IJNRD.ORG ISSN : 2456-4184
INTERNATIONAL JOURNAL OF NOVEL RESEARCH AND DEVELOPMENT (IJNRD) | IJNRD.ORG
An International Open Access, Peer-reviewed, Refereed Journal

Intrusion Detection Scheme with Dimensionality Reduction in Next Generation Networks using DenseNet

[1]Vivek S V, [2]Kumaraswamy S,
[1]PG Student, [2]Assistant Professor,
[1,2]Department of Computer Science and Engineering,
[1]University Visvesvaraya College of Engineering Bangalore, India

Abstract: The project endeavors to bolster network security by harnessing the power of the UNSW_NB15 dataset to construct a sophisticated Intrusion Detection System (IDS) deploying DenseNet architecture. A meticulous approach to data preprocessing unfolds, entailing nuanced techniques such as label encoding and feature scaling. The model training phase adopts advanced methodologies, notably class balancing through weight mechanisms and the integration of bespoke metrics like F1-score for rigorous evaluation. Real-time insights are facilitated by the inclusion of live plotting tools during the training process. The overarching objective of this initiative is to make substantive contributions to the realm of cybersecurity. By instating a resilient and efficient IDS, the project aspires to adeptly identify and classify a spectrum of network intrusions, thereby augmenting the overall fortitude of networks against potential threats.

INTRODUCTION

In the dynamic realm of cutting-edge networks, such as the revolutionary 5G, where myriad devices interconnect through diverse technologies, the challenge of fortifying security becomes paramount. These sophisticated networks, although enabling unprecedented connectivity, also expose themselves to heightened security threats. The urgency to identify and respond to atypical network activities in real-time is underscored by the rapid pace and stringent demands of 5G applications. Traditional security approaches, once reliable, find themselves grappling with the intricacies of the 5G environment, revealing a misalignment with contemporary network designs. In response to this pressing need, our project spearheads a novel approach, leveraging advanced technologies and innovative strategies to enhance the security posture of these intricate networks.

This project focuses on boosting the security of advanced networks, like the cutting-edge 5G, which have many devices and diverse technologies. These networks face increased risks of security threats. Detecting unusual network behavior in real-time is crucial due to the high traffic rates and fast response needs of 5G applications. Traditional security methods struggle in the 5G environment and aren't well-suited for modern network designs. To tackle these challenges, the project introduces a smart security approach called an Intrusion Detection System (IDS) with dimensionality reduction. It uses machine learning techniques and a specific architecture called DenseNet to enhance the accuracy and efficiency of spotting intrusions in these advanced networks. The project uses a dataset called UNSW_NB15 for its development. It employs sophisticated techniques to prepare the data, like encoding labels and scaling features. During training, it employs advanced strategies such as class balancing and custom metrics like F1-score for thorough evaluation. Live plotting tools are also used for real-time insights during the training process. The ultimate goal is to contribute to the field of cybersecurity by creating a strong and effective system capable of precisely identifying various types of network intrusions. This, in turn, fortifies the resilience of networks, making them more robust against potential security threats. The motivation for real-time anomaly detection in 5G networks is rooted in the need for adaptive, proactive, and immediate responses to the dynamic challenges posed by the network's characteristics and the evolving threat landscape. Dynamic Nature of 5G Networks: 5G networks are characterized by their dynamic nature, with a multitude of devices constantly connecting and disconnecting. This dynamicity introduces an ever-shifting landscape, requiring security measures that can adapt in real-time.

High Traffic Rates and Low Latency: The hallmark features of 5G networks include exceptionally high data traffic rates and low-latency requirements. Applications relying on instantaneous data transmission demand a security infrastructure capable of keeping pace with this rapid exchange of information. Preventing Escalation of Attacks: Real-time anomaly detection acts as a proactive measure to prevent the escalation of potential security breaches. Timely identification of anomalies enables rapid mitigation, reducing the window of opportunity for attackers to exploit vulnerabilities.

