Zero-Shot Based Hybrid Neural Network System for Enhancing Zero-Day Attack Detection

Kumar Saurabh[1], Uphar Singh[1,2*], Abhishek Mishra[1], Ranjana Vyas[1], O.P. Vyas[1]
[1]Department of Information Technology, Indian Institute of Information Technology Allahabad, Prayagraj, India
[2]School of Computer Science Engineering and Technology, Bennett University, Greater Noida, India
pwc2017001@iiita.ac.in, uphar.singh@bennett.edu.in, mse2022003@iiita.ac.in, ranjana@iiita.ac.in, opvyas@iiita.ac.in
*Corresponding Author: uphar.singh@bennett.edu.in

2024 IEEE 21st India Council International Conference (INDICON) | 979-8-3503-9128-2/24/$31.00 ©2024 IEEE | DOI: 10.1109/INDICON63790.2024.10958486

***Abstract*—Ensuring network security is increasingly vital as our digital interactions and network infrastructures expand. With the continuous evolution of cyber threats, particularly zero-day attacks that exploit unknown vulnerabilities, there is a pressing need for advanced intrusion detection systems (IDS). These zero-day vulnerabilities, undetected and unpatched, pose significant risks, especially in the burgeoning Internet of Things (IoT) landscape, which has seen a dramatic increase in such attacks. To address this challenge, we developed a hybrid deep learning model combining the strengths of CNNs and LSTMs, creating a robust framework for detecting zero-day attacks. Our model was trained on the CICIoT2023 dataset, a comprehensive dataset specifically designed for IoT security. The evaluation, conducted using a confusion matrix, demonstrated the model's efficacy in accurately identifying both known and unknown threats. The results indicate that the proposed hybrid deep learning approach significantly improves the detection of zero-day attacks, offering a promising solution for enhancing network security in an increasingly connected world. This work highlights the potential of advanced deep learning techniques in creating more effective IDS and enhancing the overall resilience of network infrastructures.**

***Index Terms*—Network security, Zero-day attacks, Intrusion detection systems, Convolutional Neural Networks, Long Short-Term Memory, Confusion matrix**

I. Introduction

The Internet is now an essential component of daily life. Dependence on the internet as a medium of passing information and knowledge has increased rapidly. However, the transfer of critical and sensitive information through the internet has made this medium more prone to network attacks. Any vulnerability or hole in the system will affect the whole network [3]. Traditional security mechanisms such as firewalls and encryption techniques encounter challenges as attackers continuously devise more sophisticated attacks [4].

Intrusion involves any risk to the confidentiality, integrity, or availability of information, including unauthorized access, changes, or destroying information systems. To detect harmful network activities and to support computer security Intrusion Detection System (IDS) along with Threat modeling [22] is suggested [5]. These systems help in detecting both known and unknown attacks and threats with high accuracy while maintaining a minimal false alarm rate is crucial. Traditional signature-based Network Intrusion Detection Systems (NIDSs) examine incoming network traffic for Indicators of Compromise (IOCs), which are attack signatures like source IP addresses, domain names, and hash values that may signify malicious activity [6]. However, the biggest challenge for these signature-based intrusion detection systems is to detect unseen or zero-day attacks. A zero-day attack is a type of cyberattack that exploits a software vulnerability that the developers or the public are unaware of. Since the flaw hasn't been discovered, there isn't a patch or fix available, making the software vulnerable to the attack. These attacks are especially risky because they can inflict serious harm before the issue is identified and addressed. The term "zero-day" indicates that developers have no time ("zero days") to fix the vulnerability once it becomes known [7] [8]. To ensure network security, an intrusion detection system must be both effective and intelligent in identifying and preventing known and unknown attacks.

Anomaly-based intrusion detection system, despite having a high false alarm rate, is capable of detecting both known and unknown attacks. Much research has been conducted to detect zero-day attacks or unseen attacks using Machine Learning models. Despite significant advancements, ML models have demonstrated considerable limitations in effectively detecting zero-day attacks. Traditional ML models rely heavily on historical data to learn patterns and classify new instances. However, zero-day attacks are by definition novel and previously unseen, meaning their patterns and signatures are absent from the training data. This reliance on historical data leads to a high incidence of false negatives, where new attacks go undetected because the model cannot recognize unfamiliar patterns [9].

