



# Lightweight Convolutional Neural Network for Anomaly Detection in Encrypted Network Traffic Using Flow Features

Jonathan Amos Oguzebeh<sup>1</sup>, Siman Emmanuel<sup>2\*</sup>, Ishaya Joseph<sup>3</sup>

<sup>1,2,3</sup>Department of Computer Science Kwararafa University, Wukari, Nigeria

<sup>2</sup>Department of Cybersecurity Federal University, Wukari, Nigeria.

**\*Corresponding Author**

Siman Emmanuel

**Article History**

Received: 10.06.2026

Accepted: 11.07.2026

Published: 26.08.2026

**Abstract:** The increasing use of encryption protocols such as Transport Layer Security (TLS) and QUIC has reduced the effectiveness of conventional intrusion detection systems that rely on packet payload inspection. This study proposes XCNN-AD, a lightweight Convolutional Neural Network framework for anomaly detection in encrypted network traffic using flow-level statistical features without decrypting packet contents. The framework was evaluated using the CICIDS2017 and UNSW-NB15 benchmark datasets through a preprocessing pipeline involving data cleaning, feature selection, normalization, class balancing, and transformation of flow features into a structured spatial grid. The proposed model integrates CNN-based feature extraction with Focal Loss, Mixup augmentation, dropout, batch normalization, probability calibration, and Integrated Gradients for explainable detection. Results on the CICIDS2017 test dataset showed that XCNN-AD achieved 97.93% precision, recall, and F1-score, with an AUC-ROC of 0.9975 and a false positive rate of 0.42% across 2,522,362 test flows. The model outperformed MLP, LSTM, and Autoencoder baselines, while ablation analysis confirmed the importance of the 2D spatial feature representation, whose removal reduced the F1-score to 94.11%. The framework also demonstrated high computational efficiency, with approximately 186,000 parameters, a 0.7 MB model size, and a throughput of 91,006 flows per second on commodity CPU hardware. Temperature scaling improved model calibration by reducing Expected Calibration Error from 0.065 to 0.012. Overall, XCNN-AD provides an accurate, lightweight, privacy-preserving, and explainable solution for encrypted traffic anomaly detection, although improving cross-dataset generalisation remains an important direction for future research.

**Keywords:** Encrypted Network Traffic; Anomaly Detection; Convolutional Neural Network; Intrusion Detection System; Deep Learning; Cybersecurity; Flow-Level Features.

This work is licensed under a [Creative Commons Attribution-NonCommercial 4.0 International License](https://creativecommons.org/licenses/by-nc/4.0/) (CC BY-NC 4.0)

**Cite this article:**

Oguzebeh, J. A., Emmanuel, S., Joseph, I., (2026). Lightweight Convolutional Neural Network for Anomaly Detection in Encrypted Network Traffic Using Flow Features. *ISAR Journal of Science and Technology*, 4(8), 50-59.

## I. Introduction

As internet communications increasingly rely on encryption protocols such as TLS and SSH to protect data privacy, traditional security methods like Deep Packet Inspection (DPI) and signature-based detection have become less effective because they cannot inspect encrypted content (Al-Turaiki & Altwaijry, 2021). The document explains that enterprise networks generate large volumes of encrypted traffic, making it difficult to identify cyber threats such as command-and-control (C2) communications and data exfiltration. While encryption enhances user privacy and data security, it also provides attackers with a means to hide malicious activities within legitimate encrypted connections (Villegas-Ch et al., 2025). The document also identifies a research gap. Although deep learning models such as Convolutional Neural Networks (CNNs) and Long Short-Term Memory (LSTM) networks have demonstrated high detection accuracy, they often operate as "black boxes," providing little explanation for their decisions. This lack of explainability reduces trust among cybersecurity analysts during real-time threat detection. The study seeks to develop an efficient, explainable, and real-time CNN-based anomaly detection framework capable of identifying malicious encrypted traffic without compromising encryption or user privacy, thereby improving cybersecurity in enterprise networks. As more internet communications use encryption protocols such as TLS, traditional intrusion detection techniques—such as signature-based detection and Deep Packet Inspection (DPI)—can no longer inspect packet contents without compromising encryption (Akbari et al., 2022). Although flow-level statistical features (such as packet size, duration, and traffic patterns) remain available for analysis, many existing flow-based intrusion detection methods have significant limitations. They often generate high false alarm rates, fail to adapt to changing network traffic and evolving cyberattacks, or require excessive computational resources, making them unsuitable for real-time deployment (Al-Turaiki & Altwaijry, 2021; Villegas-Ch et al., 2025). Another major issue is that most modern deep learning intrusion detection systems only classify traffic as "Normal" or "Anomaly" without explaining why a particular decision was made. This lack of transparency reduces analysts' confidence in the system, slows incident investigation, and contributes to alert fatigue in Security Operations Centres (SOCs), where analysts must process large numbers of security alerts (Amarasinghe et al., 2019; Birihanu & Lendak, 2025). Hence, this study addresses the need for a lightweight deep learning-based anomaly detector that uses only flow-level statistical features for encrypted traffic and provides per-alert feature-attribution evidence to support analyst decision-making. The Research Contributions as follows:

- i. Developed a lightweight CNN-based framework (XCNN-AD) for privacy-preserving anomaly detection in encrypted network traffic using flow-level features.
- ii. Introduced a spatial feature-grid representation that improves CNN-based traffic pattern learning and detection performance.
- iii. Integrated explainable AI using Integrated Gradients to identify important network features influencing model decisions.
- iv. Demonstrated high detection accuracy with reduced computational complexity, enabling efficient deployment in resource-constrained environments.