RELATED WORKS

[1] The paper introduces an innovative approach to enhancing network intrusion detection systems (NIDS) by integrating Generative Adversarial Networks (GANs). It underscores the critical role of NIDS in fortifying computer networks against malicious activities while highlighting the limitations of traditional systems, particularly in detecting novel or previously unseen attacks. Leveraging the framework of GANs, which comprises two neural networks engaged in an adversarial game, the proposed method aims to bolster NIDS' ability to discern anomalies in network traffic. By training GAN models on normal network data to learn its distribution, the system can identify deviations indicative of potential intrusions. The paper likely includes experimental evaluations to validate the effectiveness of the proposed approach, assessing metrics such as detection accuracy, false positive rates, and detection time. Results and discussions are expected to delineate the performance gains achieved with the integration of GANs compared to conventional methods. [2] The conclusion likely summarizes the findings and suggests avenues for future research, potentially encompassing further optimization of the GAN architecture, exploration of diverse training strategies, or extension of the approach to address other types of cyber threats. Overall, the paper presents a promising paradigm shift in network intrusion detection, offering a potent solution to combat evolving cyber threats more effectively.

[3] The paper presents a novel approach to intrusion detection termed the Fusional Intrusion Detection Method (FIDM), which is built upon the Hierarchical Filtering and Progressive Detection Model (HF-PDM). The HF-PDM framework integrates hierarchical filtering techniques with progressive detection mechanisms to enhance the accuracy and efficiency of intrusion detection systems. FIDM extends this model by introducing a fusional mechanism that combines the outputs of multiple detection modules at different levels of abstraction. By fusing information from diverse sources, including network traffic analysis, system logs, and anomaly detection algorithms, FIDM aims to provide a more comprehensive and robust intrusion detection solution. [4] The proposed method likely includes experimental validations to demonstrate its effectiveness in detecting various types of intrusions while minimizing false positives and false negatives. Overall, the paper presents a promising advancement in intrusion detection technology by leveraging hierarchical and progressive detection strategies coupled with a fusion-based approach for improved performance and reliability.

[5] The paper addresses the critical challenge of detecting real-time malicious intrusions and attacks within IoT-powered cybersecurity infrastructures. It likely proposes novel techniques or algorithms tailored specifically for the unique characteristics of IoT environments, which typically involve a vast number of interconnected devices with limited computational resources and diverse communication protocols. The research likely emphasizes the importance of timely detection to mitigate the risks posed by cyber threats in IoT ecosystems, where vulnerabilities can lead to severe consequences including data breaches, privacy infringements, and service disruptions. The proposed methods may leverage machine learning, anomaly detection, or other advanced approaches to effectively identify malicious activities while minimizing false positives and negatives. [6] The paper presents a Network Intrusion Detection System (NIDS) tailored specifically for Building Automation and Control Systems (BACS). With the increasing adoption of interconnected devices in building infrastructure, BACS are vulnerable to cyber threats that can disrupt operations and compromise safety. The proposed NIDS is designed to monitor and analyses network traffic within BACS environments, aiming to detect and mitigate potential intrusions in real-time. It likely employs a combination of signature-based detection, anomaly detection, and machine learning techniques to identify malicious activities and abnormal behavior patterns indicative of cyber-attacks. [7] The research may involve the development of specialized algorithms or models trained on BACS-specific data to enhance detection accuracy and minimize false positives. The paper may include experimental validations to demonstrate the effectiveness of the NIDS in detecting various types of intrusions while operating within the resource-constrained environment typical of BACS deployments. Overall, the paper contributes to improving the cybersecurity posture of building automation systems by providing a dedicated NIDS solution tailored to the unique characteristics and requirements of BACS environments.

PROBLEM STATEMENT

In the era of advanced networks like 5G, the proliferation of devices and diverse technologies has heightened security concerns due to increased susceptibility to malicious activities. Traditional security measures struggle to keep pace with the dynamic nature of these networks, necessitating the development of innovative approaches to bolster network security.

This project aims to address the pressing need for real-time anomaly detection in advanced networks by proposing a sophisticated Intrusion Detection System (IDS) leveraging DenseNet architecture. The primary objective is to enhance the security posture of these intricate networks by adeptly identifying and classifying a spectrum of network intrusions, thereby fortifying networks against potential threats.

