Journal of Global Research in Electronics and Communication
Volume 2, No. 2, February 2025
Available Online at: https://jgrec.info/index.php/jgrec
ISSN: 2321-3175

RESEARCH PAPER

Machine Learning for Cyber Threat Detection and Prevention in Critical Infrastructure

Sagar Bharat Shah
Department of Operations, Business Analytics, and Information Systems (OBAIS),
University of Cincinnati, Cincinnati, OH, USA
Shah2sb@mail.uc.edu

***Abstract*—Cybersecurity has been getting a lot of attention lately due to the proliferation of important applications and the exponential rise of data networks and computers. Cybercrimes that are well-planned and ongoing pose a greater threat to the Internet. Because hackers are smart enough to get around all of the conventional security procedures in place to detect and prevent cyberattacks, these measures are worthless. There are a lot of cybersecurity apps that use machine learning (ML) methods. This study proposes an advanced cyber threat detection framework leveraging machine learning techniques on the UNSW-NB15 dataset. The proposed Inception model is benchmarked against conventional classifiers, including Random Forest (RF), k-Nearest Neighbors (KNN), and Multi-Layer Perceptron (MLP). Experimental results demonstrate that the Inception model outperforms existing approaches, achieving an accuracy of 98.40%, precision of 99%, recall of 97.90%, and an F1-score of 98.50%. Comparative analysis highlights its superior capability in threat detection and classification. Furthermore, visualization techniques, including confusion matrices and performance graphs, validate the model's effectiveness. These results highlight the promise of models based on deep learning to improve cybersecurity by providing an efficient and scalable way to identify and prevent intrusions in real time.**

Keywords*—Cybersecurity, Cyber Threat Detection, Intrusion, Artificial Intelligence (AI), Machine Learning (ML), Critical Infrastructure Security, Network Security, Threat Intelligence, UNSW-NB15.

I. Introduction

The rapid expansion of digital technologies has transformed the way individuals, organizations, and governments operate, driving increased connectivity through networks and the Internet. While this connectivity has enabled innovation, efficiency, and global collaboration, it has also introduced significant cybersecurity challenges[1]. As digital infrastructures grow, so do the risks associated with cyberattacks, ranging from malware infections and ransomware to sophisticated nation-state cyber threats[2][3]. These attacks can target sensitive data, disrupt operations, and cause financial losses, making cybersecurity a critical concern for modern society. Cyber threats continue to evolve in complexity, exploiting vulnerabilities in networks, software applications, and cloud environments[4]. Cybercriminals defeat conventional security measures by using sophisticated methods, including social engineering, DoS assaults, and zero-day vulnerabilities[5]. As a result, cybersecurity has shifted from reactive defense strategies to proactive threat detection and prevention, requiring robust and adaptive security frameworks. In this context, Cyber Threat Detection and Prevention (CTDP) has become an essential domain, particularly in critical infrastructure, where cyberattacks can have severe consequences on national security, public safety, and economic stability[6][7].

Critical infrastructure, including power grids, healthcare systems, financial institutions, and transportation networks, plays a vital role in societal functions[8]. Cyberattacks on these systems can lead to widespread service disruptions, data breaches, and financial losses. Traditional security mechanisms such as firewalls, intrusion detection systems (IDS), and rule-based security policies are proving inadequate against sophisticated, AI-powered cyber threats[9][10]. The advent of AI and ML has been a game-changer in cybersecurity, automating threat identification, anomaly analysis, and real-time intrusion prevention, among other tasks[11]. AI-driven cybersecurity frameworks leverage ML algorithms to detect malicious activities, predict potential threats, and respond autonomously to cyber incidents[12]. Techniques such as anomaly detection, behavioral analytics, and predictive modeling enhance security measures by identifying and mitigating attacks before it cause harm.

A. *Motivation and Contribution*

This work is motivated by the need for advanced cybersecurity solutions as traditional detection methods struggle against evolving threats like APTs and zero-day attacks. Traditional security measures fall short, making ML essential for real-time threat detection and automated response. This research aims to develop an efficient intrusion detection system to enhance security and resilience against cyberattacks. The key contributions of this research are as follows:

- Leverages the UNSW-NB15 dataset to enhance machine learning-based cyber threat detection in critical infrastructure.
- Effective handling of missing values, duplicate removal, and feature scaling to enhance model performance.
- Identification of critical features influencing intrusion detection accuracy.
- Implementation and evaluation of classifiers such as Random Forest, k-NN, and MLP for cyber threat detection in critical infrastructure.
- Comparative analysis of models using accuracy, precision, recall, loss, and F1-score to assess detection effectiveness.

