**Anomaly Detection in Network Traffic: Using Hybrid Techniques**

**Dogiparthi Aasrith1, Kalla Bharath Vardhan2, Varanasi Sai Charan3, Bhavadharini R.M. 4**

1 School of Computer Science and Engineering, Vellore Institute of Technology, Chennai-600127

2 School of Computer Science and Engineering, Vellore Institute of Technology, Chennai-600127

3 School of Computer Science and Engineering, Vellore Institute of Technology, Chennai-600127

4 School of Computer Science and Engineering, Vellore Institute of Technology, Chennai-600127

|  |  |  |
| --- | --- | --- |
| **Article Info** |  | **ABSTRACT** |
| ***Article history:***  Received month dd, yyyy  Revised month dd, yyyy  Accepted month dd, yyyy |  | Network traffic deals with a lot of issues regarding data interception to ensuring confidentiality, integrity, and authenticity of communication. However, it possesses severe threats from malicious ones, such as data exfiltration, Distributed Denial-of-Service (DDoS) attacks, and command-and-control operations by cybercriminals. The inability to directly inspect payloads complicates anomaly detection, leaving security systems to rely on indirect features like packet metadata and flow patterns, which are often insufficient to comprehensively identify sophisticated threats. Traditional anomaly detection methods, such as deep packet inspection and rule-based systems, are increasingly rendered ineffective in encrypted environments. Deep packet inspection requires decryption, which is computationally expensive, violates privacy regulations, and introduces latency into the network. Similarly, rule-based systems rely heavily on predefined signatures and heuristics, which fail to adapt to emerging threats or detect zero-day anomalies. These approaches create significant gaps in security, especially as network traffic continues to grow in volume and complexity across various industries. The proposed hybrid model of Artificial Neural Network (ANN) and Convolutional Neural Network (CNN) model achieved 98.34% accuracy, with precision of 92.5%, recall of 96.5% and F1-score of 95.5% respectively. While ANN model resulted with 96.82% accuracy, and CNN model showed 94.36%. Comparatively, the hybrid system results in more accuracy and is more efficient than ANN and CNN models. |
| ***Keywords:***  Artificial Neural Network (ANN)  Convolutional Neural Network (CNN)  Anomaly  Hybrid Model |
| *This is an open access article under the* [*CC BY-SA*](https://creativecommons.org/licenses/by-sa/4.0/) *license.* |
| ***Corresponding Author:***  R.M. Bhavadharini  School of Computer Science and Engineering, Vellore Institute of Technology, Chennai-600127  Email: bhavadharini.rm@vit.ac.in | | |

1. **INTRODUCTION**

The widespread adoption of communication protocols, such as Internet Control Message Protocol (ICMP), User Datagram Protocol (UDP) and Transmission Control Protocol (TCP), has significantly enhanced data security and privacy in digital networks. These protocols ensure that sensitive information remains confidential, making them integral to modern communication systems. Advanced machine learning techniques have emerged as promising solutions, leveraging metadata and statistical features to detect deviations indicative of malicious activities. However, existing models often face limitations such as high computational costs, overfitting on imbalanced datasets, and difficulty adapting to evolving threats [1]. These limitations underscore the need for robust, scalable, and privacy-preserving solutions capable of maintaining high detection accuracy without decrypting the traffic. The challenge is further exacerbated by the need to balance privacy preservation with effective threat detection. While network traffic protects user's sensitive data, it also hinders the ability to monitor network activity, creating a blind spot for security solutions [4]. This trade-off necessitates the development of innovative approaches that can accurately detect anomalies without decrypting traffic. Achieving this balance requires a paradigm shift from traditional techniques to advanced machine learning-based models capable of leveraging network traffic metadata to identify malicious behaviors with minimal computational overhead and high accuracy.

This study proposes a hybrid anomaly detection system that combines ANNs and CNNs with custom enhancements to address these challenges. By incorporating advanced activation functions, such as Swish and GELU, and regularization techniques tailored to mitigate overfitting, the model achieves superior generalization and performance. The hybrid architecture, designed to optimize feature learning and classification accuracy, demonstrates significant potential for anomaly detection in network traffic. This approach enhances detection capabilities, allowing it a viable solution for modern cybersecurity landscapes.

1. **LITERATURE REVIEW**

Anomaly detection methods that rely on inspecting packet contents become less effective since encrypted network traffic usage steadily increases. Wu, Z et al. [1] presents an extensive analysis of advanced methodologies which solve the problems of encrypted traffic anomaly detection. The semi-supervised anomaly detection framework Poison-Resistant Anomaly Detection (PRAD operates to defend against poisoning attacks as well as reduce human labeling mistakes. Xing, J. et al.,[2] performs an online clustering approach forms the basis of anomaly detection to increase detection reliability. The deep dictionary learning with Long short-term memory (LSTM)-based autoencoder forms the basis of D2LAD which operates as a completely unsupervised online anomaly detection system. D2LAD shows high performance across real-world benchmarks through experiments which prove its ability to successfully detect anomalies with low resource requirements. In Behdadnia, T. et al.,[3] the detection method produced 93% successful results in abnormal traffic identification which demonstrates its usefulness for anomaly detection systems in power grid communication networks. Yu, T. et al.,[4] and Lv, G. et al.,[5] discussed about the combination of methods strengthens critical infrastructure cybersecurity because attacks get discovered at dawn. Through sandbox-based process data collection users can identify standard traffics along with their malicious counterparts. Labeling features run sequentially through multiple AE layers that creates expanded dimensional vectors to enhance classification quality. Experimental results demonstrate that the multi-AE system outperforms traditional models by reaching better accuracy rates with lower loss numbers. Kopp, M. et al.,[6] and Han, S. et al.,[7] also investigates dataset imbalance effects which show that unbalanced conditions reduce positive detection success. A new classification system uses a two-step randomized forest procedural approach which features modified random forest elements. The evaluation with ISCX VPN-NonVPN data demonstrated that multi-AE achieves better classification results than standard approaches by showing 5% increased accuracy since decision trees and KNN and general random forest algorithms were tested. By integrating this approach, the precision of traffic classification gets better during the time encrypted traffic prevails over the Internet.

Shen, M. et al.,[8] and Meghdouri, F. et al.,[9] discussed the analytical method proposed an unsupervised procedure to detect zero-day DDoS attacks against IoT networks through random projection and ensemble models of K-means, GMM (Gaussian Mixture Model), and one-class SVM which reached 94.55% accuracy. The authors applied deep convolutional autoencoder (MR-DCAE) using manifold regularization to detect unauthorized broadcasting through their model for feature compression autoencoders in research. In terms of machine learning framework, Wang, Z. et al.,[10] and Zhu, Z. et al.,[11] discussed a generic FL-based anomaly detection framework for energy theft detection, where each user used their private electricity usage data to train local deep autoencoder-based detectors. The authors Ucci, D. et al.,[12] and Kim, M. G. et al.,[13] report numerous academic investigations that took place in malware attack detection together with network traffic anomaly detection and related areas. An anomaly detection research team developed a reconstruction error-based system utilizing an autoencoder (AE) for packet metadata analysis without particular node information and demonstrated superior performance over conventional approaches with improved accuracy and AUC score and precision rates and F1 scores. Ashraf, J. et al.,[14] and Long, G. et al.,[15] studies provided a complete review of machine learning-based models for detecting encrypted malicious traffic is provided, together with a framework for systematic debate and analysis of these models. Multiple machine learning algorithms such as decision trees, random forests and gradient boosting and SVM and neural networks operate in identity resolution to enhance data quality and automatic resolution processes and dynamic data management. The field of encrypted traffic detection and sophisticated cyber-attack modeling depends on advanced detection methods alongside machine learning models according to two studies discussing this issue. Pratiwi, M. et al.,[16] discussed about network security research on machine learning techniques highlights the growing concern about adversarial attacks, especially ransomware attacks during model training, which can alter training data to impair model performance. Researchers have examined the effects of these attacks on various machine learning models and classified them according to their goals. Tian, Z. et al.,[17] discussed that strong training mechanisms, anomaly detection in training data, and safe aggregation methods for federated learning are some of the countermeasures that have been suggested for poisoning attacks. Anderson, B. et al.,[18] discussed, in certain situations, supervised machine learning models that have been trained on huge datasets have demonstrated promise in reaching 0% false discovery rates. Shekhawat, A. S. et al.,[19] and Velan, P. et al.,[20] discussed that, feature selection is now being automated using machine learning models to identify subtle patterns indicative of malicious activity. Payload-based and feature-based categorization algorithms are distinguished in surveys; the latter is becoming more popular because it is independent of protocols. The trade-offs between accuracy, processing cost, and adaptability to shifting network environments are highlighted in thorough studies.