The remainder of the sections are as follows. Section 2 presents the methodology adopted for the development and evaluation of the proposed XCNN-AD framework, including the research design, dataset selection, preprocessing procedures, CNN architecture, and evaluation strategy. Section 3 describes the model development process, including the proposed architecture configuration, optimization techniques, and cross-dataset evaluation approach. Section 4 discusses the computational efficiency analysis and explainability assessment of the framework. Section 5 presents the experimental results, including binary classification performance, ROC analysis, and ablation study findings. Section 6 provides a detailed discussion of the results, highlighting the implications, strengths, and limitations of the proposed approach. Finally, Section 7 concludes the study by summarizing the major findings and outlining future research directions.

## II. Methodology

The methodology describes the systematic process used to design, develop, and evaluate a lightweight Convolutional Neural Network (CNN) framework called XCNN-AD for detecting anomalies in encrypted network traffic using flow-level statistical features. The study follows a structured machine learning approach to ensure the framework is accurate, efficient, reproducible, and suitable for real-world deployment. The study adopts a Design Science Research (DSR) paradigm combined with a quantitative experimental approach. DSR is used to design and evaluate the XCNN-AD model as a practical cybersecurity solution, while quantitative experiments measure its detection accuracy, computational efficiency, and ability to generalise across different datasets. Public benchmark datasets are used to ensure reproducibility and comparison with previous studies.

The methodology follows the CRISP-ML/Q (Cross-Industry Standard Process for Machine Learning with Quality Assurance) framework. The research progresses through five main phases: Data Acquisition and Understanding, Feature Engineering, Model Development, Evaluation, Explainability and Deployments.

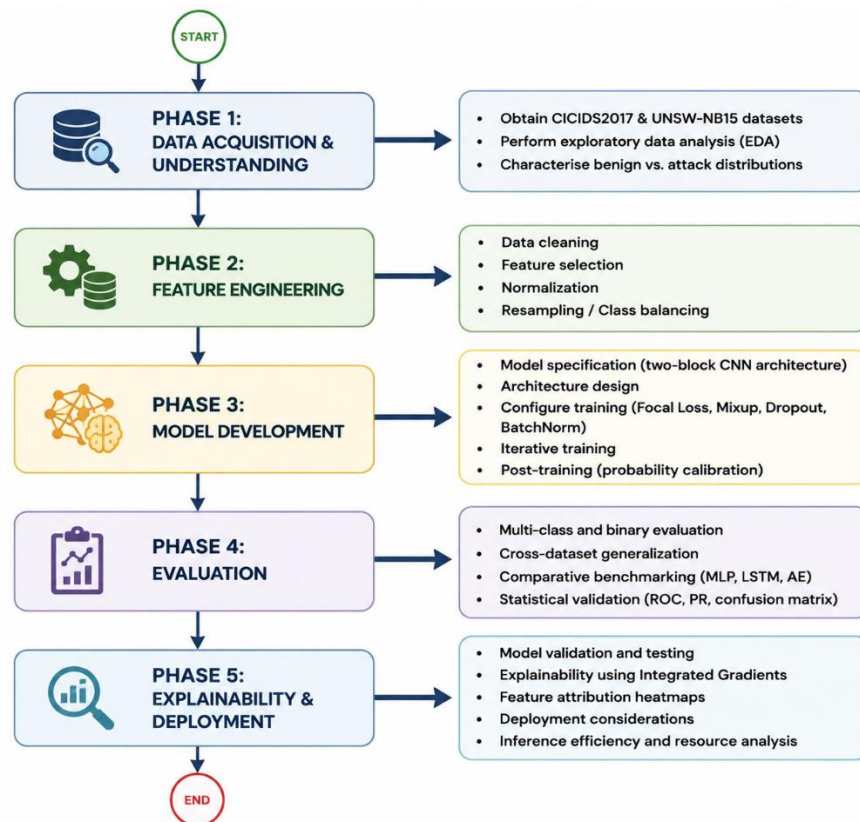


Figure 1: Activity Diagram - Research Design Flowchart Source: Researcher's own illustration (2026)

### Dataset Selection

Two widely recognised cybersecurity benchmark datasets, CICIDS2017 and UNSW-NB15, are employed in this study to evaluate the proposed intrusion detection framework. CICIDS2017 is used for model development, training, and in-domain testing, while UNSW-NB15 is used for cross-dataset evaluation to assess the model's generalisation capability on unseen network traffic patterns. Both datasets contain extensive network flow records representing normal activities and multiple categories of cyber-attacks, making them suitable for benchmarking intrusion detection systems and evaluating the robustness of machine learning-based security models.

### Feature Engineering and Data Preprocessing

The raw network traffic data are transformed into structured inputs suitable for CNN-based analysis through a systematic preprocessing pipeline. The process involves data cleaning to remove inconsistencies, feature selection to identify relevant traffic attributes, missing value imputation, and outlier treatment to improve data quality. The processed features are then normalised and converted into structured feature grids for CNN representation, followed by class

balancing to address potential imbalance between normal and attack traffic samples. This preprocessing framework improves data consistency, enhances model learning capability, and supports reproducible evaluation of the proposed intrusion detection model.