B. Justification and Novelty

The increasing sophistication of cyber threats requires more advanced detection mechanisms beyond conventional machine learning models. Traditional classifiers like RF, KNN, and MLP struggle with high-dimensional network traffic and evolving attack patterns, necessitating more robust solutions. This study introduces the Inception model, a deep learning-based approach that leverages multi-layered feature extraction to improve threat detection. Unlike traditional models, the Inception architecture enhances generalization by capturing intricate patterns in network traffic, making it more effective in distinguishing normal and malicious activities. The novelty of this work lies in integrating deep learning with extensive feature selection, preprocessing, and comparative evaluation, providing a comprehensive framework for enhancing cybersecurity defenses. Additionally, performance visualization techniques such as confusion matrices and learning curves offer deeper insights into model reliability and real-world applicability.

C. Structure of the paper

The study is structured as follows: In Section II presents a Literature review on cyber threat detection. In Section III, the methodology is utilized to compile the data for this study. Section IV provides the results and analysis of effective classification. At last, Section V provides the conclusion.

II. Literature Review

This section discusses the Literature review on, Advanced cybersecurity for threat detection and intrusion prevention Also, Table I provide the summary of these literature reviews discussed below:

Chaudhary et al. (2024) threat detection and automated response mechanisms. Compared to the current system, the results demonstrate notable increases in detection accuracy, False Positive and False Negative rates, and area under the curve (AUC) values across many test datasets. It achieves accuracy 89.7% for the existing system cyber threat detection and mitigation in cloud settings. By use of sophisticated anomaly detection and ML methods, the proposed system continuously analyzes large amounts of data to dynamically identify and eliminate new threats[13]

Sharma and Babbar (2024) attempts to evaluate these models' effectiveness in the environment of cyber threat detection. that offer a through grasp of the models' efficacy. It also provides helpful suggestions for choosing the best infrastructure for improving security using machine learning-based threat detection defense systems against changing cyber threats in actual network environments. the XGBoost model outperforms the other models (NB, LR and AdaBoost) with accuracy rates of 78%, 85%, and 90%[14].

Rajendran et al. (2024) cybersecurity solution adaptive to the changing threat framework's extraordinary success in recognizing as well as mitigating diverse cyber dangers, including insider threats and zero-day attacks, through practical testing and case studies. The framework establishes itself as a proactive. Behavioral analysis, AI explain ability, IoT security, and countering quantum computing concerns should be the main areas of future research. Revelations help the cybersecurity industry become more resilient and well-prepared. In on threat detection were conducted using deep learning algorithms, with a specific focus on CNNs and RNNs. The findings of the study reveal that a detection accuracy of 93% was achieved by CNN [15].

Gujar (2024) developed and validated through exercise scenarios in order to evaluate the impact on sectors of critical infrastructure like energy, transport, healthcare and others. The outcomes show that the system has received substantial enhancements in threat detection of multiple classes, with classification level of 94% and the false positive levels of 4%. The large-scale AI system was shown to be able to attain better scalability than the model trained on the local set without decreased performance during the high network utilization. Moreover, time responses for threat counteraction reduced dramatically as the system developed through iterations, demonstrating its real-time learning ability. It also describes difficulties that appear when applying the solution, for example, when it comes to data variety and integration of AI models with existing systems. Nonetheless, the solution that is proposed herein has the potential for achieving scalable and adaptive security in key sectors [16].

Almasri, Snober and Al-Haija (2022) employing SDNs, intrusion detection systems based on challenges, or pattern recognition employing ML. To find abnormalities, the Intrusion Detection and Prevention Systems scan network traffic for anomalies and compare it to known assaults. In order to effectively secure and defend the network against DoS and Port Scanning assaults, machine learning pattern recognition, network programmability features, and design are used. This ML technique was developed by selecting characteristics using Anova and then applying those features to several ML models. The most accurate ML model was the naïve Bayes one, coming in at 86.9% [17].

Atluri and Horne (2021) an approach to Cyber Threat Intelligence (CTI) is suggested, created, and evaluated. The study's findings are offered with the retrieved IOCs; the research used five distinct simulated assaults on a dataset derived from an ICS testbed. When it came to assessing accuracy, the Bagging Decision Trees model performed best with a score of 94.24% [18].

Tekin and Yilmaz (2021) used DL algorithms to analyze the cyber security data that Twitter provided. Classification of cyber threat intelligence (DDoS, malware, ransomware, etc.) is achieved using recursive neural networks applied to the dataset of tweets pertaining to cyber threat intelligence. An impressive 88.64% of the time were able to successfully determine the relevance of cyber threat information, and an even more impressive 89.49% of the time were able to identify the kind of threat data [19].