Rezaei, S. et al.,[21] and Wang, J. et al.,[22] discussed that, Traffic classification has developed over the course of two decades and has been instrumental in applications including Quality of Service (QoS) provisioning and billing, as well as security enforcement in firewalls and intrusion detection systems. The traditional approaches like port-based analysis, packet inspection, and machine learning techniques have found it difficult to deal with the growing dominance of encrypted traffic, paving the way for moving toward deep learning-based methods. Recent work of Wingarz, T. et al.,[23] emphasizes deep learning models such as CNNs, LSTM, and Gated Recurrent Units (GRUs) as the potential solutions to anomaly detection from encrypted traffic due to their potential to learn complex patterns without human intervention in selecting features. Experiments in Bakhshi, T. et al.,[24] and Lee, I. et al.,[25] have also been performed with hybrid architectures, and CNN-GRU has proved superior to other hybrids by combining effective feature extraction and incremental learning of temporal patterns. Models such as Deep Encrypted Traffic Detection (DETD) and autoencoder-based methods have also shown significant improvement in the accuracy of anomaly detection and effectiveness of feature extraction. Moreover, incremental machine learning methods like SVM with Stochastic Gradient Descent (SGD) have been introduced for encrypted malware detection, with adaptability via periodic updates. These developments underscore the increasing role of deep learning and adaptive techniques of machine learning in confronting the difficulties created by encrypted traffic in today's cybersecurity and network management. Table 1 shows the comparison of the referenced results.

Table 1. Comprasion Of Results Achieved Various Models

|  |  |  |  |
| --- | --- | --- | --- |
| **S.NO** | **Authors** | **Models** | **Results** |
| 1 | Wu, Z., .et al (2024) [1] | Variational Recurrent Neural Network | Accuracy-85.2% |
| 2 | Xing, J., et al (2020) [2] | LSTM-based Autoencoder | Accuracy-94.5% |
| 3 | Behdadnia, T., et al (2023) [3] | CNN | Accuracy-93% |
| 4 | Yu, T., .et al (2019) [4] | Autoencoder | Accuracy-96.3% |
| 5 | Lv, G., et al. (2020) [5] | Decision tree, KNN, Random Forest | Accuracy  Decision tree-71%,  KNN-67.47%, Random Forset-73.46% |
| 6 | Kopp, M., et al (2018) [6] | Clustering | Threshold-95% |
| 7 | Han, S., et al (2022) [7] | K Means, SVM, XGBoost, LSTM | Accuracy  K Means-93%, SVM-88%, XGBoost-88%, LSTM-88% |
| 8 | Shen, M., et al (2022) [8] | Decision Tree, Random Forest, LSTM | Accuracy  Decision Tree-97%, Random Forest-96%, LSTM-98% |
| 9 | Meghdouri, F., et al (2020) [9] | Random Forest | Accuracy  CICIDS2017-99%, UNSW-NB15-99.4%,  ISCX-Bot-2014-99.7% |
| 10 | Wang, Z., et al (2022) [10] | Convolutional Autoencoders | Accuracy-87% |
| 11 | Zhu, Z., et al (2022) [11] | Random Forest, CNN, LeNet-5 | Recall  Random Forest-85%, CNN-100%, LeNet-5-99.49% |
| 12 | Pratiwi, M. et al (2024) [16] | LSTM | Accuracy-99.37% |
| 13 | Wingarz, T., et al (2024) [23] | ANN, SVM | Accuracy  ANN-82%, SVM-91% |
| 14 | Wang, J., et al (2022) [22] | FS-Net, Single Node method, BERT | Accuracy  FS-Net-84.2%, Single Node method-71.4%, BERT-96% |
| 15 | Anderson, B., et al (2016) [18] | l1-logistic  regression | Accuracy-93.978% |
| 16 | Shekhawat, A. S., et al (2019) [19] | SVM, Random Forest, XGBoost | Accuracy  SVM-92%, Random Forest- 96.80%, XGBoost- 97% |
| 17 | Bakhshi, T., et al (2021) [24] | Hybrid: CNN+GRU | Accuracy  NSL-KDD-93.10%  NB15-91.21%,  CIC-17-90.17% |
| 18 | Long, G., et al (2023) [15] | Conventional Stacked Autoencoder | Accuracy-96.98 |
| 19 | Kim, M. G., et al (2024) [13] | IsolationForest, OneClassSVM, Auto Encoder | Accuracy  IsolationForest-85.4%, OneClassSVM-73.9%, Auto Encoder-85.6% |
| 20 | Lee, I., Roh, H., et al (2021) [25] | Stochastic Gradient Descent, Passive Aggressive, Gaussian Naive Bayes | Accuracy  Stochastic Gradient Descent-94.4%, Passive Aggressive-86.6%, Gaussian Naive Bayes-86% |

1. **PROPOSED METHODOLOGY**

**3.1. Dataset Description**

The dataset, Network Intrusion Detection [26] includes different simulated intrusions that were conducted in a military network environment. The LAN was subjected to a simulated real-world scenario and faced various attacks. A connection is a series of TCP packets that start and end at a specific time interval, during which data is transmitted from a source IP address to a target IP address using a well-defined protocol. Furthermore, each link is categorized as either normal or an attack, with only one distinct type of attack. Each link record is approximately 100 kilobytes in size. Table 2 displays the total number of records in the dataset, which is 25212. The data was divided into 80 training and 20 testing sets.

The normal and attack data consist of 41 parameters, both quantitative and qualitative, for each TCP/IP connection (3 qualitative and 38 quantitative). The class variable has two categories as shown in Figure 1:

* Normal [53.4%]
* Anomaly [46.6%]

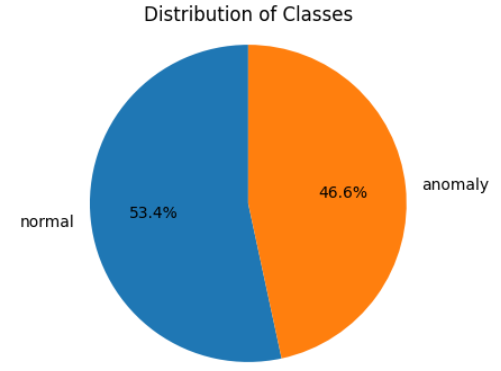
****

Figure 1. Class Distribution

Table 2. Dataset Description

|  |  |  |
| --- | --- | --- |
| **Data Split** | **Classes** | **Count** |
| Train | Normal | 10780 |
|  | Anomaly | 9395 |
|  |  |  |
| Test | Normal | 2689 |
|  | Anomaly | 2348 |
| Total |  | 25212 |

