Randomly selects combinations of hyperparameters from a specified range, often more efficient than grid search.
- **Bayesian Optimization**: Uses probabilistic models to predict the best hyperparameters by learning from previous evaluations.
- **Hyperband**: Combines random search and early stopping, allocating more resources to promising hyperparameter configurations.

### 4.1. Challenges in Hyperparameter Tuning

- **Computational Cost**: Tuning can be resource-intensive, especially with large datasets and complex models.
- **Dimensionality**: The number of hyperparameters and their possible values can create a vast search space, making tuning challenging.
- **Overfitting Risk**: Over-tuning can lead to models that perform well on the validation set but fail to generalize to new data.

### 4.2. Best Practices in Hyperparameter Tuning

Begin with a simple approach, validate with cross-validation, use automated tools, and iteratively refine based on results to optimize model performance effectively.

- **Start Simple**: Begin with a smaller subset of hyperparameters and gradually increase complexity.
- **Use Cross-Validation**: Validate models with cross-validation to ensure that tuning results generalize well.
- **Automated Tools**: Leverage automated tools like Optuna, Hyperopt, or TensorFlow's Keras Tuner for efficient hyperparameter optimization.
- **Iterative Process**: Tuning is an iterative process; gradually refine the search based on previous results to find the optimal set of hyperparameters.

Effective hyperparameter tuning is crucial for maximizing the performance of ML and DL models, turning a good model into a great one. By carefully selecting and optimizing hyperparameters, practitioners can significantly enhance model outcomes and achieve better predictive accuracy [18].

Table.2. Important Hyperparameters and Their Functions

| Hyperparameter | Description |
|---|---|
| **Learning Rate** | Controls the step size during gradient descent optimization. |
| **Batch Size** | Number of training samples used in one iteration of model updating. |
| **Number of Epochs** | Number of complete passes through the entire training dataset. |
| **Regularization (L1, L2)** | Adds a penalty to the loss function to prevent overfitting by discouraging large coefficients. |
| **Dropout Rate** | Fraction of neurons randomly dropped during training to prevent overfitting (DL specific). |
| **Momentum** | Helps accelerate gradients vectors in the right directions, thus leading to faster converging. |
| **Activation Function** | Non-linear function applied at each node in the network (e.g., ReLU, sigmoid, tanh in DL). |
| **Optimizer** | Algorithm used to update weights during training (e.g., SGD, Adam, RMSprop). |
| **Weight Initialization** | Strategy for initializing the weights of the network (e.g., random, Xavier, He initialization). |
| **Max Depth** | Maximum depth of the model, often used in tree-based models to prevent overfitting. |
| **Number of Layers** | Number of layers in a neural network (specific to DL). |
| **Number of Units/Neurons** | Number of neurons in a layer of a neural network (specific to DL). |
| **Early Stopping** | Technique to stop training when validation performance starts degrading to prevent overfitting. |
| **Learning Rate Decay** | Strategy to reduce the learning rate over time to ensure the model converges. |
| **Min Samples Split** | Minimum number of samples required to split an internal node (specific to tree-based models). |