To tackle this problem IDSs have taken advantage of Artificial Intelligence's capability to learn from the dataset without manually extracting features as it is done in machine learning systems. A deep learning system, utilizing neural networks, is capable of extracting features from a dataset and subsequently carrying out classification and detection tasks. By leveraging these capabilities, deep learning can enhance the model's detection accuracy beyond what traditional machine

978-8-3503-9128-2/24/$31.00 ©2024 IEEE

learning methods can achieve [10]. In this paper, we have proposed a zero-shot learning technique along with a hybrid neural network (CNN-LSTM) system to reduce the false alarm rate and increase zero-day attack detection. Zero-shot learning involves training a model in such a way that it can recognize and classify instances of classes that were not present in the training data [11]. Our paper uses CICIoT2023 to evaluate the proposed model. We compared the outcome of our zero-shot-based hybrid CNN-LSTM system to standalone CNN and ML-based Stacked Ensemble Learning. Our result shows that the hybrid model outperforms other models with high detection rate accuracy.

The structure of the paper is: Section 2 provides the previous work done in detecting zero-day attacks using ML and DL-based techniques. Section 3 will discuss our proposed methodology, dataset, the structure of our hybrid model along with various evaluation metrics. Section 4 will discuss the experimental setup, results, and analysis while section 5 concludes the paper.

II. Related Work

This section discusses key related papers that aim to detect zero-day attacks. Overall very few follow a zero-shot learning method to detect zero-day attacks. Zhang et al. [12] assessed the performance of machine learning-based Network Intrusion Detection Systems (NIDSs) in detecting zero-day attacks. They employed Zero-Shot Learning (ZSL) to create scenarios involving zero-day attacks. Their approach involved a sparse autoencoder model that maps features of known attacks into a semantic space, thereby enabling the detection of unknown attacks through feature-to-semantic mapping. The machine learning models differentiate between attack and benign classes by utilizing this mapping. The study used attacks from the NSL-KDD dataset, which dates back to 1998 and includes four attack scenarios, to simulate zero-day conditions. The findings reveal that the average accuracy across all attacks in the dataset is 88.3

In [13] Handy et al. proposed an autoencoder implementation for detecting zero-day attacks. The model is initialized using the optimal ANN architecture and then is evaluated using another algorithm where an attack instance is flagged as a zero-day attack if the Mean Squared Error (MSE) of the decoded (y) and the original instance (x) is larger than a given threshold. For evaluation, multiple thresholds are assessed: 0.05, 0.1, 0.15. The framework was evaluated using two modern NIDS datasets: CICIDS2017 and the NSL-KDD. The zero-day detection accuracy of this methodology for the CICIDS2017 dataset was 90.01% and for NSL-KDD it was 92.96%. However, the performance of this method is highly dependent on the chosen threshold. Using fixed thresholds (e.g., 0.05, 0.1, 0.15) can be inflexible and may not adapt well to varying network conditions or different types of attacks.

In [14] M. Sarhan introduced a zero-shot learning (ZSL) framework for Network Intrusion Detection Systems (NIDS). The proposed ZSL methodology addresses this by mapping network data features to semantic attributes that distinguish between known attacks and benign behavior. During the inference stage, the model constructs relationships between known and zero-day attacks, allowing it to detect previously unseen attacks as malicious. The framework was evaluated using two modern NIDS datasets: CICIDS2017 and the UNSW-NB15. The study introduces a novel evaluation metric called Zero-day Detection Rate (Z-DR) to measure the effectiveness of the model in detecting unknown attacks. The results indicate that ML-based NIDSs are ineffective at detecting certain zero-day attack groups, as evidenced by their low Z-DR.

Vigneswaran et al. in [15] conducted a comprehensive study comparing the effectiveness of shallow and deep neural networks in detecting network intrusions. Their research highlighted that deep learning models, particularly deep neural networks (DNNs), outperformed shallow networks in terms of accuracy and robustness. They demonstrated that DNNs could learn more complex patterns from data, making them better suited for identifying sophisticated and previously unseen attacks. A DNN with a 0.1 rate of learning was applied and run for 1000 epochs. The KDDCup-‘99 dataset was used for training and benchmarking the network.