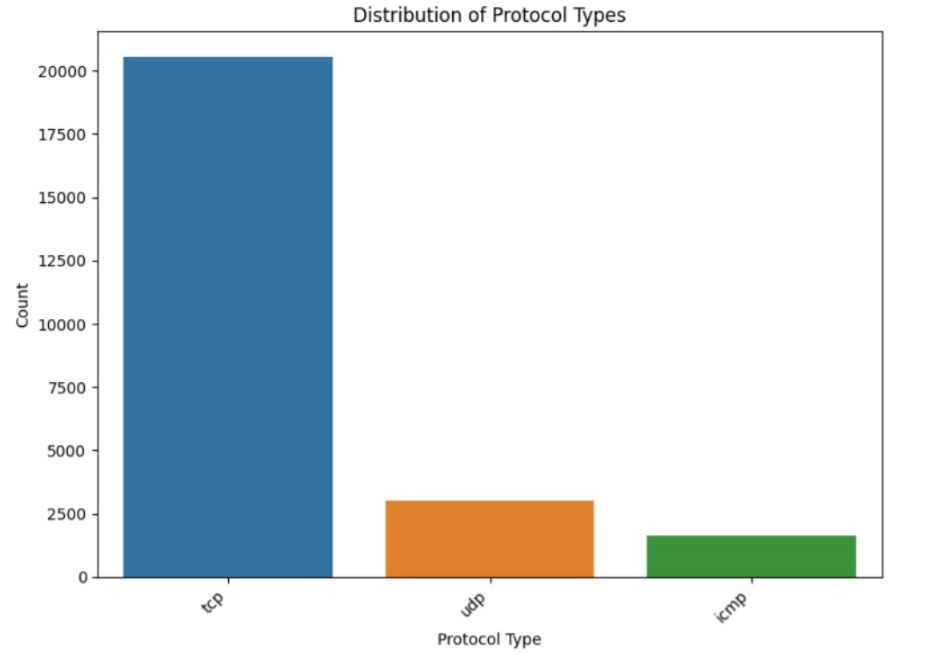


Figure 2. Various protocols in dataset

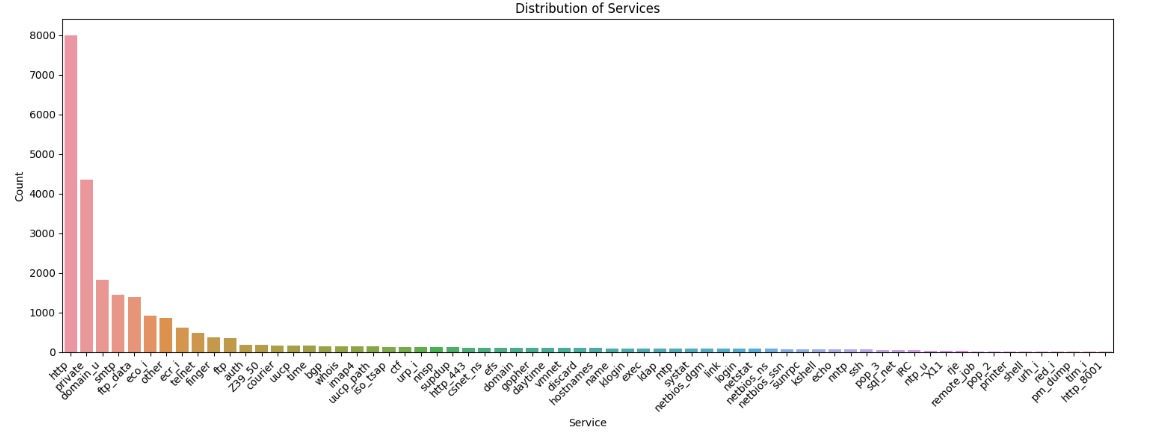


Figure 3. Services in dataset

Figure 2, shows all the protocols mentioned in the dataset and Figure 3 shows us all the services that are considered in the dataset.

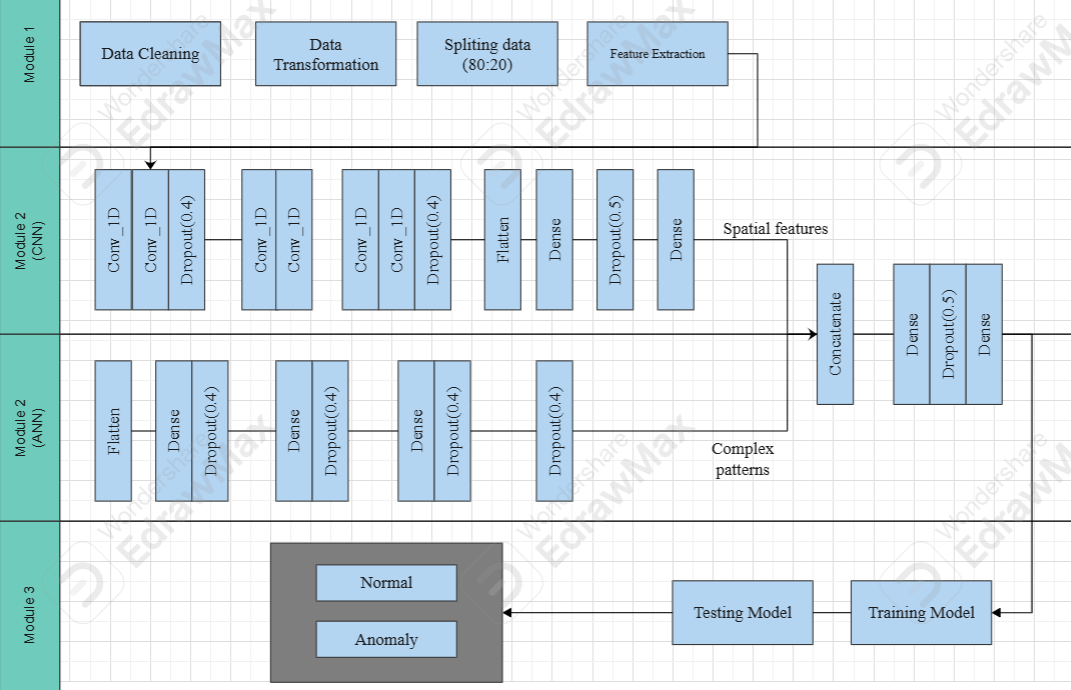


Figure 4. Architecture Diagram

The architecture (Figure 4). shows how hybrid deep learning model employs CNNs together with ANNs for classification purposes. Module-1 (Data Preprocessing) starts the process by performing data cleaning then splitting and extracting features to make the dataset ready. The preprocessed dataset enters Module-2 for training of CNN and ANN alongside each other as part of the proposed hybrid system. The model evaluation takes place during Module-3 after its training phase as part of the testing process. The proposed system produces outputs to identify instances either as "Anomaly" or "Normal."

**3.2. Data Preprocessing**

This module is responsible for cleaning and preparing raw network traffic data from dataset: Network Intrusion Detection [26] for effective model training. It involves handling values which are missing, and performing feature extraction to ensure that the dataset is optimized for analysis. Performed label-enconding, for protocol\_type, service, flag and class columns to change categorical values to numerical values, and smote technique for class imbalance as shown in Fig.5. The actual data count was normal:13449 and anomaly:11743, now we made both of them equal as 13449. These steps enhance the quality and relevance of the input data, which is critical for training a robust ANN. Techniques like data splitting further enable the separation of training and testing datasets, in 80:20 ratio as shown in Table 2, to ensure unbiased evaluation.