Current cyber threat detection models struggle with high false positives, poor adaptability, and computational inefficiencies. Traditional machine learning models lack robustness, while deep learning methods demand high resources, limiting real-time applications. To address these gaps, propose an Inception-based intrusion detection system, which enhances accuracy, reduces false alarms, and adapts effectively to evolving threats. Leveraging advanced feature extraction ensures superior detection while maintaining efficiency. Future research will focus on integrating explainable AI for better interpretability and optimizing computational efficiency to support real-time deployment, making cybersecurity systems more scalable, reliable, and responsive to emerging attack patterns.

TABLE I. BACKGROUND SUMMARY OF CYBER THREAT DETECTION USING MACHINE LEARNING

Author	Dataset	Methods	Findings	Limitations/Future Research
Chaudhary et al. (2024)	Multiple test datasets	Anomaly detection, Machine Learning (ML)	Achieved 89.7% accuracy in cyber threat detection and mitigation in cloud settings	Continuous adaptation needed for evolving threats
Sharma and Babbar (2024)	Cyber threat detection environment	XGBoost, Naïve Bayes (NB), Logistic Regression (LR), AdaBoost	XGBoost outperformed others with accuracy rates of 78%, 85%, and 90%	Model generalization in real-world environments
Rajendran et al. (2024)	Case studies and practical testing	CNN, RNN	CNN achieved 93% accuracy in threat detection	Future focus on behavioral analysis, AI explainability, and quantum security
Gujar (2024)	Critical infrastructure (energy, transport, healthcare)	Large-scale AI system for threat classification	Classification accuracy of 94%, false positive rate of 4%	Challenges in data variety and AI model integration with existing systems
Almasri, Snober, and Al-Haija (2022)	Network traffic datasets	Software-defined networks (SDNs), Machine Learning, Anova feature selection, Naïve Bayes	Naïve Bayes achieved highest accuracy of 86.9% for intrusion detection	Enhancing real-time detection and response mechanisms
Atluri and Horne, (2021)	Industrial Control System (ICS) testbed dataset	Cyber Threat Intelligence (CTI), Bagging Decision Trees	Bagging Decision Trees achieved highest testing accuracy of 94.24%	Need for testing across diverse attack scenarios
Tekin and Yilmaz (2021)	Twitter cybersecurity data	Deep Learning, Recursive Neural Networks (RNN)	88.64% accuracy in cyber threat intelligence classification, 89.49% accuracy in threat type classification	Improvement needed in handling large-scale social media data

III. METHODOLOGY

The methodology for Cyber Threat Detection and Prevention by using machine learning methods is systematized by preprocessing the given data, selection of features, model building and evaluation. The UNSW-NB15 dataset, consisting of 49 features of normal and malicious network traffic, is first used. Finally, the data are processed using mean or median values imputation, removing duplicate records, and Min-Max normalization for numeric data selection to normalize numeric features. A feature importance analysis reveals the main attributes that affect intrusion detection performance results. The next step is to use a predefined split in the dataset: 80% for training and 20% for testing. This will allow for a thorough assessment of the models. The processed data is then used to train ML classifiers like RF, KNN, and MLP. The models are then evaluated using a variety of metrics, including accuracy, precision, recall, loss, and F1-score. Then, a confusion matrix and heatmap analysis are used to examine classification performance and detect misclassifications. Finally, leveraging advanced machine learning techniques, this methodology enhances cyber threat detection and prevention in critical infrastructure. The overall process shows in Figure 1.

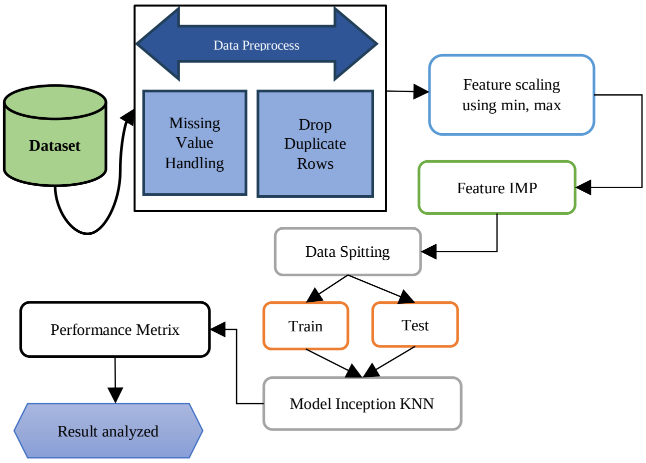

Fig. 1. Flowchart for cybersecurity of threat detectionEach step of the flowchart is provided in the next section.

Each step of the flowchart is provided in the next section.

A. Data Description

The UNSW-NB15 dataset was used in this investigation. The dataset, which has 49 attributes, is used to assess how well intrusion detection systems work. Realistic network traffic is included, mimicking both benign and malevolent activities. The attack category of the dataset is shown in Figure 2.