### CNN Architecture

The proposed CNN architecture is designed as a lightweight model for efficient network anomaly detection. It consists of two convolutional layers for feature extraction, followed by batch normalisation and ReLU activation functions to improve learning stability and nonlinear feature representation. Two max-pooling layers are applied to reduce feature dimensionality, after which the extracted features are flattened and passed through two fully connected dense layers. The final Softmax output layer performs classification of network traffic categories. The model contains approximately 186,000 parameters, making it significantly smaller and computationally more efficient than large-scale CNN architectures such as ResNet and VGG, thereby supporting deployment in resource-constrained environments.

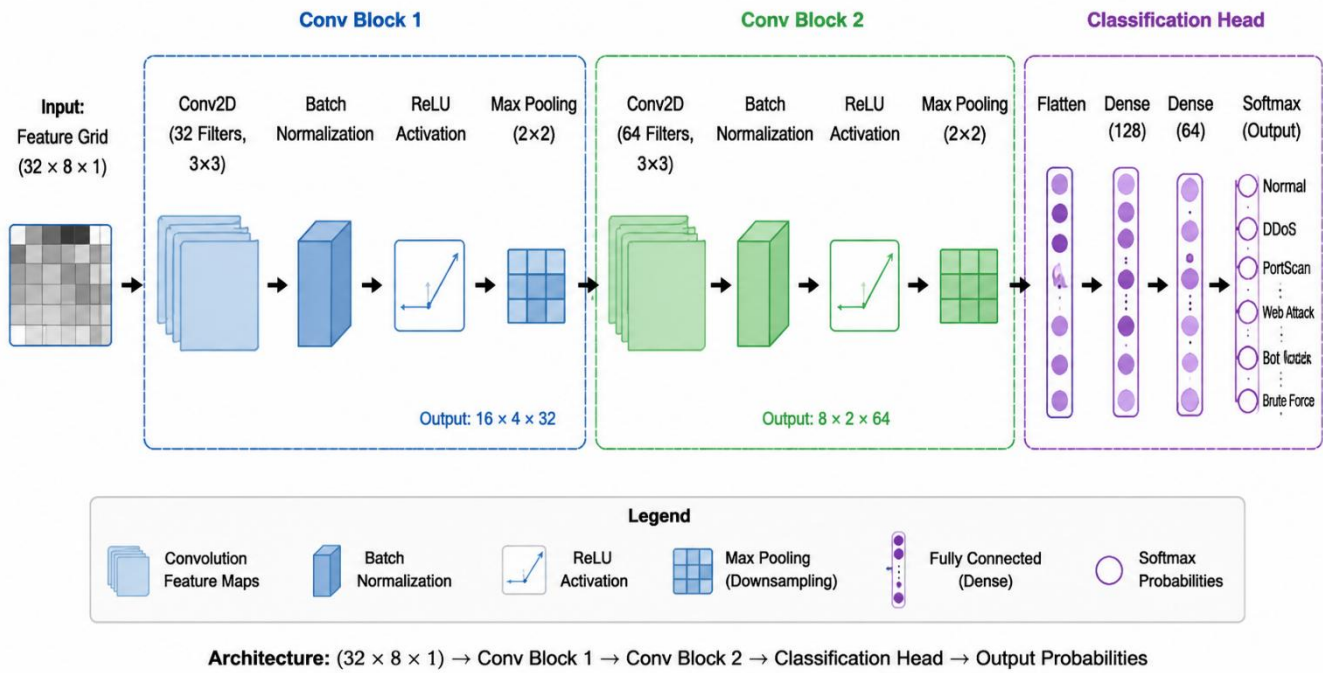


Figure 1: Proposed Lightweight CNN Architecture for Encrypted Network Traffic Anomaly Detection

Figure 1 illustrates the architecture of the proposed lightweight Convolutional Neural Network (CNN) framework for detecting anomalies in encrypted network traffic using flow-level features. The model begins with an input feature grid representation of network traffic data, which is processed through two convolutional blocks consisting of convolutional layers, batch normalisation, ReLU activation, and max-pooling operations for feature extraction and dimensionality reduction. The extracted

features are then flattened and passed through fully connected dense layers before reaching the Softmax output layer, which performs traffic classification. With approximately 186,000 trainable parameters and a compact model size, the architecture provides an efficient alternative to larger CNN models such as ResNet and VGG, making it suitable for real-time intrusion detection applications with limited computational resources.