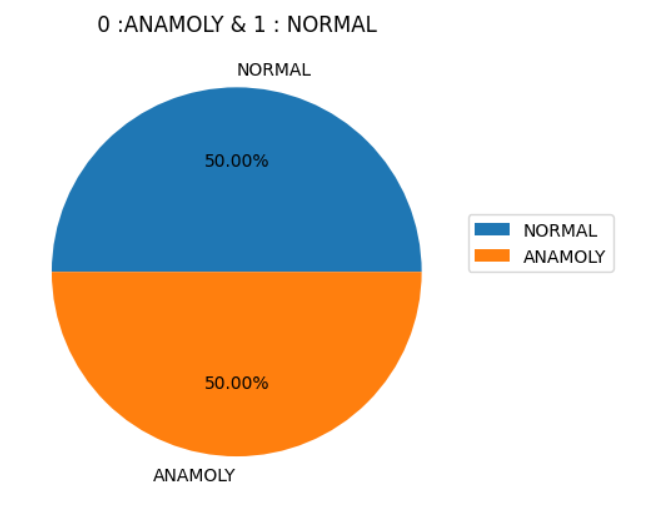


Figure 5. Balanced data using SMOTE technique

**3.3. Convolutional Neural Network (CNN) Architecture**

The proposed CNN model is designed to efficiently detect anomalies in network traffic data by leveraging the powerful feature extraction capabilities of convolutional layers. The model begins with multiple 1D convolutional layers, (1), each using 32 filters with a kernel size of 3 and RELU activation function, (2), which helps in capturing local patterns and relationships within the sequential input features of shape (41, 1). Dropout layers, (3) with a rate of 40% are introduced after every two convolutional layers to prevent overfitting and improve model generalization. Furthermore, additional convolutional layers with 64 filters are incorporated to capture more complex patterns. After feature extraction, a Flatten layer is used to convert the multi-dimensional output into a 1D vector, which is then passed to a dense layer with 256 neurons and RELU activation for high-level feature learning. A final dropout layer with a higher rate of 50% further enhances robustness. The output layer consists of two neurons with softmax activation, effectively classifying the input data into two categories: normal or malicious. This refined CNN architecture combines convolutional feature learning with regularization techniques, ensuring accurate and efficient anomaly detection. The model summary is shown in Figure 6.

**Convolutional Layer (Conv1D, 32 Filters, ReLU)**

(1)

where:

* X is the **input vector** of shape (41,1).
* ​is the **kernel (weight) matrix** of shape (3,1,32).
* k=3 is the kernel size.
* ​ is the bias vector
* is the **pre-activation output** of the convolution.
* are the input values covered by the filter.
* The filter slides along the **time/feature dimension**.

The **ReLU activation** is applied:

(2)

Where is the **activated output**.

**Dropout Layer (40%)**

(3)

where:

* ​ is a **binary dropout mask** (with probability 0.4 for dropping neurons)
* ⊙ represents **element-wise multiplication**.

**Loss Function (Binary Crossentropy)**

(4)

where:

* ​ is the **true label** (one-hot encoded)
* ​​ is the **predicted probability**.

**Adam Optimization Update Rule**

(5)

where:

* and are the first and second moment estimates
* η is the learning rate (0.001)
* ϵ prevents division by zero.

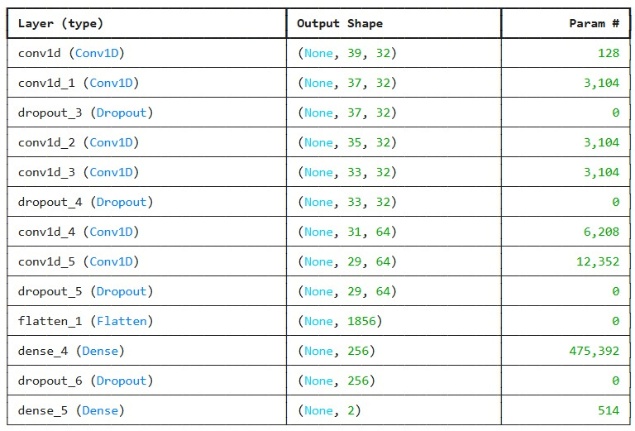


Figure 6. CNN Model Summary

**3.4. Artificial Neural Network (ANN) Architecture**

The proposed ANN model is intended exclusively for binary classification tasks, with a focus on anomaly detection in network traffic data. The model employs a sequential architecture with numerous dense layers and innovative activation functions like Swish, (8) and GELU, (9), which serve to increase non-linear learning capabilities and convergence. The input layer, (6) accepts a feature vector of shape (41, 1), representing extracted network features, and flattens it for processing. To reduce overfitting and ensure robust generalization, three hidden layers of 256, 128, and 64 neurons each are utilized, followed by a 40% dropout layer. The output layer is made up of two neurons that employ a softmax activation function to categorize network data as normal or anomaly. Furthermore, the model is created with the Adam optimizer (5), a learning rate of 0.001, and a binary cross-entropy loss function (4), which enables fast weight updates and accurate classification. The combination of advanced activation functions, dropout regularization, and optimized architecture contributes to the model’s high performance, achieving strong accuracy while maintaining computational efficiency. The model summary is shown in Figure 7.

**Input Layer Transformation:**

(6)

Since the input shape is (41,1), flattening converts it into a vector of size 41.

**Hidden Layer (256 Neurons, Swish Activation):**

(7)

(8)

where:

* is the input data.
* is the input vector
* is the weight
* ​ is the bias vector
* is the sigmoid function.

**Hidden Layer (128 Neurons, GELU Activation):**

(9)

GELU (Gaussian Error Linear Unit) smooths activation with a probabilistic approach.

where:

* is the weight
* ​ is the bias vector

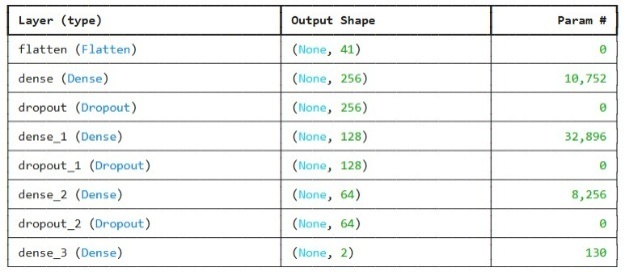


Figure 7. ANN Model Summary

**3.5. Hybrid Architecture**

The proposed hybrid model integrates both (CNN) and ANN architectures to enhance the accuracy and efficiency of anomaly detection in network traffic data. The model consists of two parallel branches: the CNN branch extracts local and spatial features using multiple Conv1D layers with GELU activation, followed by MaxPooling1D and dropout layers to reduce overfitting. The ANN branch, on the other hand, flattens the input features and passes them through several dense layers with GELU activations and dropout layers, focusing on learning complex patterns and relationships in the data. After independent feature extraction, both branches are concatenated, allowing the model to utilize the strengths of both architectures—CNN's ability to capture spatial dependencies and ANN's capacity for high-level feature abstraction. The combined output is processed through fully connected layers with dropout for regularization and ends with a sigmoid-activated output layer for binary classification. The model is compiled using the Adam optimizer and binary cross-entropy loss function, ensuring effective learning. By combining CNN and ANN branches, this hybrid model achieves superior performance in identifying anomalies while maintaining robustness against overfitting. The hybrid model summary is shown in Figure 8.

|  |  |
| --- | --- |
| Algorithm: Hybrid Model | |
| 1: | Hybrid CNN\_ANN (, , , ) |
| 2: | Input: Dataset (, , , ) |
| 3: | Output: Performance metrics |
| 4: | Define Hybrid CNN-ANN architecture: |
| 5: | CNN branch: set Conv1D layers, flatten, dense, and dropout layers |
| 6: | ANN branch: set Flatten, dense layers, and dropout layers |
| 7: | Concatenate CNN and ANN branches |
| 8: | Add fully connected layers with dense, and dropout layers |
| 9: | Compile the Hybrid CNN-ANN model |
| 10: | Set the batch size, optimizer, learning rate, epoch count (n), and early stopping criteria (esc) to their initial values. |
| 11: | for i in 1 to n do |
| 12: | while (esc) |
| 13: | Train Hybrid CNN-ANN model using , |
| 14: | end while |
| 15: | Evaluate the Hybrid CNN-ANN model using , and calculate evaluation metrics |
| 16: | Return the calculated metrics |