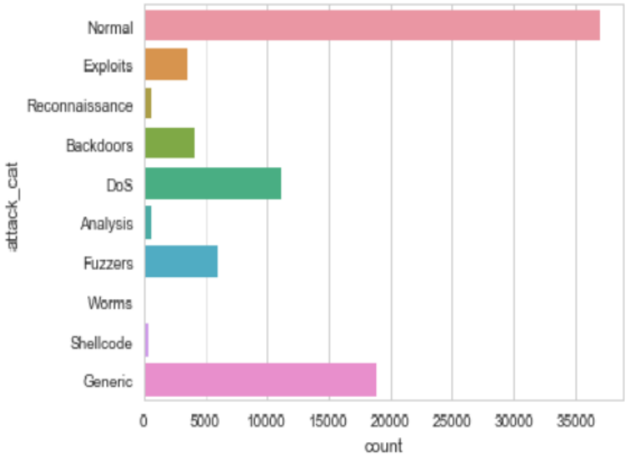

Fig. 2. Bar chart of attack categories in dataset

Figure 2 shows a horizontal barchart that displays the distribution of attack categories. The x-axis shows a count of instances, while the y-axis displays different attack categories (attack_cat). The "Normal" category exhibits the highest frequency, significantly surpassing all other attack types. Among attack categories, "Generic" and "DoS" (Denial of Service) attacks have substantial representation, followed by "Fuzzers," "Exploits," and "Backdoors". Other attack types, including "Reconnaissance," "Shellcode," "Analysis," and "Worms," appear less frequently. The imbalance in attack category distribution is an essential consideration for ML-based IDS.

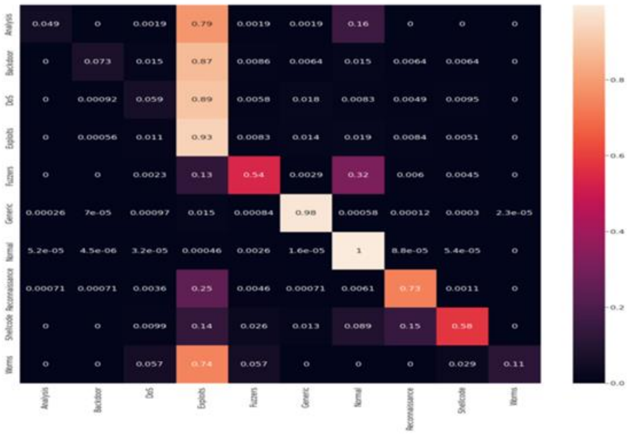

Fig. 3. Heatmap of UNSW-NB15 dataset

The following Figure 3, shows heatmap of UNSW-NB15 dataset across several categories: Fuzzers, Analysis, DDoS, Exploits, Backdoor, Normal, Generic, Shellcode, Reconnaissance, and Worms. The normalized proportions of predicted versus actual classifications, with darker shades indicating higher proportions and thus stronger classification accuracy, the model shows high accuracy in classifying 'Generic' and 'Normal' categories, indicated by values of 0.98 and 1.0 respectively, but exhibits confusion between 'Exploits' and 'DDoS' (0.89 vs 0.93), and misclassifies 'Shellcode' as 'Shellcode' only 58% of the time.

B. *Data preprocessing*

Data preparation is crucial for enhancing model performance and guaranteeing high-quality data [20]. In this pre-processing step, First, missing values are handled and Duplicate values are removed to avoid redundancy that are listed in below:

- **Handle missing value –**Replace missing values in a column with the mean or median value of that column and it missing values with a predefined constant value.
- **Drop duplicate rows –** It remove duplicate rows while modifying the original Data Frame avoid redundancy This process ensures that only unique records remain, improving data quality for analysis.

C. *Feature scaling using Min–Max*

Feature scaling is a technique used to normalize or standardize data so that numerical values fall within a specific range, enhancing a performance of ML models[21]. Decimal-scaling, Max-normalization and Min–Max scaling mathematically in Equation (1).

$$F = \frac{F - F_{min}}{F_{max} - F_{min}} \quad (1)$$

It is a (feature space) input vector denoted by U(f1,…,fn), where N is the sum of all occurrences (features) present in the domain,[22] the standardization computation.

D. *Feature Importance*

Enhancing the effectiveness, understandability, and general usefulness of the intrusion detection model requires analyzing the scores of important characteristics. The significance ratings of several elements in creating a successful intrusion detection model are shown in Figure 4.