In [16] Naseer et al. proposed an enhanced anomaly detection method using deep neural networks (DNNs). Their model aimed to increase precision and reduce false alarms in network anomaly detection. By leveraging a deep learning approach, the system can learn intricate patterns in both normal and abnormal network activities. These deep models were trained on NSLKDD training data set and evaluated on both test data sets provided by NSLKDD, namely NSLKDDTest+ and NSLKDDTest21. The study showcased high accuracy in anomaly detection and a low rate of false positives, underscoring the potential of DNNs to boost the efficiency of intrusion detection systems (IDS) based on anomaly detection.

Rustam et al. in [18] introduced an Intrusion Detection System (IDS) utilizing a residual feedforward neural network (ResNet) algorithm. This approach specifically addresses the vanishing gradient problem commonly encountered in deep learning models. The study showed that the ResNet-based IDS outperformed traditional feedforward neural networks in detecting various types of intrusions. The prominent NSL-KDD dataset was utilized for training and testing in this study. The accuracy obtained for two and five classes was 84.7 percent and 90.5 percent, respectively. Additionally, the identification speed was 15μs and 14μs, respectively, indicating that real-time detection is feasible.

In [19] Halbouni et al. present a novel approach for enhancing network intrusion detection systems (NIDS) through a hybrid deep neural network architecture combining Convolutional Neural Networks (CNN) and Long Short-Term Memory (LSTM) networks. This hybrid model aims to leverage the strengths of both CNNs and LSTMs to address the challenges of detecting network intrusions, especially in complex and dynamic environments. The model was evaluated using the UNSW-NB15, CIC-IDS2017, and WSN-DS datasets, all of which contained benign and attack records. With 5 epochs, they obtained 99.64%, 94.53%, and 99.67% accuracy for

binary classification using the CIC-IDS2017, UNSW-NB, and WSN-DS datasets. The model was unable to provide a high detection rate for certain types of attacks, such as web attacks in CIC-IDS2017 and worms, backdoors, and analysis in UNSW-NB15.

III. Methodology and Implementation Details

In Network Intrusion Detection Systems (NIDS) [21]/Network Intrusion Prevention Systems (NIPS) [20] using machine learning, the model is typically trained and evaluated based on established attack categories. Therefore, the evaluation primarily aims to gauge the model's efficacy in identifying instances from known attack classes as malicious.

The methodology proposed in this thesis involves evaluating the model during testing using samples from an attack class that was not part of its training stage, employing a zero-shot learning approach. The training set for our dataset is designed to include benign traffic as well as $n-1$ attack classes, intentionally leaving out one specific attack class designated as a "zero-day" attack. In contrast, the testing set contains all attack classes. This setup simulates real-world scenarios where the model encounters previously unseen attack types during testing, as it has not been trained on data from the excluded attack class [1]. Figure 1 below illustrates the scenario of zero-shot learning.

Fig. 1. Illustration of Training and Testing dataset.

This zero-shot learning is then integrated with a hybrid DL model. Figure 2 below showcases the methodology's framework:

A. Dataset

The first crucial task in developing a robust intrusion detection system is to choose a suitable dataset. The dataset must encompass normal and malicious records that accurately reflect the real-world scenarios the model will encounter. Our proposed methodology uses the CICIoT 2023 dataset which was developed by [2]. The dataset contains 33 attacks which are divided into 7 classes: Distributed Denial of Service (DDoS), Denial of Service (DoS), Mirai, Recon, Spoofing, Web-based & BruteForce. The number of rows for each class is shown in figure 3.