1. Security Vulnerabilities in Advanced Networks: Advanced networks, exemplified by 5G, face intricate security vulnerabilities due to extensive device connectivity. Existing security measures inadequately address the diverse threats, leaving networks exposed to potential intrusions.

2. Real-Time Anomaly Detection Complexity: The dynamic nature of advanced networks, with high traffic rates and low latency demands, presents significant hurdles for real-time anomaly detection. Current intrusion detection systems struggle to adapt, leading to delayed or inaccurate threat identification.

These challenges underscore the critical need for innovative intrusion detection solutions capable of swiftly identifying and categorizing network intrusions, thereby fortifying network security against evolving threats. Through the utilization of advanced architectures like DenseNet, this project endeavors to address these challenges head-on, paving the way for more robust and effective security measures in advanced network environments.

METHDOLOGY

Dataset preprocessing:

This dataset has nine types of attacks, namely, Fuzzers, Analysis, Backdoors, DoS, Exploits, Generic, Reconnaissance, Shellcode and Worms. The number of records in the training set is 175,341 records and the testing set is 82,332 records from the different types, attack and normal.

Utilize the UNSW_NB15 Dataset: Dataset Selection: The UNSW_NB15 dataset is chosen for both training and testing the intrusion detection system. This dataset is diverse and widely used for evaluating network intrusion detection methods.

Robust techniques are applied to handle missing values, ensuring a complete dataset free from irregularities. Numerical features are standardized to bring them to a uniform scale, preventing biases during model training. Encoding Categorical Variable is Categorical variables are encoded to numerical formats suitable for machine learning algorithms, a crucial step in preparing the dataset for intrusion detection. Strategies are employed to address class imbalances, ensuring that the dataset is well-suited for training. Techniques like class weight balancing contribute to fair representation of different intrusion types. Dimensionality Reduction is a fundamental technique in data analysis, aimed at reducing the number of features or variables in a dataset while preserving its essential information. These methods are widely used in various fields such as machine learning, data mining, signal processing, and image analysis. One common approach to dimensionality reduction is feature selection, involving selecting a subset of the original features based on their relevance to the task at hand.

Feature selection methods can be categorized into filter, wrapper, and embedded methods, each with its own strengths and weaknesses**.** SoftMax Activation for Multi-Class Classification: The SoftMax activation function is employed in the final layer for multi-class classification, assigning probabilities to different intrusion types.