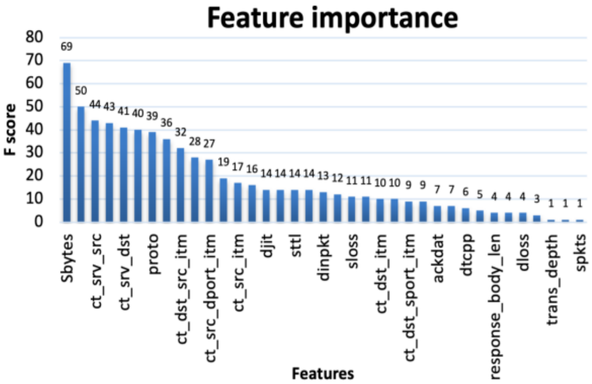

Fig. 4. Feature importance score

Figure 4 displayed a feature importance score for intrusion detection, with Sbytes being the most influential (F-score: 69), followed by ct_srv_src (44), ct_srv_dst (43), and proto (41). Other key features include ct_dst_src_ltm, ct_src_dport_ltm, and dlt, while features like response_body_len, oloss, and spkts have minimal impact. This highlights the critical network parameters essential for effective threat detection.

E. *Data splitting*

By splitting the datasets into 80% training and 20% testing sets, the chance of overfitting was decreased and thorough model performance monitoring throughout the whole dataset was made possible.

F. *Classification with Inception model*

Szegedy et al. presented the Inception model, a deep CNN architecture [23] with several branching structures in a single block, in the Large-Scale ImageNet Visual Identification. Without adding additional nodes to the network, each branch independently pulls features from the source maps, each with a unique receptive field size [24]. The dimensions of a feature map X are H × W × C. With n branches in an Inception unit and h, w, and ci being the dimensions of the output feature maps of the i-th branch, the last concatenation operation for each Inception is given by Equation (2):

$$\mathrm{H_{out}} \times \mathrm{W_{out}} \times \mathrm{C_{out}} = \mathrm{h} \times \mathrm{w} \times \sum_{i=1}^{n} \mathrm{C_i} \quad (2)$$

Inception's branches are made up of maximum pooling layers and convolutional layers. There is a single convolutional layer in each of the initial Inception unit's branches. The convolutional layer in the first branch collects samples and reduces their dimensions; in the second branch, it only collects samples; and in the third branch, the maximum pooling layer reduces their dimensions while keeping the greyscale maps' texture features. Two convolutional layers[25], make up the second Inception unit, which has a parallel branch topology. The third Inception unit has two branches: one for sampling and dimensionality reduction (a convolutional layer), and another for texture reduction (a maximum pooling layer) [26].

G. *Performance Metrics*

The Performance Metrics assess and contrast the algorithms' outputs using four metrics. These metrics allow for a thorough evaluation of the model's performance in identifying network intrusions; their values range from 0 to 1. Metrics such as TP (True Positive), TN (True Negative), FP (False Positive), and FN (False Negative) are crucial for evaluating the accuracy and reliability; meanwhile, the confusion matrix values must be understood. A performance matrix including accuracy, recall, precision, and F1-Score, is shown below:

Accuracy (ACC): It is the percentage of instances that were correctly classified relative to the total number of occurrences, taking into account both true positives and true negatives, as stated by Equation (3):

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \tag{3}$$

Precision: As shown in Equation (4), it is the percentage of positive outcomes that were accurately anticipated relative to the total number of positive instances.

$$Precision = \frac{TP}{TP+FP} \tag{4}$$

Recall (R): The percentage of accurate predictions relative to the overall number of true positives is called the recall value. Recall is defined mathematically in Equation (5) as follows:

$$Recall = \frac{TP}{TP+FN} \tag{5}$$

F1-Score: Equation (6) defines the F1-score as the harmonic mean of precision and recall, providing a fair assessment of the two measures.

$$F1-score = 2 * \frac{Precision*Recall}{Precision+Recall} \tag{6}$$

Loss: The mathematical function known as loss assesses the difference between the model's projected normal and intrusion class classifications of network traffic. Some loss functions, such cross-entropy loss, are unable to reliably forecast occurrences of minority classes [27].

IV. Result Analysis And Discussion

The experiments are conducted on the Python programming language and utilize other devices such as 32 GB of RAM, Personal computers, and intel core i7-8th Gen. This section provides the result analysis on the UNSW-NB15dataset for threat detection and prevention system model across performance including accuracy, precision, F1-score and recall for different classifications. The proposed Inception model performance are shows in Table II.