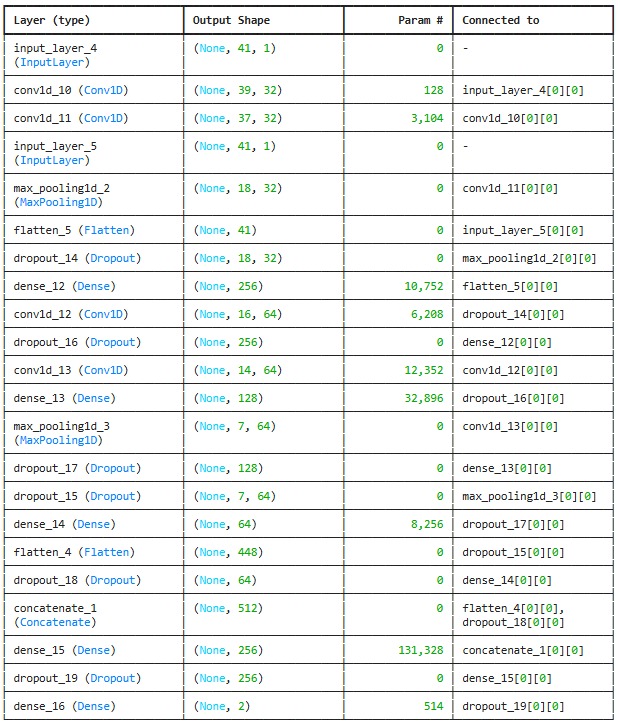


Figure 8. Hybrid Model Summary

1. **RESULTS AND DISCUSSION**

Table 3. Results of the proposed system for 40 epochs

|  |  |  |  |
| --- | --- | --- | --- |
| **Model** | **Accuracy** | **Loss** | **Val-accuracy** |
| CNN | 94.36% | 2.009% | 95.53% |
| ANN | 96.82% | 2.2037% | 94.63% |
| Hybrid | 97.65% | 2.6534% | 98.34% |

For all our models we tried different epoch sizes i.e. 20, 40 and 60, and among those, 40 epochs showed better fit. Table 3 shows the accuracy, loss, validation accuracy, and validation loss evaluation of the proposed models for 40 epochs from the dataset: Network Intrusion Detection [26]. We ran all our models on same device having Intel core i5 processor with 8 cores and NVIDIA GeForce GTX 6090. The CNN model demonstrated 94.36% accuracy with 95.53% validation accuracy although it resulted in higher validation loss. The ANN delivered marginal higher accuracy of 96.82% but showed a 94.63% validation accuracy which indicates an overfitting problem. The hybrid model delivered superior performance with 98.34% accuracy and 97.65% validation accuracy compared to standalone models due to its better learning ability. The higher model complexity in this system seems to lead to a 4.58% training loss quantity.

Table 4. Performance Metrics of the proposed system

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Model** | **Class type** | **Precision** | **Recall** | **F1-score** |
| CNN | Normal | 92% | 94% | 93% |
|  | Anomaly | 95% | 93% | 94% |
| ANN | Normal | 98% | 93% | 96% |
|  | Anomaly | 94% | 95% | 97% |
| Hybrid | Normal | 100% | 95% | 97% |
|  | Anomaly | 96% | 100% | 98% |

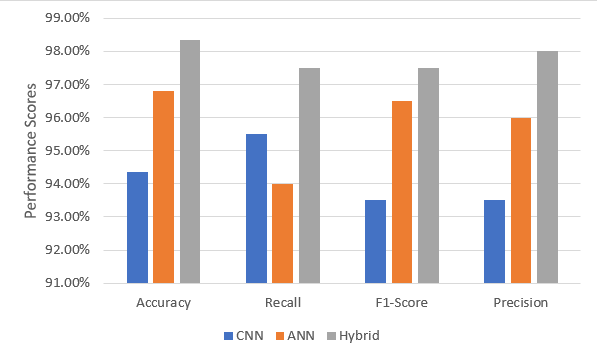


Figure 9. Comparison of Results

Table 4 and Figure 9 shows the performance analysis of CNN, ANN, and Hybrid models through performance metrics measurements for Normal and Anomaly detection. The CNN model shows precision of 92%, recall of 94% and F1-score of 93% for normal class and 95%, 93% and 94% respectively for anomaly class. ANN demonstrates precision of 98%, recall of 93% and F1-score of 94% for normal class and 94%, 95% and 97% respectively for anomaly class. The Hybrid model surpasses both previous models with its precision reaching 100%, 95% of recall and 97% of F1-score for normal instances while achieving 96% precision, 100% of recall and 98% of F1-score for anomalies.

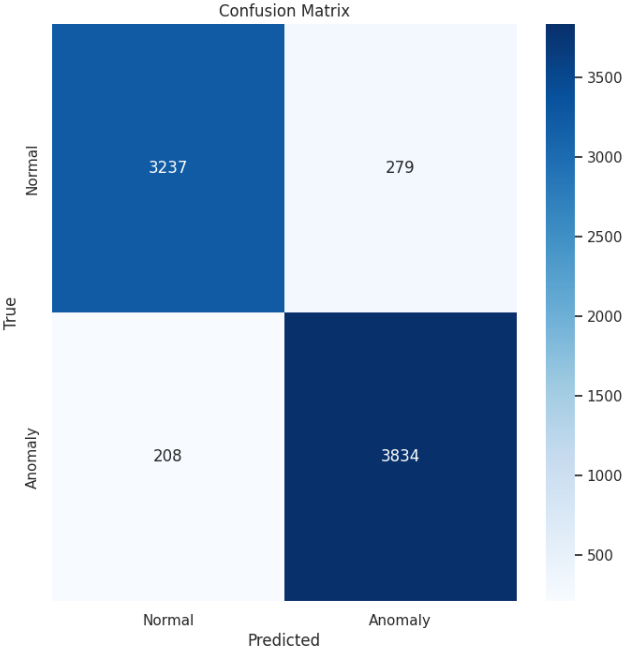


Figure 10. Confusion Martix

The confusion matrix evaluates how well a classification model diagnoses Normal and Anomaly cases. Where the model successfully detected Normal cases among a total of 6661 instances it obtained 3,237 True Positive results while achieving 3,834 True Negative results. Two hundred seventy-nine legitimate Normal cases were mistakenly identified as Anomalies while two hundred eight actual cases of Anomalies were mistakenly classified as Normal. The model presents good overall performance through its numerous correct classifications yet displays several misclassifications shown through its false positive and false negative values as shown in Figure 10.