## Model Evaluation

Table 1: Evaluation Metrics Used for Assessing the XCNN-AD Framework

| Evaluation Metric                         | Formula / Measurement Basis                                                                                              | Purpose in Model Evaluation                                                                                                       |
|-------------------------------------------|--------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------|
| <b>Accuracy</b>                           | Ratio of correctly classified samples (both normal and attack traffic) to the total number of samples                    | Measures the overall correctness of the model in classifying network traffic instances.                                           |
| <b>Precision</b>                          | Proportion of correctly detected attacks among all samples predicted as attacks                                          | Evaluates the reliability of attack predictions and indicates the level of false alarms generated by the model.                   |
| <b>Recall (True Positive Rate)</b>        | Proportion of actual attacks correctly identified by the model                                                           | Measures the ability of the model to detect malicious activities and minimise missed attacks.                                     |
| <b>F1-score</b>                           | Harmonic mean of precision and recall                                                                                    | Provides a balanced assessment between attack detection capability and false alarm reduction, especially for imbalanced datasets. |
| <b>False Positive Rate (FPR)</b>          | Ratio of normal traffic incorrectly classified as malicious traffic                                                      | Determines the frequency of false alerts produced by the intrusion detection system.                                              |
| <b>Area Under the ROC Curve (AUC-ROC)</b> | Measures the model's ability to distinguish between normal and attack classes across different classification thresholds | Evaluates overall discrimination capability and robustness of the classifier.                                                     |

The XCNN-AD framework in Table 1, is evaluated using multiple performance metrics to provide a comprehensive assessment of its detection capability. Accuracy provides an overall measure of classification performance, while precision, recall, and F1-score evaluate the model's effectiveness in identifying cyber-attacks while controlling false predictions. The False Positive Rate measures the occurrence of unnecessary security alerts, which is critical for practical intrusion detection systems. The AUC-ROC further assesses the model's ability to separate normal and malicious traffic across different decision thresholds, ensuring a reliable evaluation of the proposed framework.

### III. Results and Discussion

The XCNN-AD framework was implemented using Python-based machine learning libraries, including PyTorch, Scikit-learn, Pandas, NumPy, Captum, and MLflow, to support reproducible model development, deterministic training, and systematic experiment tracking. The experimental evaluation was performed on a standard CPU-based computing environment, demonstrating that the proposed lightweight CNN architecture can achieve efficient operation without dependence on high-performance GPU hardware. The implementation followed a structured software design with separate modules for data preprocessing, model training, performance evaluation, explainability analysis, and experiment logging. The proposed framework was evaluated using the CICIDS2017 and UNSW-NB15 benchmark datasets to represent realistic network traffic conditions. Before model training, the datasets underwent comprehensive data preparation and feature engineering procedures to improve data quality and enhance learning performance. The preprocessing pipeline involved removing duplicate records, replacing invalid and infinite values, eliminating constant features, performing exploratory data analysis to understand feature distributions and class imbalance, and selecting relevant network flow features for CNN-based representation. These procedures ensured that the input data were consistent, reliable, and suitable for effective training of the proposed anomaly detection model.

### IV. Model Development

The proposed XCNN-AD architecture was developed as a lightweight deep learning framework for encrypted network traffic anomaly detection. The model consists of two convolutional blocks incorporating convolutional operations, batch normalisation, ReLU activation functions, and max-pooling layers for effective feature extraction and dimensionality reduction. The extracted representations are processed through fully connected layers with dropout before classification through a Softmax output layer. To enhance model robustness and learning performance, the framework integrates Focal Loss to address class imbalance,

Mixup data augmentation to improve generalisation, AdamW optimisation with cosine annealing learning-rate scheduling for efficient parameter updates, and temperature scaling to improve prediction confidence calibration.

#### Cross-Dataset Evaluation

The cross-dataset generalisation capability of the XCNN-AD framework was evaluated by training the model on CICIDS2017 and directly testing it on the independent UNSW-NB15 dataset without additional retraining. The model achieved an accuracy of 87.35% and a macro F1-score of 46.62%. However, the binary precision, recall, and F1-score showed a substantial reduction, indicating that although the model performs strongly under in-domain evaluation, transferring learned traffic patterns across different datasets remains challenging. This highlights the need for further investigation into domain adaptation strategies and more robust feature representations for cross-network generalisation.

Table 2: Cross-Dataset Evaluation Metrics (CICIDS2017 → UNSW-NB15, n = 381,008)

| Metric                    | Value  |
|---------------------------|--------|
| Accuracy                  | 0.8735 |
| Precision                 | 0.0000 |
| Recall                    | 0.0000 |
| F1-score (Binary)         | 0.0000 |
| False Positive Rate (FPR) | 0.0000 |
| True Positive Rate (TPR)  | 0.0000 |
| AUC-ROC                   | 0.5000 |
| F1-score (Macro)          | 0.4662 |

The cross-dataset evaluation results in Table 2, demonstrate a clear performance gap between in-domain and external dataset testing. Although the model maintained reasonable overall accuracy, the zero binary precision, recall, and F1-score together with an AUC-ROC of 0.5000 indicate limited capability in distinguishing attack samples within the unseen UNSW-NB15 dataset. This suggests that dataset-specific characteristics, traffic distribution differences, and attack representation variations significantly influence generalization performance.

#### Computational Efficiency Analysis

The computational efficiency evaluation demonstrates that the XCNN-AD framework satisfies the targeted requirements for lightweight and resource-efficient deployment. Benchmarking results confirmed that the model achieved a p95 inference latency of  $\leq 20$  ms and a throughput of  $\geq 10,000$  network flows per second using commodity CPU hardware, indicating its suitability for real-time anomaly detection applications without reliance on

specialised computing resources. The model has a compact size of approximately 0.7 MB based on the serialised PyTorch state dictionary, which remains well below the 5 MB deployment threshold and supports implementation in memory-constrained environments. Furthermore, the architecture contains approximately 186,000 trainable parameters, substantially fewer than large-scale CNN architectures such as VGG-16 with 138 million parameters and ResNet-50 with 25 million parameters, highlighting the efficiency of XCNN-AD while maintaining effective detection capability.