TABLE II. Results of Inception Model Performance on the UNSW-NB15 dataset For Cyber Threat Detection

Measure	Inception model
Accuracy	98.4
Precision	99.0
Recall	97.90
F1-score	98.50

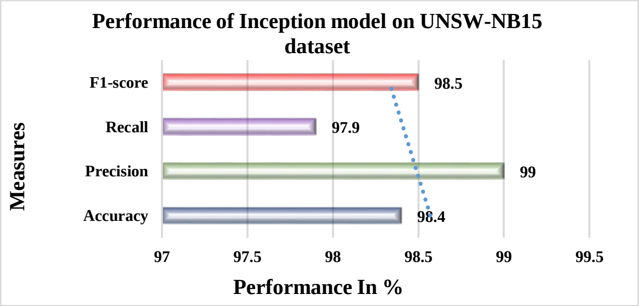

Fig. 5. Bar Graph for Inception Model

The above-following Figure 5 and Table II represent the Inception model performance on the UNSW-NB15 dataset. As shown in the graph, Inception achieves a high overall performance in classification tasks with an accuracy of 98.40%, precision of 99.00%, re-call of 97.90%, and F1-Score of 98.50% for intrusion and threat detection.

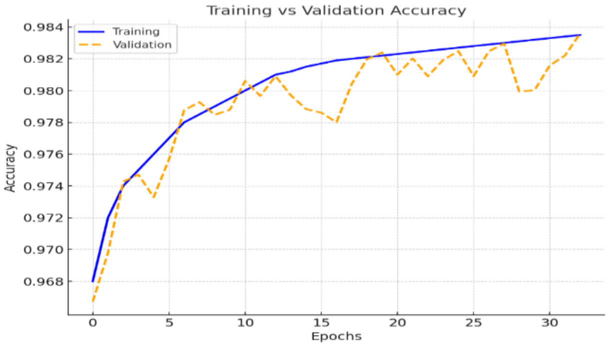

Fig. 6. Accuracy graph of Inception model

The line plot showing an Inception model's training and validation accuracy across 50 epochs is displayed in Figure 6 above. A y-axis shows the accuracy values, while an x-axis shows the number of epochs. The plot shows increasing training accuracy and fluctuating validation accuracy over 32 epochs, stabilizing near 98.2%, indicating learning progress with minor generalization variations.

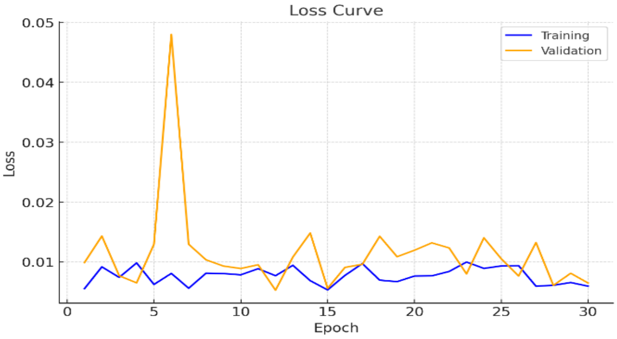

Fig. 7. Loss graph of Inception model

The loss curve graph in Figure 7 illustrates the training and validation loss over 30 epochs. The training loss (blue line) remains relatively stable with slight fluctuations, indicating consistent learning. The validity measure (orange line) displays a sudden increase at epoch 5 yet continues downward toward stability thereafter. The downward slope pattern of both lines demonstrates that learning has been successful while the slight validation loss changes point toward difficulties with generalization. The graphical display shows the model development process and its convergence characteristics in a simple manner.

A. *Comparative Analysis*

The comparative analysis for threat detection and prevention for different classification models on the UNSW-NB15dataset is provided in this section. The comparison provides between Inception proposed and existing RF, KNN and MLP models based on performance matrices like accuracy, precision, recall, and f1-score, shows in Table III.

TABLE III. Comparison between Inception and Existing model for threat detection and intrusion prevention

Measure	Inception	RF[28]	KNN [29]	MLP[30]
Accuracy	98.40	92.05	84.7	89.75
Precision	99	95.34	83.1	83.74
Recall	97.90	92.05	85.1	93.86
F1-score	98.50	93.04	82.2	88.51

The proposed Inception model gets compared to existing threat detection and intrusion prevention models RF, KNN, and MLP through this analysis presented in Table III. Inception model surpasses other evaluated models by achieving 98.40% accuracy which stands as the highest score and surpasses RF at 92.05% and KNN at 84.7% and MLP at 89.75%. Digital Inception Model provides a precision level of 99% while exceeding RF at 95.34% and KNN at 83.1% and MLP at 83.74%. Similarly, recall 97.90% remains higher than RF 92.05%, KNN 85.1%, and MLP 93.86%. The F1-score 98.50% of the Inception model demonstrates superior performance by surpassing RF 93.04%, KNN 82.2% along with MLP 88.51%, which indicates its balanced precision and recall configuration. The intrusion detection functions of the Inception model outperform conventional ML methods according to these assessment results.