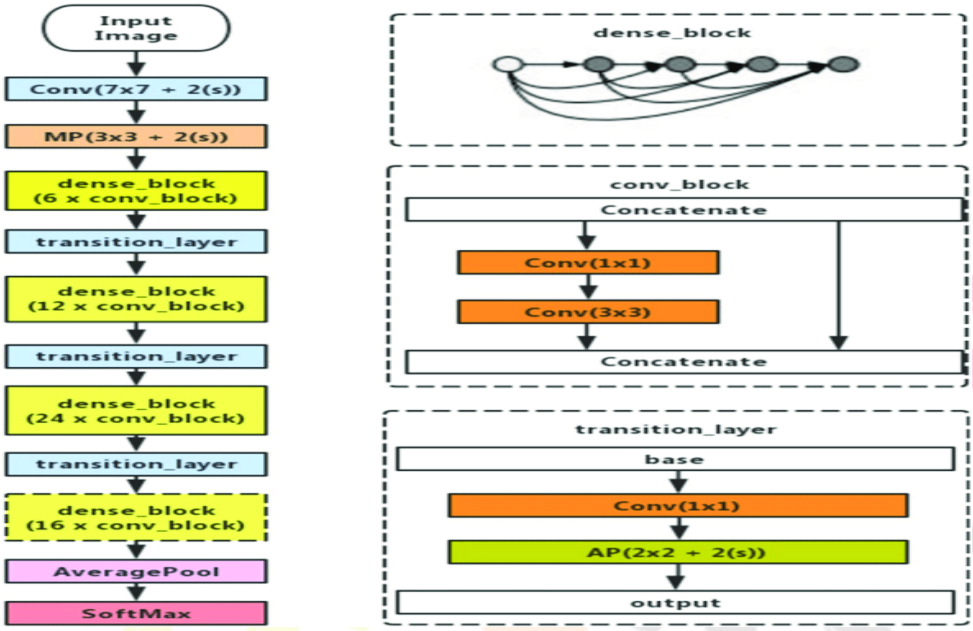

Figure 1: DenseNet121 Architecture

Model Training will split Dataset and Train and pre-processed dataset is divided into training and testing sets, ensuring an unbiased evaluation of the model's performance. Training with Hyperparameters in DenseNet model is trained on the training set with appropriate hyperparameters, optimizing its ability to learn from the data. Techniques like class weight balancing are employed during training to address imbalances in the distribution of intrusion types. Model performance is monitored using various evaluation metrics such as accuracy, precision, recall, and F1-score, ensuring a comprehensive understanding of its effectiveness.

IMPLEMENTATION

The model built will be trained for 20 epochs to make it learn the patterns in the image to detect the tumor. The model trained will be evaluated using the evaluation metricslike Accuracy, Loss, error-rate, Precision, Recall, F1-score, Accuracy-graph, Loss-loss and confusion-matrix. Analyzing a given model's performance is one of the most important steps in developing an effective ML model. A range of indicators, also referred to as performance metrics or evaluation metrics, are used to evaluate the effectiveness or quality of the model. We may use these performance markers to evaluate how well our model handles the supplied data. By changing the hyper-parameters, we may enhance the performance of the model. Performance measures allow for evaluation of a deep learning. Model's generalizability to fresh or unexplored data. The F1-score, recall, accuracy, precision, and AUC of the recommended model are assessed using the confusion chart. Assess the efficiency metrics using the confusion matrix. The model's output is seen in the confusion matrix. The primary metric used to assess how well the model predicts the exact both positive and negative events is accuracy. Categorization accuracy is typically discussed in terms of "accuracy". This is obtained by dividing the proportion of accurate predictions by the total number of inputs. Samples. Despite having a high classification accuracy, false positives might give the impression of a low accuracy. The difficulty is brought on by the greater possibility of misclassifying small class samples.

$$accuracy = \frac{TP + TN}{TP + FN + TN + FP}$$

Figure 2: Accuracy

The value is TP if the model correctly indicates that both the actual label as well as the picture are normal. The value is TN when the algorithm predicts an abnormal image and indeed the actual label are similarly abnormal. When the model predicts that the picture will be normal but the actual label will be incorrect, the value is FP. The value is FN when the algorithm predicts an aberrant picture

but the actual label is normal. The precision is the ratio of all correctly predicted positive observations to actual positive observations (PR). A model with a precision of 1 is considered to be satisfactory. Forecasting accuracy is determined by how many correct optimistic forecasts were generated. As a result, precision determines accuracy for the minority class. It may be determined by dividing the total number of properly predicted positive cases by the percentage of correctly detected positive models. Precision is the ratio of correctly predicted positive outcomes to all positive outcome.

$$\text{Precision} = \frac{TP}{TP + FP}$$

Figure 3: Precision

Recall (REC), also known as sensitivity, measures how well the classifier can find all positive samples. It is equivalent to the Precision metric and counts the percentage of true positives that were wrongly detected. Recall displays the percentage of genuinely positive values that were anticipated out of all positive values. It determines the proportion of positively anticipated cases that really occurred to positively observed cases in the recall.

$$\text{Recall} = \frac{TP}{TP + FN}$$

Figure 4: Recall

The recall is calculated using Equation 3. Recall evaluates a classifier's performance in relation to a false negative, in accordance with the definitions of precision and recall given above. Contrarily, precision provides information on how well a classifier performs in terms of false positives. As a result, accuracy should be as near to 100% as practical to reduce false positives and recall ought to be as close to 100% as practical to limit false negatives. Simply said, increasing accuracy will minimize FP errors while increasing recall will minimize FN error.