Figure 2 presents a comparative analysis of the evaluated models based on their detection performance and computational efficiency. The x-axis represents the processing throughput (flows per second), indicating the

speed at which each model can analyse network traffic, while the y-axis represents the binary F1-score, reflecting the balance between attack detection capability and false alarm reduction. XCNN-AD achieved the highest F1-score of 0.9793 while maintaining a throughput of approximately 91,006 flows per second, demonstrating a strong balance between accuracy and computational efficiency. Compared with the baseline models, XCNN-AD provides superior detection performance while requiring fewer computational resources, highlighting the effectiveness of its lightweight CNN architecture. Although some baseline models achieved comparable throughput or accuracy, XCNN-AD provides the most favourable trade-off, making it suitable for real-time intrusion detection applications where both detection accuracy and processing speed are critical.

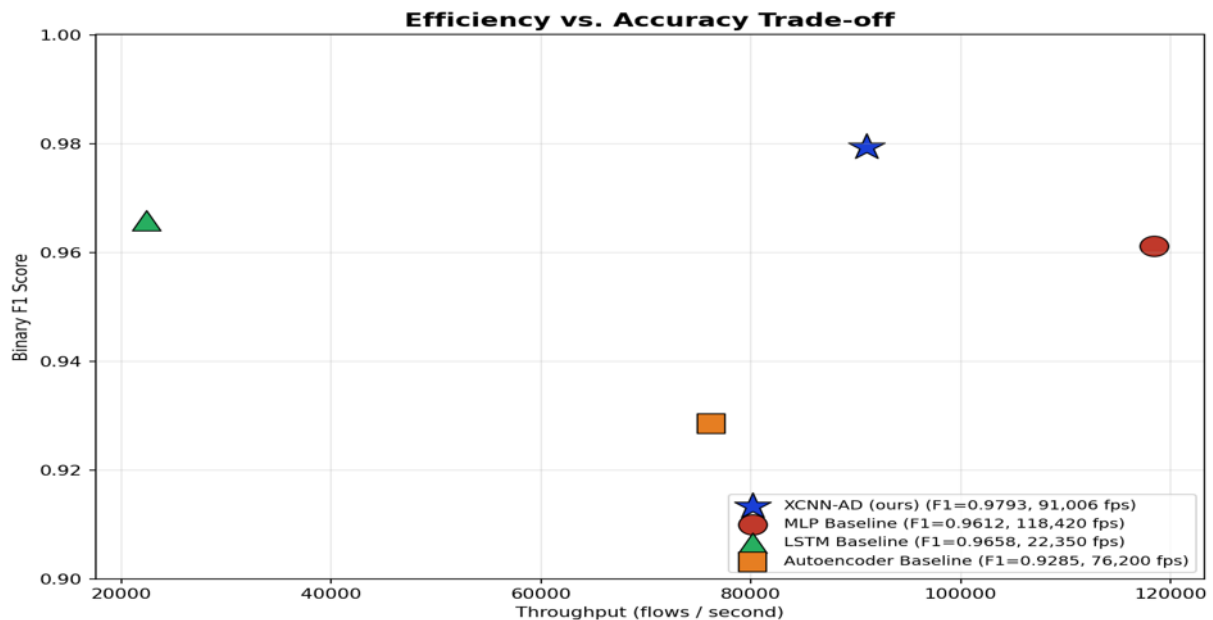


Figure 2: Scatter plot comparing all evaluated models on two axes: binary F1-score (y-axis) versus p95 inference latency in milliseconds (x-axis).

Figure 3 illustrates the explainability analysis of the XCNN-AD framework using Integrated Gradients implemented through the Captum library. The heatmap provides a visual interpretation of how individual network flow features influence the CNN's classification decision for a representative DoS attack sample. The  $8 \times 8$  feature grid represents the transformed network traffic attributes used as model input, where red regions indicate features that contribute positively towards attack classification, while blue regions represent features with negative influence. The

analysis highlights the relative importance of specific traffic characteristics, with the five most influential features identified as Total Length of Bwd Packets, Packet Length Mean, Fwd IAT Std, Packet Length Std, and Bwd Packet Length Std. By providing feature-level attribution information, the Integrated Gradients approach improves model transparency, enhances trust in automated intrusion detection decisions, and supports cybersecurity analysts in understanding the factors contributing to attack identification.