The proposed Inception model provides superior features than conventional ML classifiers for detecting cyber threats. The DL structure of this model provides efficient processing of multiple feature layers which enables it to detect sophisticated network patterns with greater effectiveness than models including RF, KNN and MLP. Autonomous learning of intricate representations by the Inception model enhances its ability to adapt to changing cyber threats instead of requiring human-engineered features. High-dimensional data processing within its architecture enables the model to achieve better attack-type generalization. Additional validation of model robustness emerges from both comparative evaluation techniques followed by performance visualization results that prove its effective classification abilities.

V. Conclusion And Future Work

Cyber-attacks are growing and requirements of intrusion detection mechanisms are assumed to be robust, responsive and adaptive. In this context, deep learning models demonstrate the essence of implementing artificial intelligence related techniques in the modern cybersecurity framework to upgrade the real time threat detection systems. This paper employs various design multiple models like Inception, RF, KNN, and MLP on the UNSW-NB15 dataset after performing very robust preprocessing such as handling missing value, removal of duplicate, and feature scaling. Among these models, the Inception model achieved highest performance with 98.40% accuracy, 99.00% precision, 97.90% re-call and F1 score of 98.50%, higher than traditional classifiers. However, the model's results are strong, and it has also drawbacks in terms of suitability for deployment in the real world given the model's high computational complexity and its risk of overfitting to certain attack patterns. Future work will be on optimizing the computationally efficient, as well as real time adaptive learning for attacking threats, and as well as incorporating the explainable artificial intelligence techniques in cybersecurity applications.

References

[1] V. Kolluri, "An Extensive Investigation into Guardians of The Digital Realm: Ai-Driven Antivirus and Cyber Threat Intelligence," *TIJER - Int. Res. J.*, vol. 2, no. 1, 2015.

[2] A. H. Salem, S. M. Azzam, O. E. Emam, and A. A. Abohany, *Advancing cybersecurity: a comprehensive review of AI-driven detection techniques*, vol. 11, no. 1. Springer International Publishing, 2024. doi: 10.1186/s40537-024-00957-y.

[3] M. Gopalsamy, "AI-Driven Solutions for Detecting and Mitigating Cyber Threats on Social Media Networks," *Int. J. Adv. Res. Sci. Commun. Technol.*, vol. 3, no. 2, 2023.

[4] D. Kavitha and S. Thejas, "AI Enabled Threat Detection: Leveraging Artificial Intelligence for Advanced Security and Cyber Threat Mitigation," *IEEE Access*, vol. PP, p. 1, 2024, doi: 10.1109/ACCESS.2024.3493957.

[5] A. Khraisat, I. Gondal, P. Vamplew, and J. Kamruzzaman, "Survey of intrusion detection systems: techniques, datasets and challenges," *Cybersecurity*, 2019, doi: 10.1186/s42400-019-0038-7.

[6] R. K. Arora, Mohd.Muqeem, and M. Saxena, "Developing a Comprehensive Security Framework for Detecting and Mitigating IoT device Attack." Sep. 2024. doi: 10.21203/rs.3.rs-5165811/v1.

[7] V. Kolluri, "A Pioneering Approach To Forensic Insights: Utilization AI for Cybersecurity Incident Investigations," *Int. J. Res. Anal. Rev.*, vol. 3, no. 3, 2016.

[8] S. More, M. Idrissi, H. Mahmoud, and A. T. Asyhari, "Enhanced Intrusion Detection Systems Performance with UNSW-NB15 Data Analysis," *Algorithms*, vol. 17, no. 2, 2024, doi: 10.3390/a17020064.

[9] Z. Wang, "Artificial Intelligence in Cybersecurity Threat Detection," *Int. J. Comput. Sci. Inf. Technol.*, vol. 4, pp. 203–209, 2024, doi: 10.62051/ijcsit.v4n1.24.

[10] Mani Gopalsamy, "An Optimal Artificial Intelligence (AI) technique for cybersecurity threat detection in IoT Networks," *Int. J. Sci. Res. Arch.*, vol. 7, no. 2, pp. 661–671, Dec. 2022, doi: 10.30574/ijsra.2022.7.2.0235.

[11] A. K. Aftab Arif, Muhammad Ismaeel Khan, "An overview of cyber threats generated by AI," *Int. J. Multidiscip. Sci. Arts*, vol. 3, no. 4, pp. 67–76, 2024.

[12] M. I. Khan, A. Arif, and A. R. A. Khan, "AI-Driven Threat Detection: A Brief Overview of AI Techniques in Cybersecurity," *BIN Bull. Informatics*, vol. 2, no. 2, pp. 248–261, 2024.

[13] D. Chaudhary, S. K. Verma, V. Mohan Shrimal, R. Madala, R. Baliyan, and S. M, "AI-Based Methods to Detect and Counter Cyber Threats in Cloud Environments to Strengthen Cloud Security," in *2024 International Conference on Electrical Electronics and Computing Technologies (ICEECT)*, 2024, pp. 1–6. doi: 10.1109/ICEECT61758.2024.10739173.