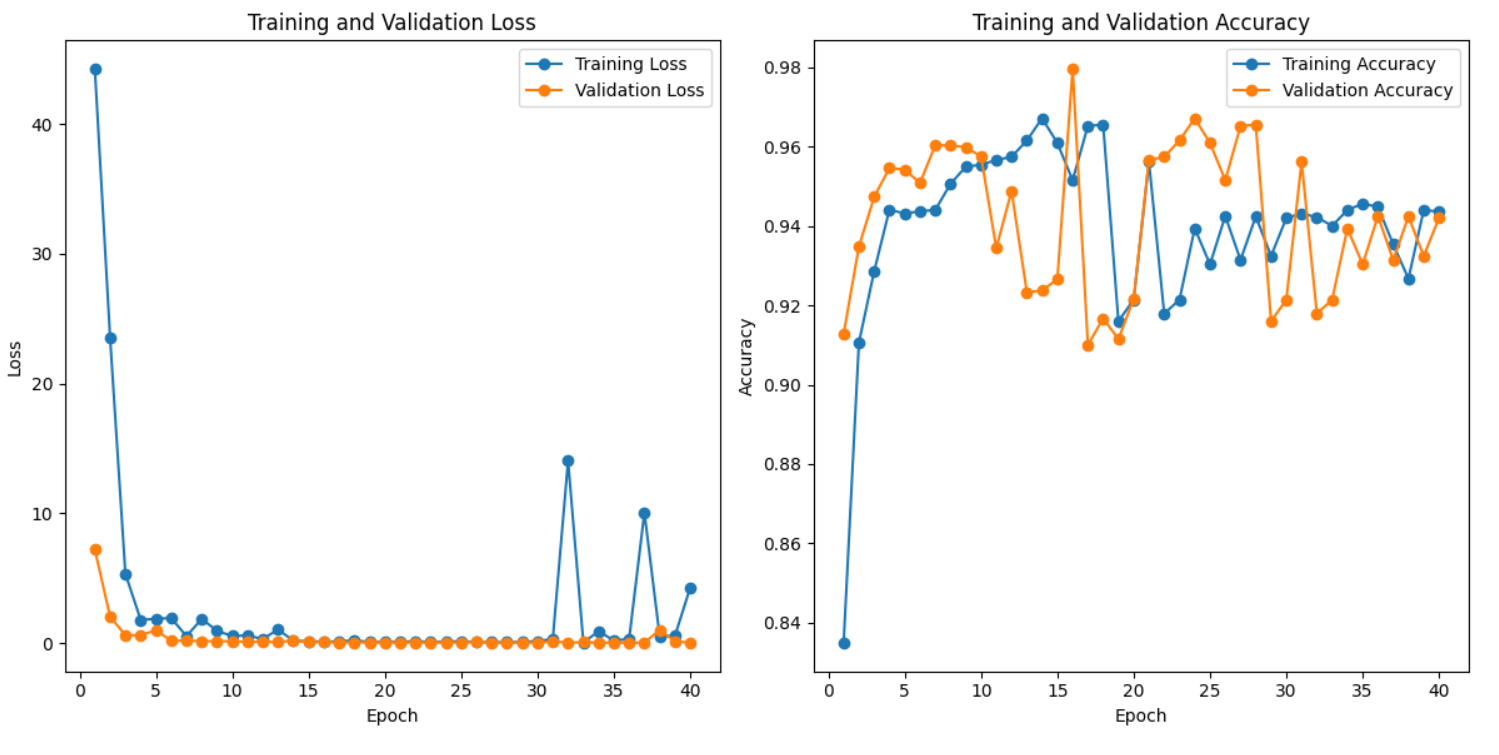


Figure 11. Performance of CNN

Figure 11 graph shows accuracy graph of CNN model with validation accuracy as red which is 95.53% and training accuracy as blue which is 94.36% across 40 epochs and loss graph of CNN model with validation loss as red which is 3.37% and training loss as blue which is 2.009% across 40 epochs.

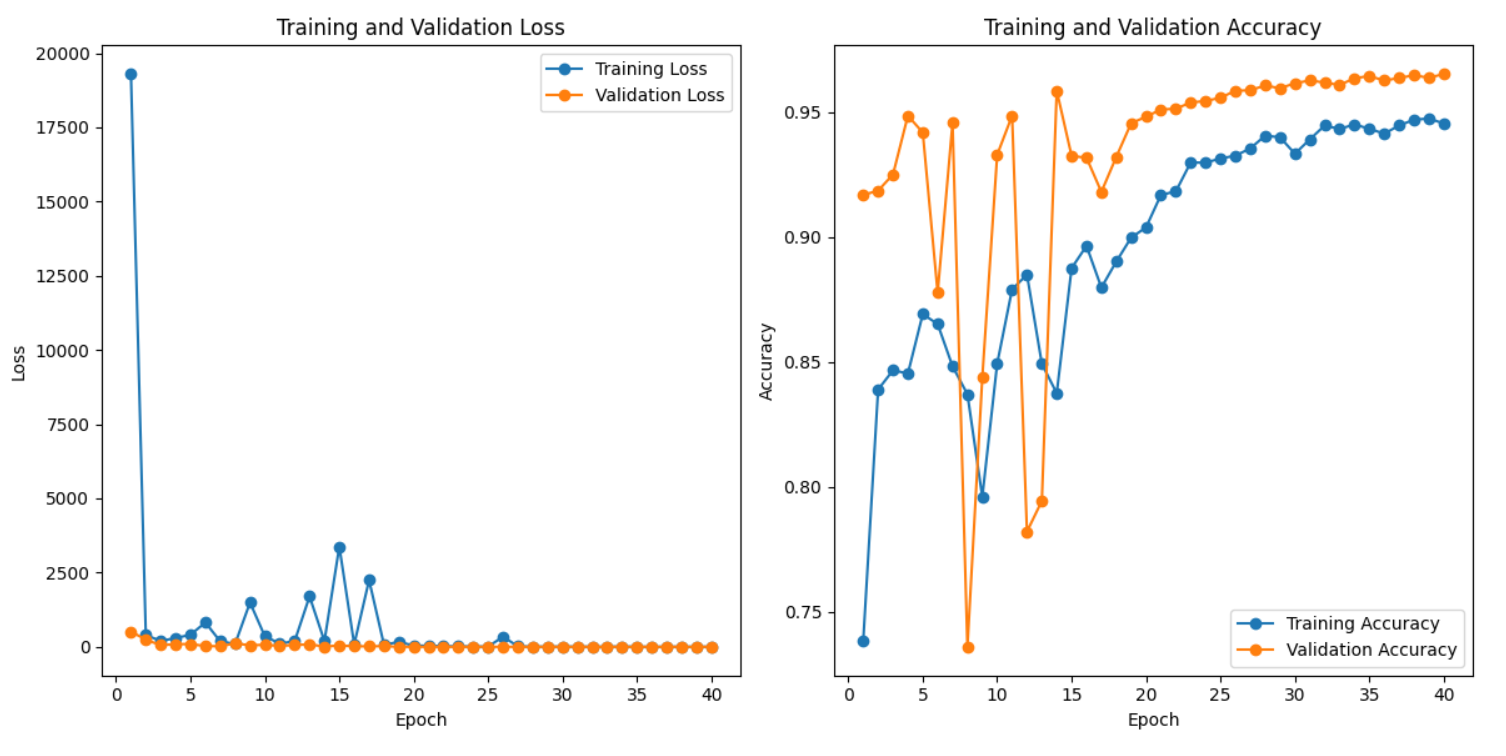


Figure 12. Performance of ANN

Figure 12 graph shows accuracy graph of ANN model with validation accuracy as red which is 94.63% and training accuracy as blue which is 96.82% across 40 epochs and graph shows loss graph of ANN model with validation loss as red which is 11.85% and training loss as blue which is 2.2037% across 40 epochs.

******

Figure 13. Performance of Hybrid

Figure 13 graph shows accuracy graph of hybrid model with validation accuracy as red which is 98.34% and training accuracy as blue which is 97.65% across 40 epochs. Fig.16 graph shows loss graph of hybrid model with validation loss as red which is 98.34% and training loss as blue which is 2.534% across 40 epochs. The authors Behdadnia, T., et al in [3] used CNN model which achieved 93% accuracy, and the authors in Zhu, Z., et al [11] also used CNN model which shows signs of overfitting as it is achieving 100% accuracy, and the authors in Bakhshi, T., et al [17] used a hybrid of CNN+GRU(Gated Recurrent Unit) which achieved 93.10% accuracy, comparative with all these researches our proposed hybrid model of CNN+ANN shows more efficiency, reduces overfitting and increased performance as shown in Fig.17. The training accuracy shows rapid growth starting from zero up to approximately 98.34% before experiencing both small improvements and small declines. The validation accuracy levels stay stable at 97.65% after exceeding 95% with negligible ups and downs. The model demonstrates strong performance in test data although training accuracy fails to match it which might result from regularization techniques or the model achieving widespread generalization as opposed to overfitting.