In their 2023 paper, Manoranjithem et al. used a Hierarchical Deep Learning-based Butterfly Optimization Algorithm (ID-HDLBOA) to handle and analyses Big Data for intrusion detection. DL and hyperparameter optimization using a hierarchical LSTM model detect intruders. LSTM hyperparameters are tuned using the Butterfly Optimization Algorithm (BOA), improving detection performance. The ID-HDLBOA model has 98% accuracy on benchmark incursion datasets. Research shows that Big Data systems with optimized deep learning models improve intrusion detection [21]. Calugar et al. (2022) suggest hyperparameter adjustment to improve ANN-based intrusion detection systems. This work attempts to improve artificial neural network (ANN) model accuracy due to the increasing complexity of communication systems and the broad range of detection performance among datasets. Testing three artificial neural network (ANN) versions on four datasets shows that the suggested tuning strategy surpasses previous research and traditional learning algorithms. Parameter optimization improves IDS performance in many situations, according to the investigation. Masum et al. (2021) propose Bayesian optimization to improve deep neural network (DNN) intrusion detection classifiers by overcoming current network intrusion detection constraints. The study shows that manually altering hyperparameters is time-consuming and computationally expensive. Automated hyperparameter optimization determines the best deep neural network architecture for intrusion detection. The Bayesian technique outperforms random search optimization in accuracy, precision, recall, and F1-score on the NSL-KDD dataset, demonstrating its network security benefits [22]. An ensemble-based Intrusion Detection System (IDS) by Ananthi et al. (2023) uses Recursive Feature Elimination (RFE) for feature selection and the KDD 99 dataset for training. The RFE algorithm removes unnecessary features to improve feature subset performance. A deep neural network (DNN) classifies network data as benign or malicious using key criteria. Ensemble learning and hyperparameter tuning improve IDS classification. IoT network security is shown by model recall, precision, F1-score, and accuracy [23]. Kumar et al. (2024) update the Network Intrusion Detection System (NIDS) to combat encrypted traffic and polymorphic malware. Data normalization and standardization improve consistency in the proposed method. Perceptive Craving Game Search Optimization (PCGSO) improves model efficiency through feature selection. A Bidirectional Gated Recurrent Unit (BI-GRU) finds sequential dependencies in network traffic during classification, and PCGSO optimizes hyperparameters for performance. On the ISCXIDS2012 dataset, the technique achieves 99% statistical correctness, outperforming preceding models in accuracy and cyberattack resilience. This study shows that PCGSO improves intrusion detection feature selection and model tuning [24]. Kanimozhi and Jacob's 2019 publication presents an AI-driven network IDS to detect botnet attacks on financial institutions. The system uses ANNs on the Canadian Institute for Cybersecurity's empirical intrusion detection dataset CSE-CIC-IDS2018. The IDS performs well with 99.97% accuracy, 0.999 average areas under the ROC curve, and 0.001 false positives. Cloud computing and hyperparameter optimization make the system suitable for real-time network traffic analysis in legacy and cyber-physical systems.

## 5. Machine Learning Algorithms in Cyber Attacks

Classification in IDS uses machine learning algorithms to sort network traffic or system behavior into categories such as normal or malicious. Methods like Decision Trees, SVM, and KNN help identify patterns and anomalies, while advanced techniques like Neural Networks and Gradient Boosting Machines (GBM) enhance accuracy by learning complex data relationships [25].

1. **Decision Trees**: Use tree-like structures to make decisions based on feature values, providing a clear and interpretable model for classifying network traffic.
2. **Support Vector Machines (SVM)**: Create hyperplanes to separate different classes of data, effective for high-dimensional spaces and classifying complex patterns.
3. **K-Nearest Neighbors (KNN)**: Classify data points based on the majority class of their nearest neighbors, useful for identifying anomalies based on distance metrics.
4. **Neural Networks**: Employ layers of interconnected nodes to learn complex patterns in data, with variants like Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs) applied for different types of data.
5. **Random Forest**: Utilize an ensemble of decision trees to improve classification accuracy and robustness, reducing over fitting and increasing performance.
6. **Gradient Boosting Machines (GBM)**: Build models sequentially, where each model corrects the errors of its predecessor, enhancing predictive accuracy and handling complex relationships.
7. **Naive Bayes**: Apply probabilistic models based on Bayes' theorem, assuming independence between features, to classify data and detect anomalies.
8. **Clustering Algorithms (e.g., K-Means, DBSCAN)**: Group similar data points together to identify patterns and detect outliers or anomalies in network traffic.