B. Preprocessing

1) **Data Encoding:** This process involves transforming the labels within the datasets. Because deep neural networks operate on numerical inputs and the labels are not naturally numerical, we employ a One-Hot Encoder. This encoder translates labels, like benign or malicious, into numerical forms appropriate for the neural network.
2) **Data Normalization:** This preprocessing technique is used to optimize data characteristics within a specific range. The differing means and standard deviations of the data extracted from the CSV file can influence the model's learning efficiency. To address this, we utilized the Standard Scaler to adjust the input data, with a mean equal to zero and a standard deviation equal to one.
3) **Data Splitting:** We divided our dataset as a training set and testing set. Distribution is such that 80% is used for training and 20% is used for testing. Furthermore, the training set was subdivided into training and validation subsets to fine-tune hyperparameters [18] and improve the model's performance.

C. Hybrid CNN-LSTM Model

CNNs are proficient at extracting spatial features, while LSTMs excel in capturing temporal characteristics. Leveraging CNN's capability to extract high-level features from large datasets, our model begins with a CNN layer. Initially, the data passes through the convolution layer, where filters identify key features to create a feature map. This feature map undergoes max pooling to retain the most significant features, and then batch normalization is performed. The resulting output is then given as input to an LSTM layer to capture temporal features. To prevent overfitting a dropout layer was added.

The architecture begins with a series of Conv1D layers, each applying filters to the input time-series data to extract features. The first Conv1D layer uses 32 filters of size 3 with the ReLU activation function, followed by a MaxPooling1D layer with a pool size of 2 to reduce dimensionality. This process is repeated with the second Conv1D layer using 64 filters and the third using 128 filters, each also followed by max pooling.

After feature extraction, the data is transformed into a one-dimensional array using a Flatten layer. This array is then passed into a Dense layer with 128 units, preparing the data for compatibility with the LSTM layers. The Reshape layer modifies the output to a 2D array that is appropriate for the LSTM layer, specifically with the dimensions (128, 1).

The LSTM layers then capture temporal dependencies in the data. The first LSTM layer has 64 units and returns the full sequence of outputs to the subsequent LSTM layer. The second LSTM layer, with 32 units, processes this sequence and outputs the final temporal features.

Finally, the model includes two Dense layers for classification. The first Dense layer has 64 units and utilizes the ReLU activation function. The last Dense layer, intended for multi-class classification, employs a softmax activation function to output the probabilities for each class. The model is compiled

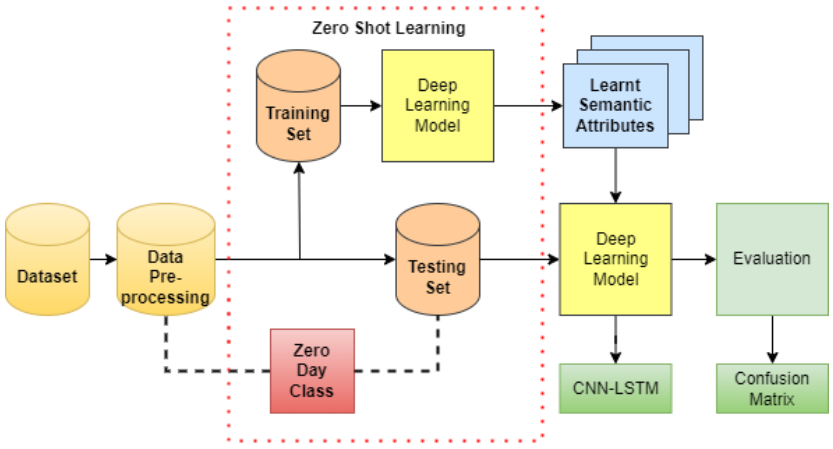

Fig. 2. Methodology Flowchart.

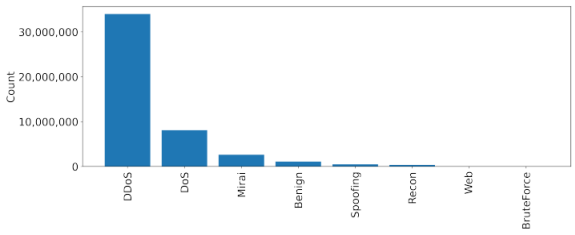

Fig. 3. Number of rows for each class.

with the Adam optimizer, loss function (categorical cross-entropy) with a learning rate of 0.001. The evaluation metric is accuracy. Callbacks, specifically ReduceLROnPlateau, are employed to adjust the learning rate based on validation loss. To make the model robust for detecting unseen attacks, it is trained along with validation after each epoch by combining the strengths of both CNNs and LSTMs.