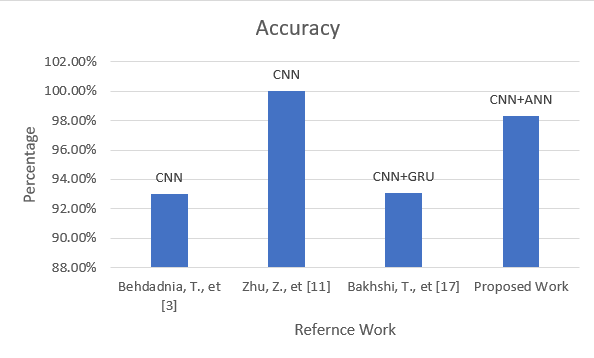


Figure 14. Comparison of Proposed work with existing studies

1. **CONCLUSION**

The proposed hybrid model of ANN and CNN system achieve higher accuracy and shows much more promising results for anomaly detection in encryption traffic. The data preprocessing module assures that no null values existed and we performed label encoding to change categorical data to numerical data. To remove class imbalance and balance the dataset we used SMOTE technique. The ANN and CNN architectures were refined with added hidden layers to increase the efficiency, and we used ReLU and GeLU activation functions to update weights for better performance. We also used Adam and RMSprop optimizers for the model to learn more effectively. Sigmoid activation function is used to venture more accurate results for binary classification. Both of the models did give very good results with 96.82% and 94.36% accuracies respectively, but for more compelling and promising outputs we combined them both into a hybrid model, which gave us excellent results with 98.34% accuracy. Comparatively the hybrid model is more promising, efficient and effective than refined ANN and CNN architectures. The proposed hybrid model also achieved higher precision of 98%, recall of 96.5% and F1-score of 97.5%.

**ACKNOWLEDGMENTS *(if applicable)***

This section should acknowledge individuals who provided personal assistance to the work but do not meet the criteria for authorship, detailing their contributions. It is imperative to obtain consent from all individuals listed in the acknowledgments.

**FUNDING INFORMATION *(mandatory)***

This section should describe sources of funding agency that have supported the work. Authors should state how the research described in their article was funded, including grant numbers if applicable. Include the following (or similar) statement if there is no funding involved: Authors state no funding involved.

**AUTHOR CONTRIBUTIONS STATEMENT**

This journal uses the Contributor Roles Taxonomy (CRediT) to recognize individual author contributions, reduce authorship disputes, and facilitate collaboration. **The recommended number of authors is at least two, with one of them designated as the corresponding author.** The corresponding author will be responsible for all correspondence related to the paper and must ensure that the other authors are included in the communication regarding submission, revision, and publication processes. We encourage authors to include a statement in the paper that shares and accurately describes each author's contribution. **To be eligible for authorship, each individual must have contributed to at least one of the following: conceptualization, methodology, formal analysis, or investigation, as well as at least one aspect of writing (either original draft preparation or writing reviews and editing).**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Name of Author** | **C** | **M** | **So** | **Va** | **Fo** | **I** | **R** | **D** | **O** | **E** | **Vi** | **Su** | **P** | **Fu** |
| Dogiparthi Aasrith | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |  | ✓ | ✓ | ✓ |  |  | ✓ |  |
| Kalla Bharath Vardhan |  | ✓ |  |  |  | ✓ |  | ✓ | ✓ | ✓ | ✓ | ✓ |  |  |
| Varanasi Sai Charan | ✓ |  | ✓ | ✓ |  |  | ✓ |  |  | ✓ | ✓ |  | ✓ | ✓ |
| Dr. Bhavadharini R M |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

|  |  |  |
| --- | --- | --- |
| C : **C**onceptualization  M : **M**ethodology  So : **So**ftware  Va : **Va**lidation  Fo : **Fo**rmal analysis | I : **I**nvestigation  R : **R**esources  D : **D**ata Curation  O : Writing - **O**riginal Draft  E : Writing - Review & **E**diting | Vi : **Vi**sualization  Su : **Su**pervision  P : **P**roject administration  Fu : **Fu**nding acquisition |

**CONFLICT OF INTEREST STATEMENT *(mandatory)* (10 PT)**

To ensure fair and objective decision-making, authors must declare any associations that pose a conflict of interest (financial, personal, or professional) in connection with manuscripts. Non-financial competing interests include a declaration of political, personal, religious, ideological, academic, and intellectual competing interests. The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper. If there are no conflicts of interest, please include the following author's statement: Authors state no conflict of interest.

**INFORMED CONSENT *(if applicable)* (10 PT)**

The protection of privacy is a legal right that must not be breached without individual informed consent. In cases where the identification of personal information is necessary for scientific reasons, authors should obtain full documentation of informed consent, including written permission from the patient prior to inclusion in the study. Incorporate the following (or a similar) statement: We have obtained informed consent from all individuals included in this study.

**ETHICAL APPROVAL *(if applicable)* (10 PT)**

When papers talk about using people or animals, authors should make it clear that the research followed all national rules and institutional policies, and it was approved by the authors' institutional review board or a similar committee. The Helsinki Declaration's tenets must guide all investigations involving human subjects. Authors must also identify the committee or review board approving the experiments and provide a statement indicating approval of the research. Incorporate the following (or a similar) statement: The research related to human use has been complied with all the relevant national regulations and institutional policies in accordance with the tenets of the Helsinki Declaration and has been approved by the authors' institutional review board or equivalent committee; or: The research related to animal use has been complied with all the relevant national regulations and institutional policies for the care and use of animals.

**DATA AVAILABILITY *(mandatory)* (10 PT)**

The data that support the findings of this study are openly available in Kaggle at <https://www.kaggle.com/datasets/sampadab17/network-intrusion-detection> [26]

**REFERENCES**

[1] Wu, Z., Li, H., Qian, Y., Hua, Y., & Gan, H. (2024). Poison-resilient anomaly detection: Mitigating poisoning attacks in semi-supervised encrypted traffic anomaly detection. IEEE Transactions on Network Science and Engineering.

[2] Xing, J., & Wu, C. (2020, July). Detecting anomalies in encrypted traffic via deep dictionary learning. In IEEE INFOCOM 2020-IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS) (pp. 734-739). IEEE.

[3] Behdadnia, T., Deconinck, G., Ozkan, C., & Singelee, D. (2023, October). Encrypted Traffic Classification for Early-Stage Anomaly Detection in Power Grid Communication Network. In 2023 IEEE PES Innovative Smart Grid Technologies Europe (ISGT EUROPE) (pp. 1-6). IEEE.

[4] Yu, T., Zou, F., Li, L., & Yi, P. (2019, October). An encrypted malicious traffic detection system based on neural network. In 2019 international conference on cyber-enabled distributed computing and knowledge discovery (CyberC) (pp. 62-70). IEEE.

[5] Lv, G., Yang, R., Wang, Y., & Tang, Z. (2020, August). Network encrypted traffic classification based on secondary voting enhanced random forest. In 2020 IEEE 3rd International Conference on Computer and Communication Engineering Technology (CCET) (pp. 60-66). IEEE.

[6] Kopp, M., Grill, M., & Kohout, J. (2018, December). Community-based anomaly detection. In 2018 IEEE International Workshop on Information Forensics and Security (WIFS) (pp. 1-6). IEEE.

[7] Han, S., Wu, Q., Zhang, H., & Qin, B. (2022, July). Light-weight unsupervised anomaly detection for encrypted malware traffic. In 2022 7th IEEE International Conference on Data Science in Cyberspace (DSC) (pp. 206-213). IEEE.