A binary classification model is evaluated using the F-score or F1 Score metric based on the predictions made for the positive class. Recall and Precision are used in the calculation. It is a particular kind of score that incorporates accuracy and memory. As a result, the harmonic means of recall and accuracy may be included, with each variable being given an equal amount of weight to calculate the F1 Score. The F1-score, which measures how well memory and accuracy are matched using equation 4, is the harmonic mean of recall and precision. Since memory and accuracy are both factors in the F1-score, it should be utilized when one of them (precision or recall) is somewhat more crucial to take into account than the other even if both are crucial for evaluation. The opposite can be true, for instance, if false positives are far more significant than false negatives.

$$\text{F1 Score} = \frac{2}{\frac{1}{\text{Precision}} + \frac{1}{\text{Recall}}} = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

Figure 5: F1 Score

AUC measures overall performance and evaluates it across all thresholds. AUC has a value between 0 and 1. Area under the curve (AUC) values range from 0.0 for a model with 100% incorrect predictions to 1.0 for a model with 100% correct predictions. It is a measure that is frequently used for binary classification, primarily. The likelihood that a classifier would consider a randomly selected positive example to be better than a negative example is measured by the AUC of the classifier. AUC should be used to determine how well the predictions are scored rather than focusing on the forecasts' absolute values. Additionally, it assesses the forecasting precision of the model without taking the categorization threshold into account. True positive rate: known as sensitivity at times. The percentage of positive data points that are properly recognized as positive for all positive data points is known as the true positive rate. True Negative Rate: similar to as specificity. A false negative rate is the percentage of negative data points that are incorrectly classified as negative when compared to all other negative data points. True Negative Rate: similar to as specificity. The percentage of negative data points that are deemed negative relative to all other negative data points is known as a false negative rate. False-positive Rate: The ratio of false negatives, or negative data points, to all other negative data points is known as the false negative rate. The ratio of true positives to false positives ranges from [0, 1]. The False Positive Rate Versus forms a curve called the AUC. True Positive Rate over all available data points, with a [0, 1] range. The model performs better at a higher value of AUC.

RESULTS

Convolutional Neural Network (CNN): CNN is a deep learning algorithm primarily used for processing and analyzing visual data, such as images and videos. It is a type of feedforward neural network that has proven to be highly effective in tasks like image classification, object detection, and image segmentation. Here's a brief overview of CNN consist of multiple layers, including convolutional layers, pooling layers, and fully connected layers. Convolutional layers are the core building blocks of CNNs, where convolution operations are applied to input images to extract spatial hierarchies of features. Convolutional layers use learnable filters (kernels) to convolve over the input image, extracting various features such as edges, textures, and patterns. These features are learned through the training process and become increasingly complex in deeper layers. Pooling layers are interspersed between convolutional layers to reduce the spatial dimensions of the feature maps while retaining the most important information. Common pooling operations include max-pooling and average-pooling. After feature extraction, the high-level features are flattened and passed through fully connected layers, also known as dense layers, for classification or regression tasks. These layers integrate the extracted features and make predictions based on them. Non-linear activation functions like ReLU (Rectified Linear Unit) are applied to introduce non-linearity into the model, enabling it to learn complex patterns and relationships in the data. DenseNet (Densely Connected Convolutional Networks) is a type of CNN architecture known for its dense connectivity pattern and efficient feature reuse. DenseNet introduces skip connections between all layers, allowing each layer to receive direct inputs from all preceding layers. DenseNet121 is a specific variant of DenseNet that comprises 121 layers, including convolutional, pooling, and dense layers. Here's a brief overview of DenseNet121, each layer is connected to every other layer in a feed-forward fashion. This dense connectivity facilitates feature reuse and gradient flow throughout the network, addressing the vanishing gradient problem and promoting feature propagation. DenseNet uses bottleneck layers, consisting of 1x1 convolutional filters, to reduce the number of input channels before the 3x3 convolutional layers.