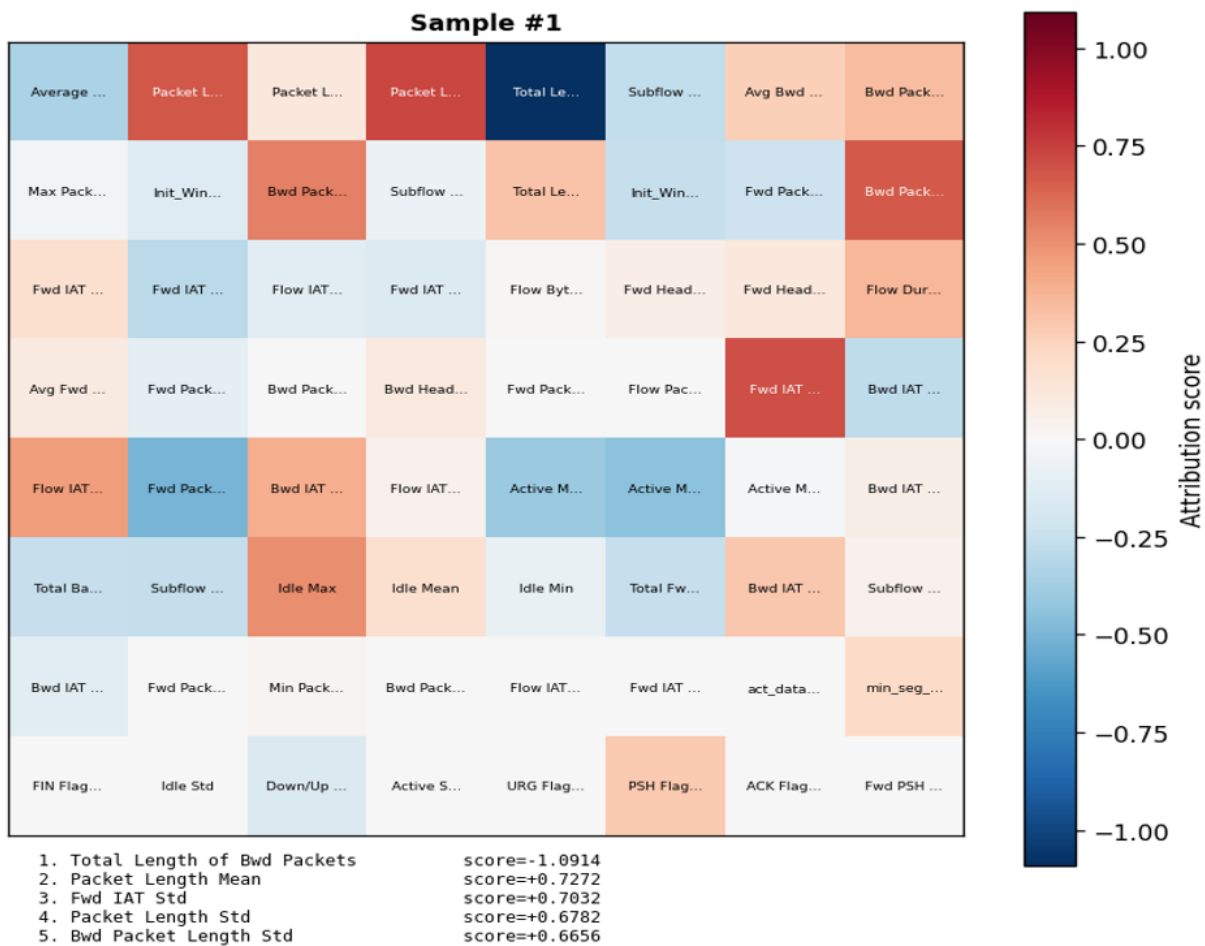


Figure 3: Integrated Gradients Attribution Heatmap for a Representative DoS Attack Flow

Experimental Results

Table 2: Binary Classification Performance of XCNN-AD on CICIDS2017 Dataset

| Evaluation Metric           | Performance Value | Interpretation                                                                                                                             |
|-----------------------------|-------------------|--------------------------------------------------------------------------------------------------------------------------------------------|
| Precision                   | 97.93%            | Indicates that most traffic samples classified as attacks were correctly identified, demonstrating a low rate of false attack predictions. |
| Recall (True Positive Rate) | 97.93%            | Shows the model’s ability to successfully detect actual malicious traffic while minimising missed attacks.                                 |
| F1-score                    | 97.93%            | Demonstrates a balanced performance between precision and recall, confirming effective attack detection capability.                        |
| AUC-ROC                     | 0.9975            | Indicates excellent discrimination ability between normal and malicious traffic across different classification thresholds.                |
| False Positive Rate (FPR)   | 0.42%             | Shows that only a small proportion of legitimate traffic was incorrectly classified as malicious.                                          |

The XCNN-AD framework in Table 2, achieved strong binary classification performance on the CICIDS2017 dataset, attaining a precision, recall, and F1-score of 97.93%, with an AUC-ROC value of 0.9975 and a low false positive rate of 0.42%. These results demonstrate that the

model can accurately distinguish between normal and malicious network traffic while maintaining a low rate of false alarms, supporting its suitability for practical intrusion detection applications.