[14] A. Sharma and H. Babbar, "Detecting Cyber Threats in Real-Time: A Supervised Learning Perspective on the CTU-13 Dataset," in *2024 5th International Conference for Emerging Technology (INCET)*, 2024, pp. 1–5. doi: 10.1109/INCET61516.2024.10593100.

[15] T. Rajendran, N. Mohamed Imtiaz, K. Jagadeesh, and B. Sampathkumar, "Cybersecurity Threat Detection Using Deep Learning and Anomaly Detection Techniques," in *2024 International Conference on Knowledge Engineering and Communication Systems (ICKECS)*, 2024, pp. 1–7. doi: 10.1109/ICKECS61492.2024.10617347.

[16] S. S. Gujar, "Optimizing Threat Mitigation in Critical Infrastructure through AI-Driven Cybersecurity Solutions," in *2024 Global Conference on Communications and Information Technologies (GCCIT)*, 2024, pp. 1–7. doi: 10.1109/GCCIT63234.2024.10862689.

[17] T. Almasri, M. A. Snober, and Q. A. Al-Haija, "IDPS-SDN-ML: An Intrusion Detection and Prevention System Using Software-Defined Networks and Machine Learning," in *APICS 2022 - 2022 1st International Conference on Smart Technology, Applied Informatics, and Engineering, Proceedings*, 2022. doi: 10.1109/APICS56469.2022.9918804.

[18] V. Atluri and J. Horne, "A Machine Learning based Threat Intelligence Framework for Industrial Control System Network Traffic Indicators of Compromise," in *SoutheastCon 2021*, 2021, pp. 1–5. doi: 10.1109/SoutheastCon45413.2021.9401809.

[19] U. Tekin and E. N. Yilmaz, "Obtaining Cyber Threat Intelligence Data from Twitter with Deep Learning Methods," in *ISMSIT 2021 - 5th International Symposium on Multidisciplinary Studies and Innovative Technologies, Proceedings*, 2021. doi: 10.1109/ISMSIT52890.2021.9604715.

[20] B. Boddu, "Ensuring Data Integrity and Privacy: A Guide for Database Administrators," *Int. J. Multidiscip. Res.*, vol. 4, no. 6, pp. 1–6, 2022.

[21] B. Boddu, "Scaling Data Processing with Amazon Redshift Dba Best Practices for Heavy Loads," vol. 7, no. 7, p. 5, 2023.

[22] S. M. Kasongo and Y. Sun, “Performance Analysis of Intrusion Detection Systems Using a Feature Selection Method on the UNSW-NB15 Dataset,” *J. Big Data*, 2020, doi: 10.1186/s40537-020-00379-6.

[23] R. Tandon, “Face mask detection model based on deep CNN techniques using AWS,” *Int. J. Eng. Res. Appl.*, vol. 13, no. 5, pp. 12–19, 2023.

[24] Y. H. Rajarshi Tarafdar, “Finding majority for integer elements,” *J. Comput. Sci. Coll.*, vol. 33, no. 5, pp. 187–191, 2018.

[25] K. Ullah *et al.*, “Short-Term Load Forecasting: A Comprehensive Review and Simulation Study With CNN-LSTM Hybrids Approach,” *IEEE Access*, vol. 12, no. July, pp. 111858–111881, 2024, doi: 10.1109/ACCESS.2024.3440631.

[26] B. Xia, D. Han, X. Yin, and N. Gao, “RICNN: A ResNet&Inception Convolutional Neural Network for Intrusion Detection of Abnormal Traffic,” *Comput. Sci. Inf. Syst.*, 2022, doi: 10.2298/CSIS210617055X.

[27] A. S. Dina, A. B. Siddique, and D. Manivannan, “A deep learning approach for intrusion detection in Internet of Things using focal loss function,” *Internet of Things (Netherlands)*, 2023, doi: 10.1016/j.iot.2023.100699.

[28] J. Song, X. Wang, M. He, and L. Jin, “CSK-CNN: Network Intrusion Detection Model Based on Two-Layer Convolution Neural Network for Handling Imbalanced Dataset,” *Inf.*, vol. 14, no. 2, 2023, doi: 10.3390/info14020130.

[29] H. A. Ahmed, A. Hameed, and N. Z. Bawany, “Network intrusion detection using oversampling technique and machine learning algorithms,” *PeerJ Comput. Sci.*, 2022, doi: 10.7717/PEERJ-CS.820.

[30] A. Deshmukh and K. Ravulakollu, “An Efficient CNN-Based Intrusion Detection System for IoT: Use Case Towards Cybersecurity,” *Technologies*, vol. 12, no. 10, p. 203, Oct. 2024, doi: 10.3390/technologies12100203.