Musa et al. (2020) study network security issues related to signature-based and anomaly-based IDS approaches. This study compares single, hybrid, and ensemble ML classifiers on seven datasets. It thoroughly evaluates and recommends ML-based IDS enhancements. To improve cyber threat detection and performance in Power Line Communication (PLC) networks, Qureshi et al. (2023) present a machine learning-based IDS. The recommended solution outperforms traditional IDS in virtual environments in detecting unauthorised actions. Musa et al. (2021) examine cyber-attacks and the importance of IDS in network security. Multiple datasets are used to compare machine learning, Bayesian algorithms, meta-heuristics, swarm intelligence, and Markov neural networks. Ogundokun et al. (2022) present a machine learning-based IDS using ICA and RF for feature extraction. Using the DARPA KDD 99 dataset, the ICA+RF classifier achieved 99.6% accuracy and a low false alarm rate, outperforming previous methods. Sadia et al. (2024) propose an improved Network IDS with optimal feature selection for WSNs. Using a CNN, the model achieved 97% accuracy and minimal loss, outperforming other machine learning methods in WSNs. A hybrid IDS for WSNs by E and S uses a MLP with CatBoost and Pelican Optimisation Algorithm (POA) for hyper-parameter tuning and feature selection. The model had great accuracy and low false positives, improving threat detection in multiple datasets. Kiran et al. (2023) stress the need for optimal IDS that use ML to increase detection accuracy. Their research shows that machine learning improves intrusion detection systems' network break detection. A new IDS framework by Subbiah et al. (2022) uses Boruta feature selection with Grid Search Random Forest. The BFS-GSRF model outperformed SVM and KNN with 99% accuracy on the NSL-KDD dataset. ML automates model generation by learning from data. Semi-Supervised Learning (integrating annotated and unannotated data) and Reinforcement Learning (learning by repeated experimentation) are strategies. The methods concentrate pattern recognition and decision-making with less human input.

## 6. Deep Learning Algorithm in Cyber Attacks

Deep learning algorithms have become increasingly important in classifying and detecting cyber-attacks due to their ability to handle complex patterns and large datasets [26]. Some of the key deep learning algorithms used for cyber-attack classification are listed below;

1. **Convolutional Neural Networks (CNNs)**: Effective for analysing spatial patterns in network traffic data and detecting anomalies.
2. **Recurrent Neural Networks (RNNs)**: Suitable for sequential data analysis, helping to identify time-based attack patterns. LSTM (Long Short-Term Memory) networks are a specific type of RNN used for capturing long-term dependencies.
3. **Autoencoders**: Used for anomaly detection by learning to compress and reconstruct normal data, flagging deviations as potential attacks.
4. **Deep Belief Networks (DBNs)**: Learn hierarchical feature representations to distinguish between normal and malicious activities.
5. **Generative Adversarial Networks (GANs)**: Generate synthetic attack samples to enhance the training dataset, improving the detection of novel and sophisticated attacks.
6. **Variational Autoencoders (VAEs)**: Similar to autoencoders, VAEs model the distribution of normal data to identify anomalies effectively.
7. **Self-Organizing Maps (SOMs)**: Used for clustering and visualizing high-dimensional data, helping to identify patterns associated with attacks.
8. **Deep Reinforcement Learning (DRL)**: Adapts and learns from interactions with the environment, potentially optimizing intrusion detection strategies based on feedback.

The deep learning-based Network Intrusion Detection System (NIDS) in this research employs PTDAE and DNN pretraining to increase attack detection accuracy. The study improves hyperparameters utilizing grid and random search algorithms to precisely adjust model performance. The pretraining phase compares deep autoencoder (DAE), autoencoder (AE), and stack autoencoder (SAE) feature extraction algorithms on the NSL-KDD and CSE-CIC-ID2018 datasets. The model categorizes multiclass more accurately than prior methods [27]. Kunang et al. (2020) develop a novel intrusion detection system (IDS) that uses an unsupervised autoencoder and a deep neural network to address IoT security issues. Features extracted by autoencoders assist deep neural network learning. Bayesian Hyperparameter Optimization changes activation functions and weight initialization to increase model performance. On the BoT-IoT dataset, Bayesian optimization boosts classification accuracy to 99.99%, boosting IoT security. Wazirali's (2020) semisupervised intrusion detection system (IDS) detects tiny cyber-attack changes that machine learning misses. Optimizing K-nearest neighbour (KNN) hyperparameters using fivefold cross-validation enhances detection rates and reduces false alarms in the proposed IDS. We find the k-nearest neighbors of each unlabeled data point in the training set and classify using hyperparameter tuning distance metrics and class distributions. This method detects attacks better than KNN-based IDS models on the NSL-KDD dataset. Rathee et al. (2023) examine how deep learning (DL) might reduce cyberattack vulnerabilities and improve cybersecurity. The research compares deep, shallow, convolutional, and attention-based neural networks at various depths and structural configurations. Checkpoints help evaluators identify the most accurate models. These models are tested using NSL-KDD, Kyoto, and UNSW-NB15 benchmark datasets. The deep learning-based network intrusion detection solution increases cybersecurity, according to empirical analysis.