D. Evaluation Metrics

The efficacy of a neural network is evaluated through a confusion matrix. In circumstances involving anomaly detection, data is commonly categorized as either negative (representing normal data) or positive (representing anomalous data) depending on the presence of an anomaly in the input. A binary confusion matrix is used to represent datasets that are labeled based on these criteria.

A confusion matrix is a table that shows the values for True Positive, False Negative, False Positive, and True Negative. These values are used to calculate metrics that measure the effectiveness of anomaly detection techniques. Those metrics are Detection Rate (DR), precision (P), recall (R), and F1 score (F1). Detection Rate is the percentage of correctly classified total attack samples in the test set [1].

$$DR = \frac{TP}{TP + FN} \tag{1}$$

$$P = \frac{TP}{TP + FP} \tag{2}$$

$$R = \frac{TP}{TP + FN} \tag{3}$$

$$F1 = 2 \cdot \frac{P \cdot R}{P + R} \tag{4}$$

IV. Experimental Setup, Results and Analysis

A. Experimental Setup

The proposed work trains a hybrid deep-learning model using a zero-shot learning approach. The experiment was executed on a machine equipped with 16 GB DDR6 RAM and NVIDIA GPU of 2560 CUDA cores. The experiment was conducted for 50 epochs.

B. Results and Analysis

In our research, we compared the performance of ML-based Stacked Ensemble learning, which utilizes MLP and Random Forest as the baseline classifier, with CNN and hybrid CNN-LSTM models on the CICIoT2023 dataset. The outcomes are presented in table I and II. Due to their significantly lower occurrence in the dataset, we disregarded the BruteForce attack class and the web class as the zero-day class.

TABLE I
Detection Rate Comparison of different Models

ZDA Class	Stacked EL	CNN	CNN-LSTM
DDoS	67.02%	94.91%	94.95%
DoS	48.95%	81.34%	85.23%
Recon	98.90%	99.60%	99.76%
Spoofing	94.00%	95.87%	95.88%
Mirai	76.43%	92.73%	94.98%

In table I the Zero-day Attack Class is abbreviated as ZDA Class, and Stacked Ensemble Learning is abbreviated as Stacked EL. We can see that the CNN-LSTM hybrid model consistently outperforms both the Stacked Ensemble Learning (Stacked EL) model and the standalone CNN model across most attack classes. For instance, in detecting DDoS attacks, the CNN-LSTM model achieves a detection of 94.95%, slightly higher than the CNN model at 94.91%, and substantially better than the Stacked EL model at 67.02%. Similarly,

for DoS attacks, the CNN-LSTM model shows a detection rate of 85.23%, outperforming the CNN model's 81.34% and the Stacked EL model's 48.95%. The trend continues for Recon, Spoofing, and Mirai as ZDA classes.

TABLE II
METRICS FOR DIFFERENT ATTACK CLASSES BASED ON CNN-LSTM MODEL

ZDA Class	Recall	Precision	F1 Score
DDoS	0.9496	0.9526	0.9484
DoS	0.8929	0.8023	0.8428
Recon	1.000	0.9959	0.9979
Spoofing	0.9600	0.9579	0.9589
Mirai	0.9592	0.9400	0.9495

From Table II, we can observe that the CNN-LSTM model demonstrates exceptional performance in detecting Recon attacks, achieving perfect Recall (1.000), high Precision (0.9959), and an outstanding F1 Score (0.9979). This indicates the model's remarkable capability in identifying all instances of Recon attacks with minimal false positives. Similarly, for DDoS, Spoofing, and Mirai attacks, the model achieves impressive results, showcasing its robust performance in effectively capturing and accurately classifying these attacks. However, the performance for DoS attacks, while still notable, is slightly lower, with a Recall of 0.8929, Precision of 0.8023, and an F1 Score of 0.8428. This suggests that although the model is quite effective in detecting DoS attacks, there is room for improvement, particularly in reducing false positives and increasing overall classification accuracy.