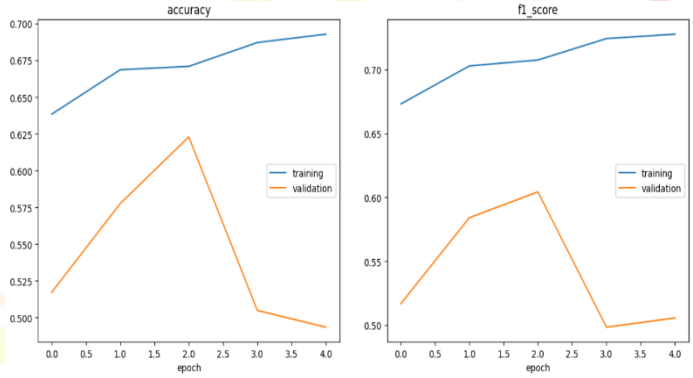

Fig 6: Accuracy and f1_score

This helps in reducing computational complexity while preserving important features. Skip connections in DenseNet concatenate feature maps from previous layers with the current layer's feature maps. This allows the network to access features at different levels of abstraction and enhances information flow. Transition layers are used in DenseNet to control the growth of feature maps and reduce the number of parameters in the network. They consist of a batch normalization layer, followed by a 1x1 convolutional layer and average pooling layer.

Efficient Feature Learning: DenseNet dense connectivity and feature reuse mechanism enable efficient feature learning, resulting in better parameter efficiency, improved gradient flow, and enhanced representation learning compared to traditional CNN architectures. Overall, DenseNet121 is a powerful CNN architecture that has demonstrated state-of-the-art performance in various computer vision tasks, including image classification, object detection, and image segmentation.

Specifies the number of training epochs, i.e., the number of times the entire training dataset will be passed forward and backward through the network. In this case, the model will be trained for epochs. Determines the number of samples processed before the model's parameters are updated. The dataset is divided into batches, and each batch containing 128 samples will be fed into the network during training. In this case, the model will be trained for epochs. Determines the number of samples processed before the model's parameters are updated. The dataset is divided into batches, and each batch containing 128 samples will be fed into the network during training.

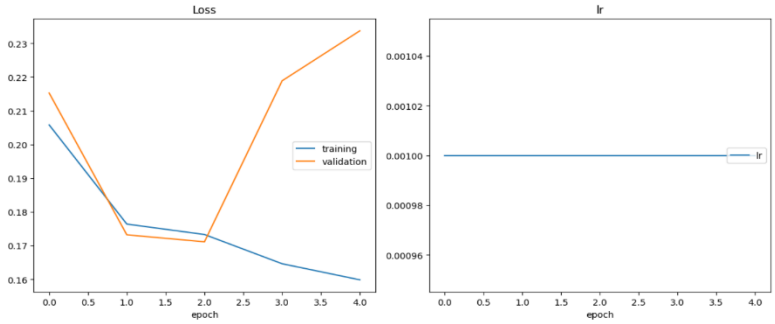

Fig 7: Loss and epoch

A tuple containing the validation data (X_test, y_test). This data is used to evaluate the model's performance on a separate dataset after each epoch during training. A list of callbacks to apply during training. Callbacks are functions that are executed at various stages of training, such as saving the best model, reducing learning rate, and early stopping.

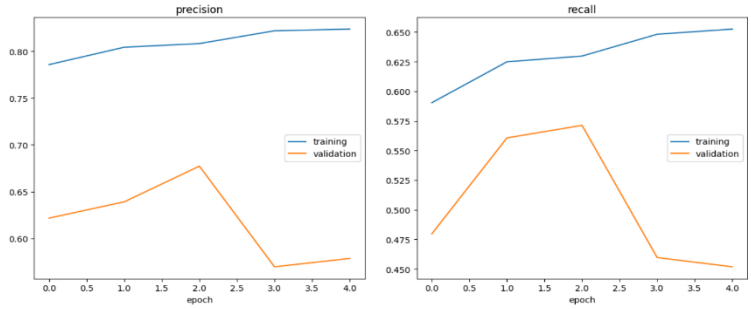

Figure 8: Precision and Recall

In this case, the callbacks include model checkpointing early stopping reducing learning rate on plateau and plotting losses. A dictionary specifying the class weights to be applied during training. Class weights are used to address class imbalance in the dataset by assigning higher weights to underrepresented classes. The weights are computed based on the distribution of classes in the training data. Controls the verbosity of the training process. A value of 1 indicates that progress bars will be displayed during training, providing information about the training and validation metrics for each epoch.