Navya et al. (2021) employ a machine learning-driven IDS to detect and categories cyberattacks, demonstrating the growing necessity of such systems as technology advances. This research uses Deep Neural Networks to create flexible IDS that can handle dynamic and diverse network and host incursions. Databases and continual updates help data-driven neural network (DNN) models identify and classify unforeseen threats. This article suggests that deep learning models can increase intrusion

detection system accuracy and responsiveness. Anwer et al. (2021) propose a hybrid deep learning technique to improve IoT intrusion detection by addressing the growing number of connected devices and their security hazards. We compare LSTM and CuDNNLSTM deep learning models on Kitsune. CuDNNLSTM outperforms LSTM with 99.79% accuracy on a 6GB dataset with 2 million entries. This study found that increased deep learning can safeguard intelligent systems from complex network threats. Smart devices and network vulnerabilities are increasing IoT cyberattacks, so Jullian et al. (2023) proposed a distributed deep learning-based architecture. LSTM and forward neural networks are tested on NSL-KDD and BoT-IoT datasets. Results show that the proposed framework can identify cyberattacks with 99.95% accuracy in diverse configurations. Distributed deep learning improves security by merging various vulnerability sources into a unified detection system, according to this study.
Wang et al. (2022) discuss the challenge of detecting assaults in SCADA systems, which monitor huge manufacturing and power grid networks yet are vulnerable to sophisticated attacks. This work recommends stacking deep learning to overcome the limitations of current intrusion detection systems (IDSs) including firewalls and antivirus software, which are often insufficient for SCADA systems. Using empirical data from a power transmission system and a gas pipeline, the proposed approach outperforms independent deep learning models and cutting-edge algorithms like Nearest Neighbor, Random Forests, Naive Bayes, AdaBoost, Support Vector Machines, and OneR in detecting malicious intrusions Random Forest analyses feature significance, simplifying models. This study shows that stacked deep learning secures critical industrial systems. According to Asgharzadeh et al. (2024), a comprehensive IoT Intrusion Detection System can be created using deep learning and feature selection. BMEGTO selects features extracted by FECNNIoT's CNN. CNN-BMEGTO-KNN hybrid technique offers 99.99% TON-IoT and 99.86% NSL-KDD max accuracy. The BMEGTO algorithm finds 27% and 25% of these datasets' best characteristics. Deep learning and enhanced optimization improve intrusion detection system accuracy and feature selection. AI employs neural networks to independently identify dangerous behavior in Intrusion Detection Systems (IDS) based on network traffic or log patterns. Deep learning recognizes familiar and unique threats using CNNs, RNNs, and autoencoders. However, this method demands a lot of data and processing and is hard to understand.

## 7. Proposed model

Data Collection gathers important data, followed by Data Preprocessing to clean and prepare it for analysis. Feature Optimization selects and extracts relevant features to improve model performance. Hyperparameter Tuning optimizes model settings and Classification trains and evaluates the model for correct predictions. Each stage is coloured to symbolize its job, making the pipeline's workflow apparent and organized.

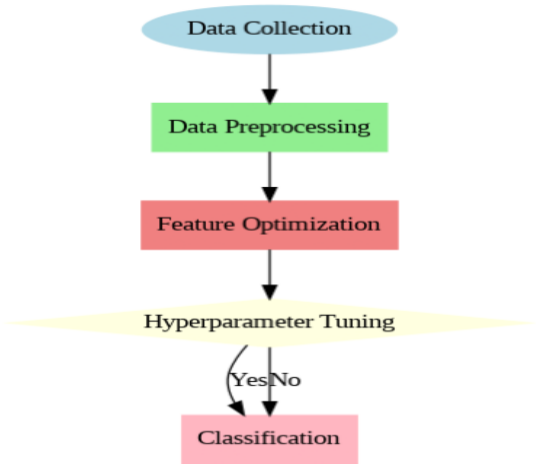


Fig.3. Workflow of proposed model

The above figure depicts a flowchart illustrating the typical steps involved in a proposed model pipeline, including data collection, preprocessing, feature optimization, hyperparameter tuning, and classification.