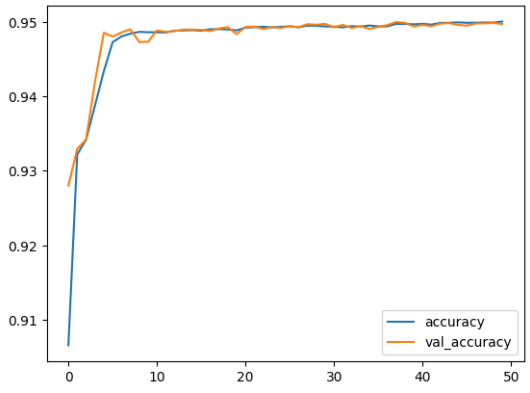

Fig. 4. Accuracy and Validation Accuracy of CNN-LSTM model while detecting DDoS attacks as ZDA.

Figure 4 illustrates the training and validation accuracy of the CNN-LSTM model for detecting DDoS attacks over 50 epochs. As observed, both training and validation accuracies increase rapidly during the initial epochs. The training accuracy begins at approximately 90% and swiftly rises, stabilizing around 95% after about 10 epochs. Similarly, the validation accuracy, which starts slightly higher than the training accuracy, follows a parallel trajectory and converges to a similar value of around 95%. The close alignment between the training and validation accuracy curves indicates that the model is learning effectively without significant overfitting.

Similarly, figure 5 illustrates the training and validation loss of the CNN-LSTM model for detecting DDoS attacks over 50 epochs. The graph demonstrates a rapid decrease in both training and validation loss during the initial epochs, indicating the model's quick adaptation to the training data. Initially, the training loss starts at approximately 0.24 and quickly declines, stabilizing around 0.10 after about 10 epochs. The validation loss follows a similar trajectory, starting slightly lower than the training loss and converging to a value of around 0.10. This behaviour indicates that the model has generalised well to the unseen data, maintaining a low error rate during validation.

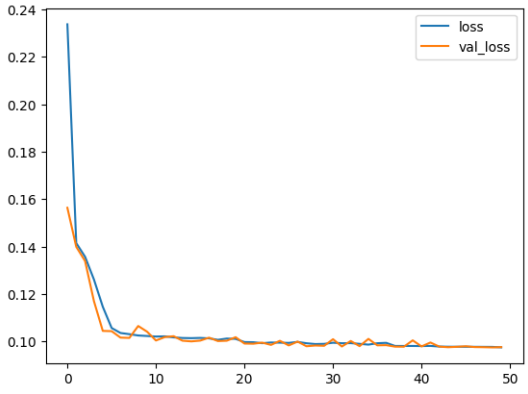

Fig. 5. Loss and Validation Loss of CNN-LSTM model while detecting DDoS attacks as ZDA.

The confusion matrix in figure 6 demonstrates that the model is particularly effective at identifying DDoS attacks. The high true positive rate for DDoS, coupled with relatively low misclassification rates, suggests that the model can effectively detect an unseen attack. However, there is still room for improvement in reducing the misclassification rates, particularly in distinguishing between DDoS and other types of denial-of-service attacks.

To further evaluate the model's performance, we generated the ROC curve and calculated the AUC for the DDoS attack class as ZDA. The ROC curve, shown in Figure 7, plots the true positive rate against the false positive rate across various threshold settings. The AUC for the DDoS class is 0.92, demonstrating a high probability that the model will correctly classify DDoS attacks as zero-day attacks. This high AUC value, along with the high detection rate observed in the confusion matrix, confirms the model's effectiveness in distinguishing DDoS attacks even under zero-day conditions.

V. CONCLUSION AND FUTURE SCOPE

In this proposed work a unique Hybrid Deep Neural Network System has been developed using zero-shot learning to detect previously undetected, known as zero-day attacks. The proposed approach involves training the model by leaving out one type of attack and then testing it with all types of attacks, simulating a scenario where a new, unknown attack occurs.