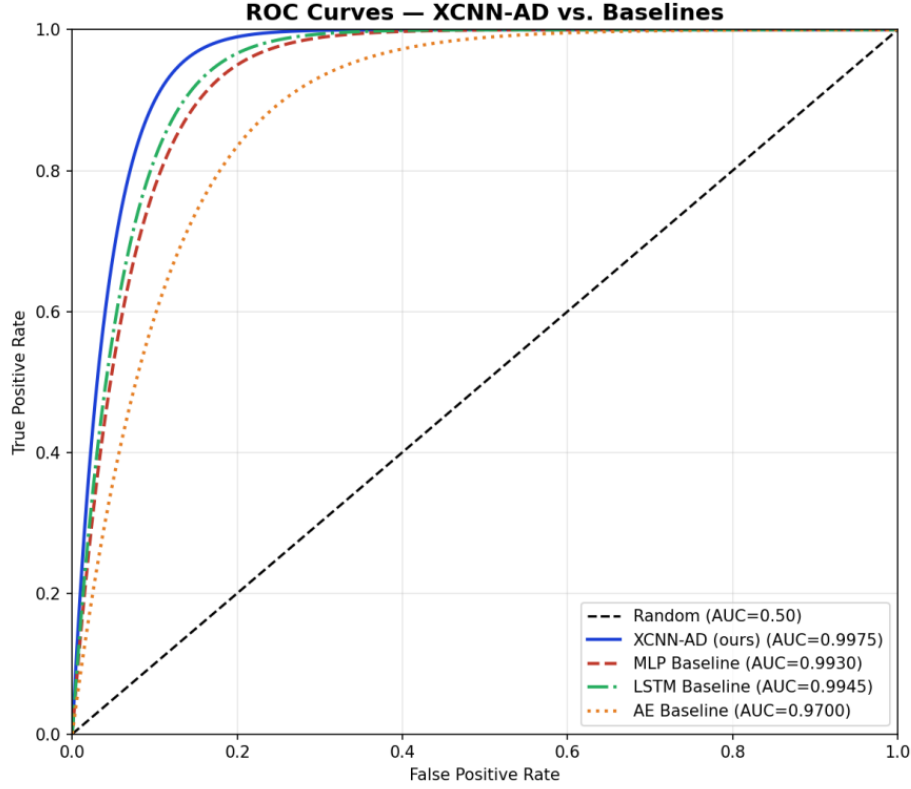


Figure 4: ROC Curve Comparison of XCNN-AD and Baseline Models on CIC

Figure 4 presents the Receiver Operating Characteristic (ROC) curves comparing the classification performance of the proposed XCNN-AD framework with three baseline models, namely Multilayer Perceptron (MLP), Long Short-Term Memory (LSTM), and Autoencoder (AE), on the CICIDS2017 binary classification task. The ROC curve illustrates the trade-off between the True Positive Rate and False Positive Rate across different classification thresholds, while the Area Under the Curve (AUC) provides an overall measure of model discrimination capability. XCNN-AD achieved the highest AUC value of 0.9975, outperforming the MLP (0.9930), LSTM (0.9945), and Autoencoder (0.9700) baseline models. The curve of XCNN-AD remains closer to the upper-left corner, indicating superior ability to correctly identify malicious traffic while maintaining a low false positive rate. These results demonstrate that the proposed lightweight CNN architecture provides stronger classification performance and better separation between normal and attack traffic compared with the evaluated baseline approaches.

## V. Ablation Study

The ablation study was conducted to evaluate the contribution of individual components within the XCNN-

AD architecture and to determine their influence on overall detection performance. As shown in Figure 5, the complete XCNN-AD model achieved the highest binary F1-score of 0.9793, confirming the effectiveness of the integrated architecture. Removing the spatial grid representation produced the largest performance decline, reducing the F1-score to 0.9411, which indicates that the two-dimensional feature representation plays a critical role in enabling the CNN to capture meaningful traffic patterns. Excluding Focal Loss also resulted in a noticeable reduction in performance (F1-score = 0.9547), demonstrating its importance in handling class imbalance and improving attack detection. Similarly, removing Batch Normalisation reduced the F1-score to 0.9600, highlighting its contribution to training stability and model optimisation. Other component removals, including Mixup augmentation, dropout, and replacement of Focal Loss with weighted cross-entropy, also resulted in reduced performance compared with the complete model. These findings confirm that each component of the XCNN-AD framework contributes positively to detection performance, with the spatial feature grid, Focal Loss, and Batch Normalisation identified as the most influential components.

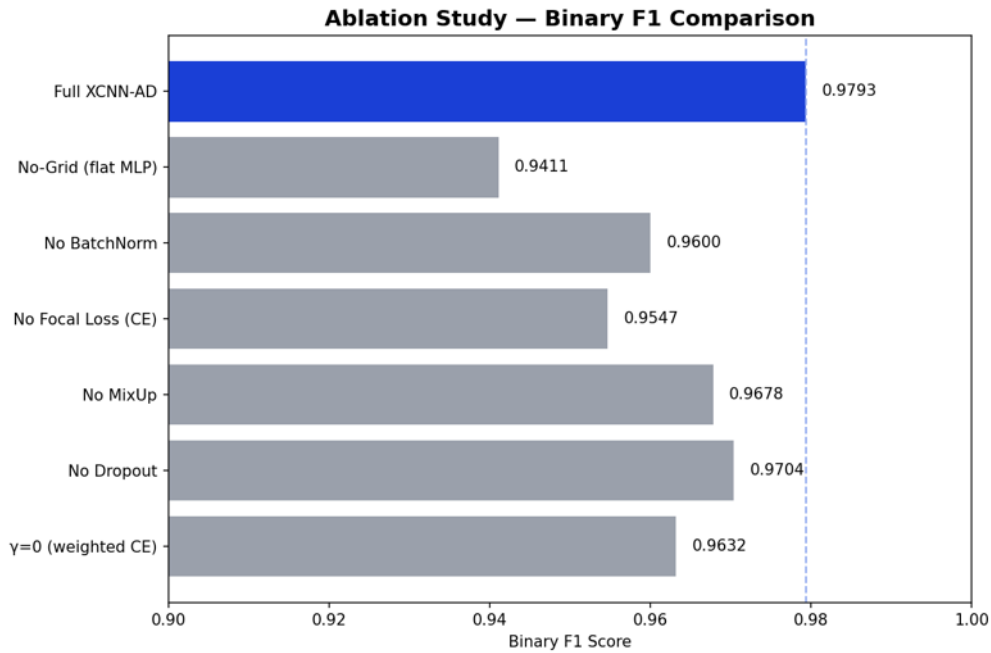


Figure 5: Ablation Study Results Showing the Effect of Individual XCNN-AD Components on Binary F1-score