Overall, it trains a CNN model on the provided data, monitors its performance on the validation set, and applies various strategies such as model checkpointing, early stopping, and learning rate reduction to improve training efficiency and prevent overfitting.

CONCLUSION

The utilization of DenseNet, known for its ability to capture intricate patterns in network data, has contributed significantly to the system's effectiveness. The integration of dimensionality reduction techniques has not only enhanced computational efficiency but also demonstrated a delicate balance in retaining crucial information. This project underscores the importance of innovative approaches in addressing security challenges in next-generation networks. The amalgamation of machine learning, dimensionality reduction, and deep learning architectures positions this IDS as a foundation for further advancements in securing network infrastructures against evolving threats. Future work could focus on refining hyperparameters, exploring additional feature engineering strategies, or even considering alternative models. The project's significance lies in its contribution to the development of adaptive and responsive security systems, aligning with the evolving landscape of network security challenges.

REFERENCES

[1] K. Sood, K. K. Karmakar, V. Varadharajan, U. Tupakula, and S. Yu, “Analysis of policy-based security management system in software defined networks,” IEEE Commun. Lett., vol. 23, no. 4, pp. 612–615, Apr. 2019.

[2] R. Abdulhammed, M. Faezipour, A. Abuzneid, and A. AbuMallouh, “Deep and machine learning approaches for anomaly-based intrusion detection of imbalanced network traffic,” IEEE Sensors Lett., vol. 3, no. 1, pp. 1–4, Jan. 2019.

[3] S. P. Rao, S. Holtmanns, and T. Aura, “Threat modeling framework for mobile communication systems,” 2020, arXiv:2005.05110.

[4] J. Lam and R. Abbas, “Machine learning based anomaly detection for 5G networks,” 2020, arXiv:2003.03474.

[5] L. F. Maimó, A. H. Celdrán, M. G. Pérez, F. J. G. Clemente, and G. M. Pérez, “Dynamic management of a deep learning-based anomaly detection system for 5G networks,” J. Ambient Intell. Humanized Comput., vol. 10, no. 8, pp. 3083–3097, Aug. 2019.

[6] N. Khatri, S. Lee, A. Mateen, and S. Y. Nam, ‘‘Event message clustering algorithm for selection of majority message in VANETs,’’ IEEE Access, vol. 11, pp. 14621–14635, 2023.

[7] R. Shrestha, S. Y. Nam, R. Bajracharya, and S. Kim, ‘‘Evolution of V2X communication and integration of blockchain for security enhancements,’’ Electronics, vol. 9, no. 9, p. 1338, Aug. 2020.

[8] S. T. Mehedi, A. Anwar, Z. Rahman, and K. Ahmed, ‘‘Deep transfer learning-based intrusion detection system for electric vehicular networks,’’ Sensors, vol. 21, no. 14, p. 4736, Jul. 2021.

[9] R. Shrestha, R. Bajracharya, A. P. Shrestha, and S. Y. Nam, ‘‘A new type of blockchain for secure message exchange in VANET,’’ Digit. Communication Network, vol. 6, no. 2, pp. 177–186, May 2020.

[10] N. Sangeeta and S. Y. Nam, ‘‘Blockchain and interplanetary file system (IPFS)-based data storage system for vehicular networks with keyword search capability,’’ Electronics, vol. 12, no. 7, p. 1545, Mar. 2023.

[11] M. Hanselmann, T. Strauss, K. Dormann, and H. Ulmer, ‘‘CANet: An unsupervised intrusion detection system for high dimensional CAN bus data,’’ IEEE Access, vol. 8, pp. 58194–58205, 2020.

[12] A. Gazdag, S. Lestyán, M. Remeli, G. Ács, T. Holczer, and G. Biczók, ‘‘Privacy pitfalls of releasing in-vehicle network data,’’ Veh. Commun., vol. 39, Feb. 2023, Art. no. 100565.