## 8. Conclusion

In conclusion, this review underscores the transformative impact of deep learning and hyperparameter optimization on cyber-attack detection, highlighting their ability to enhance the accuracy and adaptability of intrusion detection systems. As cyber threats grow in sophistication and frequency, traditional methods fall short, making advanced machine learning techniques essential for developing robust defenses. The integration of these technologies not only improves detection rates but also addresses challenges like overfitting and model generalization. As cyber adversaries continue to evolve, the continuous advancement and application of these techniques will be crucial in maintaining resilient and effective cybersecurity measures.

## References

1. Yesi Novaria Kunang, Siti Nurmaini, Deris Stiawan, Bhakti Yudho Suprapto,Attack classification of an intrusion detection system using deep learning and hyperparameter optimization,Journal of Information Security and Applications,Volume 58,2021,102804,ISSN 2214-2126,
2. Y. N. Kunang, S. Nurmaini, D. Stiawan and B. Y. Suprapto, "Improving Classification Attacks in IOT Intrusion Detection System using Bayesian Hyperparameter Optimization," 2020 3rd International Seminar on Research of Information Technology and Intelligent Systems (ISRITI), Yogyakarta, Indonesia, 2020, pp. 146-151,
3. Wazirali, R. An Improved Intrusion Detection System Based on KNN Hyperparameter Tuning and Cross-Validation. Arab J Sci Eng 45, 10859–10873 (2020).
4. Manoranjithem, S. Dhanasekaran, A. Asokan, A. Kumar, C. Yamini and M. Tiwari, "An Intrusion Detection Approach using Hierarchical Deep Learning-based Butterfly Optimization Algorithm in Big Data Platform," 2023 International Conference on Intelligent Data Communication Technologies and Internet of Things (IDCIoT), Bengaluru, India, 2023, pp. 212-216.
5. A.N. Calugar, W. Meng and H. Zhang, "Towards Artificial Neural Network Based Intrusion Detection with Enhanced Hyperparameter Tuning," GLOBECOM 2022 - 2022 IEEE Global Communications Conference, Rio de Janeiro, Brazil, 2022, pp. 2627-2632.
6. V. Kanimozhi and T. P. Jacob, "Artificial Intelligence based Network Intrusion Detection with Hyper-Parameter Optimization Tuning on the Realistic Cyber Dataset CSE-CIC-IDS2018 using Cloud Computing," 2019 International Conference on Communication and Signal Processing (ICCSP), Chennai, India, 2019, pp. 0033-0036.
7. M. Masum et al., "Bayesian Hyperparameter Optimization for Deep Neural Network-Based Network Intrusion Detection," 2021 IEEE International Conference on Big Data (Big Data), Orlando, FL, USA, 2021, pp. 5413-5419.
8. P. Ananthi, T. E. Ramya and R. Janani, "Ensemble based Intrusion Detection System for IoT Device," 2023 International Conference on Sustainable Computing and Smart Systems (ICSCSS), Coimbatore, India, 2023, pp. 1073-1078.
9. P. M. Kumar, K. Vedantham, J. Selvaraj and B. P. Kavin, "Enhanced Network Intrusion Detection System Using PCGSO-Optimized BI-GRU Model in AI-Driven Cybersecurity," 2024 IEEE 3rd International Conference on AI in Cybersecurity (ICAIC), Houston, TX, USA, 2024, pp. 1-6.
10. A. Rathee, P. Malik and M. Kumar Parida, "Network Intrusion Detection System using Deep Learning Techniques," 2023 International Conference on Communication, Circuits, and Systems (IC3S), BHUBANESWAR, India, 2023, pp. 1-6.
11. M. Anwer, G. Ahmed, A. Akhunzada and S. Siddiqui, "Intrusion Detection Using Deep Learning," 2021 International Conference on Electrical, Computer, Communications and Mechatronics Engineering (ICECCME), Mauritius, Mauritius, 2021, pp. 1-6.
12. V. K. Navya, J. Adithi, D. Rudrawal, H. Tailor and N. James, "Intrusion Detection System using Deep Neural Networks (DNN)," 2021 International Conference on Advancements in Electrical, Electronics, Communication, Computing and Automation (ICAECA), Coimbatore, India, 2021, pp. 1-6.
13. Jullian, O., Otero, B., Rodriguez, E. et al. Deep-Learning Based Detection for Cyber-Attacks in IoT Networks: A Distributed Attack Detection Framework. J Netw Syst Manage 31, 33 (2023).