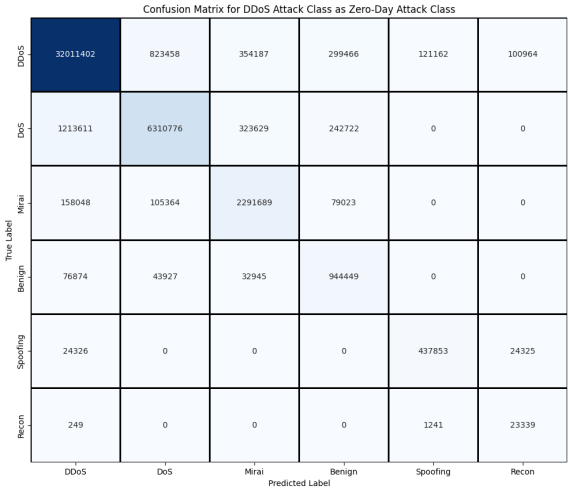

Fig. 6. Confusion Matrix for DDoS attack as Zero-Day Attack class.

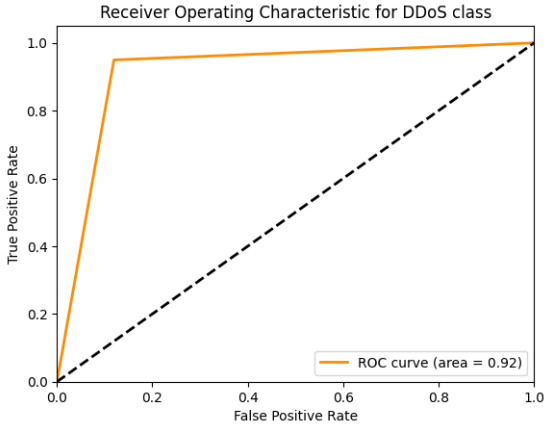

Fig. 7. RoC of DDoS attack as Zero-Day Attack class.

This system combines CNN and LSTM to accurately identify various attacks in the CICIoT 2023 dataset, including zero-day attacks. The results show that the hybrid model (CNN-LSTM) performs better in detecting zero-day attacks compared to standalone CNN and ensemble learning models. To improve the model's efficiency and accuracy, It is suggested to try different optimization algorithms and hyperparameter tuning methods. Additionally, exploring advanced feature engineering techniques could further enhance the model's ability to detect a wide range of zero-day attacks.

VI. Acknowledgment

The work is supported by the School of Computer Science Engineering and Technology, Bennett University, Greater Noida, India.

References

[1] Zhang, Z., Liu, Q., Qiu, S., Zhou, S., & Zhang, C. (2020). Unknown attack detection based on zero-shot learning. *IEEE Access*, *8*, 193981-193991.

[2] Neto, E. C. P., Dadkhah, S., Ferreira, R., Zohourian, A., Lu, R., & Ghorbani, A. A. (2023). CICIoT2023: A real-time dataset and benchmark for large-scale attacks in IoT environment. *Sensors*, *23*(13), 5941. MDPI.

[3] M. Almansor and K. Gan, "Intrusion detection systems: Principles and perspectives," J. Multidisciplinary Eng. Sci. Stud., vol. 4, no. 11, pp. 2458–2925, 2018.

[4] M. Almansor and K. Gan, "Intrusion detection systems: Principles and perspectives," J. Multidisciplinary Eng. Sci. Stud., vol. 4, no. 11, pp. 2458–2925, 2018.

[5] A. Khraisat, I. Gondal, P. Vamplew, and J. Kamruzzaman, "Survey of intrusion detection systems: Techniques, datasets and challenges," Cybersecurity, vol. 2, no. 1, pp. 1–22, 2019.

[6] V. Kumar and O.P. Sangwan, "Signature based intrusion detection system using snort," Int. J. Comput. Appl. Inf. Technol., vol. 1, no. 3, pp. 35–41, 2012.

[7] Saurabh, K., Sharma, V., Singh, U. et al. HMS-IDS: Threat Intelligence Integration for Zero-Day Exploits and Advanced Persistent Threats in IIoT. Arab J Sci Eng (2024). https://doi.org/10.1007/s13369024089355

[8] R. Ahmad, I. Alsmadi, W. Alhamdani, and L. Tawalbeh, "Zero-day attack detection: a systematic literature review," Artificial Intelligence Review, vol. 56, no. 10, pp. 10733–10811, 2023.