[8] Shen, M., Ye, K., Liu, X., Zhu, L., Kang, J., Yu, S., ... & Xu, K. (2022). Machine learning-powered encrypted network traffic analysis: A comprehensive survey. IEEE Communications Surveys & Tutorials, 25(1), 791-824.

[9] Meghdouri, F., Vázquez, F. I., & Zseby, T. (2020, October). Cross-layer profiling of encrypted network data for anomaly detection. In 2020 IEEE 7th International Conference on Data Science and Advanced Analytics (DSAA) (pp. 469-478). IEEE.

[10] Wang, Z., Wang, P., & Sun, Z. (2022, December). SDN traffic anomaly detection method based on convolutional autoencoder and federated learning. In GLOBECOM 2022-2022 IEEE Global Communications Conference (pp. 4154-4160). IEEE.

[11] Zhu, Z., Zhou, H., Yang, Q., Wang, C., & Li, Z. (2022, December). Anomaly Detection in Encrypted Identity Resolution Traffic based on Machine Learning. In 2022 IEEE 22nd International Conference on Software Quality, Reliability and Security (QRS) (pp. 264-275). IEEE.

[12] Ucci, D., Sobrero, F., Bisio, F., & Zorzino, M. (2021, December). Near-real-time anomaly detection in encrypted traffic using machine learning techniques. In 2021 IEEE Symposium Series on Computational Intelligence (SSCI) (pp. 01-08). IEEE.

[13] Kim, M. G., & Kim, H. (2024). Anomaly Detection in Imbalanced Encrypted Traffic with Few Packet Metadata-Based Feature Extraction. CMES-Computer Modeling in Engineering & Sciences, 141(1).

[14] Ashraf, J., Bakhshi, A. D., Moustafa, N., Khurshid, H., Javed, A., & Beheshti, A. (2020). Novel deep learning-enabled LSTM autoencoder architecture for discovering anomalous events from intelligent transportation systems. IEEE Transactions on Intelligent Transportation Systems, 22(7), 4507-4518.

[15] Long, G., & Zhang, Z. (2023). Deep encrypted traffic detection: An anomaly detection framework for encryption traffic based on parallel automatic feature extraction. Computational Intelligence and Neuroscience, 2023(1), 3316642.

[16] Pratiwi, M., & Choi, Y. H. (2024, September). Context-Aware Anomaly-based Detection for Ransomware using Multivariate Feature. In 2024 IEEE Conference on Communications and Network Security (CNS) (pp. 1-9). IEEE.

[17] Tian, Z., Cui, L., Liang, J., & Yu, S. (2022). A comprehensive survey on poisoning attacks and countermeasures in machine learning. ACM Computing Surveys, 55(8), 1-35.

[18] Anderson, B., & McGrew, D. (2016, October). Identifying encrypted malware traffic with contextual flow data. In Proceedings of the 2016 ACM workshop on artificial intelligence and security (pp. 35-46).

[19] Shekhawat, A. S., Di Troia, F., & Stamp, M. (2019). Feature analysis of encrypted malicious traffic. Expert Systems with Applications, 125, 130-141.

[20] Velan, P., Čermák, M., Čeleda, P., & Drašar, M. (2015). A survey of methods for encrypted traffic classification and analysis. International Journal of Network Management, 25(5), 355-374.

[21] Rezaei, S., & Liu, X. (2019). Deep learning for encrypted traffic classification: An overview. IEEE communications magazine, 57(5), 76-81.

[22] Wang, J., He, T., He, G., Zhu, H., Xu, B., & Zhang, L. (2023, December). Cooperative Detection of Camouflaged Malicious TLS Traffic. In 2023 Eleventh International Conference on Advanced Cloud and Big Data (CBD) (pp. 116-121). IEEE.

[23] Wingarz, T., See, A., Gondesen, F., & Fischer, M. (2024, September). Privacy-preserving Network Anomaly Detection on Homomorphically Encrypted Data. In 2024 IEEE Conference on Communications and Network Security (CNS) (pp. 1-9). IEEE.

[24] Bakhshi, T., & Ghita, B. (2021). Anomaly detection in encrypted internet traffic using hybrid deep learning. Security and Communication Networks, 2021(1), 5363750.

[25] Lee, I., Roh, H., & Lee, W. (2020, July). Encrypted malware traffic detection using incremental learning. In IEEE INFOCOM 2020-IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS) (pp. 1348-1349). IEEE.

[26] https://www.kaggle.com/datasets/sampadab17/network-intrusion-detection

**BIOGRAPHIES OF AUTHORS**

|  |  |
| --- | --- |
|  | **Dogiparthi Aasrith** currently in his final year B. Tech Computer Science and Engineering with Specialization in Artificial Intelligence and Machine Learning at Vellore Institute of Technology, Chennai. He is a dedicated student with a strong foundation in machine learning and deep learning. His studies focus on CSE, allowing him to delve into various facets of technology and its applications. He has adeptly executed research and projects in machine learning and deep learning, demonstrating a blend of creativity and technical prowess. These endeavors not only attest to his hands-on experience but also underscore his passion for devising inventive solutions within the field. With his proactive and forward-thinking approach distinguishes him, offering a valuable amalgamation of academic excellence and an unwavering dedication to remaining at the vanguard of technological advancements. He can be contacted at email: dogiparthi.aasrith2021@vitstudent.ac.in |
|  |  |
| **D:\Certificates\Passport Photo-min.jpeg** | **Kalla Bharath Vardhan** is in his final year of B. Tech Computer Science and Engineering at Vellore Institute of Technology, Chennai with a curiosity and hunger for knowledge. Being focused and with a solid foundation in Machine learning and deep learning, he seamlessly integrated theoretical knowledge with practical applications. Even though he just started his journey, it is marked by an impressive ability to navigate complex algorithms and a keen eye for innovation. Bharath's proactive approach and collaborative spirit have enriched his academic experience and set him apart as a multifaceted individual with a passion for pushing the boundaries of technological advancement. With a combination of academic excellence and hands-on expertise, he makes a noteworthy contribution through this work. He can be contacted at email: kallabharathvardhan.2021@vitstudent.ac.in. |
|  |  |
|  | **Varanasi Sai Charan** is currently a final year student pursuing a degree in Computer Science and Engineering at Vellore Institute of Technology in Chennai. He possesses a strong dedication to his studies and exhibits advanced skills in machine learning and deep learning. His academic pursuits encompass a wide range of technological subjects. Sai Charan has notably excelled in both research and project work related to machine learning, showcasing a blend of creativity and technical expertise. These endeavors highlight his practical experience and a genuine passion for developing innovative solutions. His proactive and forward- thinking approach distinguishes him, as he seamlessly integrates academic excellence with a commitment to staying abreast of the latest technological advancements. As he progresses in his studies, Sai Charan maintains a steadfast focus on continuous learning and research. He can be contacted at email: varanasisai.charan2021@vitstudent.ac.in. |
|  |  |
|  | **Dr. Bhavadharini R M** is currently working as Associate Professor in the School of Computer Science Engineering -Vellore Institute of Technology, Chennai Campus. She has specialized in the domain of Mobile Ad Hoc Networks and obtained her Doctoral degree from Anna University, Chennai in the year 2016. She has received her B.E in CSE in the year 2002 and M.E in CSE in the year 2009 with First class with Distinction. She has around 17 years of teaching experience. Her research interest spans across wireless networks, Internet of Things, Machine Learning and Deep Learning. She has published more than 30 research papers in reputed journals with an h-index value of 5. She has acted as reviewer for book chapter published by Wiley. She has chaired many technical conference sessions and seminars. She has one patent granted in the domain of Blockchain and IoT and one patent published in vehicle tracking system. She can be contacted at email: bhavadharini.rm@vit.ac.in. |