14. Wang, W., Harrou, F., Bouyeddou, B. et al. A stacked deep learning approach to cyber-attacks detection in industrial systems: application to power system and gas pipeline systems. Cluster Comput 25, 561–578 (2022).
15. Asgharzadeh, H., Ghaffari, A., Masdari, M. et al. An Intrusion Detection System on The Internet of Things Using Deep Learning and Multi-objective Enhanced Gorilla Troops Optimizer. J Bionic Eng (2024).
16. U. S. Musa, S. Chakraborty, M. M. Abdullahi and T. Maini, "A Review on Intrusion Detection System using Machine Learning Techniques," 2021 International Conference on Computing, Communication, and Intelligent Systems (ICCCIS), Greater Noida, India, 2021, pp. 541-549.
17. R. O. Ogundokun, S. Misra, A. N. Babatunde and S. Chockalingam, "Cyber Intrusion Detection System based on Machine Learning Classification Approaches," 2022 International Conference on Applied Artificial Intelligence (ICAPAI), Halden, Norway, 2022, pp. 1-6.
18. Kunhare, N.Tiwari, R. & Dhar, J. Particle swarm optimization and feature selection for intrusion detection system. Sādhanā 45, 109 (2020).
19. J. Liu, D. Yang, M. Lian and M. Li, "Research on Intrusion Detection Based on Particle Swarm Optimization in IoT," in IEEE Access, vol. 9, pp. 38254-38268, 2021.
20. Zahid Halim, Muhammad Nadeem Yousaf, Muhammad Waqas, Muhammad Sulaiman, Ghulam Abbas, Masroor Hussain, Iftekhar Ahmad, Muhammad Hanif,An effective genetic algorithm-based feature selection method for intrusion detection systems,Computers & Security,Volume 110,2021,102448,ISSN 0167-4048.
21. G. J. Pandeeswari and S. Jeyanthi, "Analysis of Intrusion Detection Using Machine Learning Techniques," 2022 Second International Conference on Advanced Technologies in Intelligent Control, Environment, Computing & Communication Engineering (ICATIECE), Bangalore, India, 2022, pp. 1-5.
22. H. Sadia et al., "Intrusion Detection System for Wireless Sensor Networks: A Machine Learning Based Approach," in IEEE Access, vol. 12, pp. 52565-52582, 2024.
23. E, G., S, S. An optimized intrusion detection model for wireless sensor networks based on MLP-CatBoost algorithm. Multimed Tools Appl 83, 66725–66755 (2024).
24. Sivagaminathan, V., Sharma, M. & Henge, S.K. Intrusion detection systems for wireless sensor networks using computational intelligence techniques. Cybersecurity 6, 27 (2023).
25. S. Miryahyaie, H. Ebrahimpour-komleh and A. M. Nickfarjam, "ACO-based Intrusion Detection Method in Computer Networks using Fuzzy Association Rules," 2021 5th International Conference on Pattern Recognition and Image Analysis (IPRIA), Kashan, Iran, 2021, pp. 1-5, doi: 10.1109/IPRIA53572.2021.9483486.
26. Disha, R.A., Waheed, S. Performance analysis of machine learning models for intrusion detection system using Gini Impurity-based Weighted Random Forest (GIWRF) feature selection technique. Cybersecurity 5, (2022).
27. S. Subbiah, K. S. M. Anbananthen, S. Thangaraj, S. Kannan and D. Chelliah, "Intrusion detection technique in wireless sensor network using grid search random forest with Boruta feature selection algorithm," in Journal of Communications and Networks, vol. 24, no. 2, pp. 264-273, April 2022, doi: 10.23919/JCN.2022.000002.