[9] D. I. Edeh, "Network intrusion detection system using deep learning technique," M.S. thesis, Dept. Comput., Univ. Turku, Turku, Finland, 2021.

[10] P. Sun, P. Liu, Q. Li, C. Liu, X. Lu, R. Hao, and J. Chen, "DL-IDS: Extracting features using CNN-LSTM hybrid network for intrusion detection system," *Secur. Commun. Netw.*, vol. 2020, Aug. 2020, Art. no. 8890306.

[11] Y. Xian, B. Schiele, and Z. Akata, "Zero-shot learning—the good, the bad and the ugly," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 4582–4591, 2017.

[12] Z. Zhang, Q. Liu, S. Qiu, S. Zhou, and C. Zhang, "Unknown attack detection based on zero-shot learning," *IEEE Access*, vol. 8, pp. 193981–193991, 2020.

[13] H. Hindy, R. Atkinson, C. Tachtatzis, J.-N. Colin, et al., "Utilising deep learning techniques for effective zero-day attack detection," *Electronics*, vol. 9, no. 10, pp. 1684, 2020.

[14] M. Sarhan, S. Layeghy, M. Gallagher, and M. Portmann, "From zero-shot machine learning to zero-day attack detection," *International Journal of Information Security*, vol. 22, no. 4, pp. 947–959, 2023. Springer.

[15] R. K. Vigneswaran, R. Vinayakumar, K. P. Soman, and P. Poornachandran, "Evaluating shallow and deep neural networks for network intrusion detection systems in cyber security," in *2018 9th International Conference on Computing, Communication and Networking Technologies (ICCCNT)*, pp. 1–6, 2018. IEEE.

[16] S. Naseer, Y. Saleem, S. Khalid, M. K. Bashir, J. Han, M. M. Iqbal, and K. Han, "Enhanced network anomaly detection based on deep neural networks," *IEEE Access*, vol. 6, pp. 48231–48246, 2018. IEEE.

[17] K. Ramli, N. Hayati, E. Ihsanto, T. S. Gunawan, A. H. Halbouni, et al., "Development of intrusion detection system using residual feedforward neural network algorithm," in *2021 4th International Seminar on Research of Information Technology and Intelligent Systems (ISRITI)*, pp. 539–543, 2021. IEEE

[18] U. Singh, S. Tamrakar, K. Saurabh, R. Vyas and O. P. Vyas, "Hyperparameter Tuning for LSTM and ARIMA Time Series Model: A Comparative Study," 2023 IEEE 4th Annual Flagship India Council International Subsections Conference (INDISCON), Mysore, India, 2023, pp. 1-6, doi: 10.1109/INDISCON58499.2023.10270325.

[19] A. Halbouni, T. S. Gunawan, M. H. Habaebi, M. Halbouni, M. Kartiwi, and R. Ahmad, "CNN-LSTM: hybrid deep neural network for network intrusion detection system," *IEEE Access*, vol. 10, pp. 99837–99849, 2022. IEEE.

[20] K. Saurabh, S. Singh, R. Vyas, O. P. Vyas and R. Khondoker, "MLAPS: A Machine Learning based Second Line of Defense for Attack Prevention in IoT Network," 2022 IEEE 19th India Council International Conference (INDICON), Kochi, India, 2022, pp. 1-6, doi: 10.1109/INDICON56171.2022.10039777.

[21] M. Matke, K. Saurabh and U. Singh, "An Empirical Evaluation of Machine Learning Algorithms for Intrusion Detection in IIoT Networks," 2023 IEEE 20th India Council International Conference (INDICON), Hyderabad, India, 2023, pp. 1353-1358, doi: 10.1109/INDICON59947.2023.10440779

[22] Saurabh, K., Gajjala, D., Kaipa, K. et al. TMAP: A Threat Modeling and Attack Path Analysis Framework for Industrial IoT Systems (A Case Study of IoM and IoP). Arab J Sci Eng 49, 13163–13183 (2024). https://doi.org/10.1007/s13369-023-08600-3