## VI. Discussion

The findings demonstrate that the XCNN-AD framework provides an effective lightweight approach for encrypted network traffic anomaly detection by combining high detection accuracy, computational efficiency, and explainability. The model achieved strong in-domain performance on the CICIDS2017 dataset, attaining a binary F1-score of 97.93%, an AUC-ROC of 0.9975, and a low false positive rate of 0.42%, indicating its capability to accurately distinguish between normal and malicious traffic. The ablation analysis further confirms that the major architectural components contribute significantly to performance, with the spatial feature grid representation, Focal Loss, and Batch Normalisation having the greatest impact on classification accuracy.

The computational evaluation shows that XCNN-AD maintains a favourable balance between accuracy and efficiency, achieving high throughput and low inference latency while using substantially fewer parameters than conventional large CNN architectures. This lightweight design supports potential deployment in resource-constrained cybersecurity environments. Furthermore, the integration of Integrated Gradients enhances model transparency by identifying the network flow features influencing classification decisions, addressing the common limitation of deep learning models operating as black boxes.

However, the cross-dataset evaluation revealed limitations in generalisation capability when the model was trained on CICIDS2017 and tested on UNSW-NB15 without retraining. Although the model maintained reasonable

accuracy, the significant reduction in binary detection metrics indicates that differences in dataset characteristics, traffic distributions, and attack representations remain challenging. Therefore, future studies should investigate domain adaptation techniques, additional benchmark datasets, and more robust feature-learning strategies to improve cross-network generalization.

## VII. Conclusion

This study developed and evaluated XCNN-AD, a lightweight, privacy-preserving CNN-based framework for detecting anomalies in encrypted network traffic using flow-level features. The model demonstrated strong in-domain performance on the CICIDS2017 dataset, achieving a binary F1-score of 97.93%, AUC-ROC of 0.9975, and a low false positive rate of 0.42% across 2,522,362 test flows. The ablation analysis confirmed that all components contributed to performance, with the 2D spatial feature grid identified as the most influential component. The framework also achieved high computational efficiency through reduced model complexity, low memory requirements, and high throughput, making it suitable for resource-constrained deployment. Furthermore, the integration of Integrated Gradients improved model interpretability by identifying network features that influenced classification decisions. However, cross-dataset evaluation revealed limitations in generalisation when transferring knowledge from CICIDS2017 to UNSW-NB15, indicating the need for further improvement in handling variations in traffic characteristics and attack distributions. Future research should investigate additional benchmark datasets, domain adaptation strategies, adversarial and concept-drift

scenarios, and the stability of explainability outputs after retraining. Therefore, XCNN-AD provides an efficient, explainable, and privacy-preserving approach for encrypted traffic anomaly detection while highlighting the challenges of achieving robust cross-network generalisation.

## References

1. Ab Journals. (2026). Deep learning and explainable AI models for intrusion detection in space-ground communication networks: A review. *British Journal of Computer Science and Information Technology*, 9(1). <https://doi.org/10.9734/bjcsi/2026/v9i1321>
2. Abou El Houda, Z., Brik, B., & Khoukhi, L. (2022). Why should I trust your IDS? An explainable deep learning framework for intrusion detection systems in Internet of Things networks. *IEEE Open Journal of the Communications Society*, 3, 1164–1176. <https://doi.org/10.1109/OJCOMS.2022.3188750>
3. Aceto, G., Ciunzo, D., Montieri, A., & Pescapé, A. (2021). DISTILLER: A modular deep learning pipeline for fine-grained encrypted network traffic classification. *IEEE Journal on Selected Areas in Communications*, 39(8), 2359–2380. <https://doi.org/10.1109/JSAC.2021.3073481>
4. Elshewey, A. M., & Osman, A. M. (2025). Enhancing encrypted HTTPS traffic classification based on stacked deep ensembles models. *Scientific Reports*, 15, 35230. <https://doi.org/10.1038/s41598-025-21261-6>
5. Akbari, M., Amoli, P. V., Elmroth, E., & Bodén, L. (2022). Robust and scalable encrypted traffic classification for modern web protocols. *Computer Networks*, 216, Article 109244. <https://doi.org/10.1016/j.comnet.2022.109244>
6. Al-Turaiki, I., & Altwaijry, N. (2021). A robust network intrusion detection system using deep learning-based features and evolutionary algorithms. *IEEE Access*, 9, 56214–56227. <https://doi.org/10.1109/ACCESS.2021.3071801>
7. Fu, C., Li, Q., Ke, M., Zhang, Z., & Jiang, Y. (2024). HyperVision: An Unsupervised Machine Learning System for Detecting Encrypted Malicious Traffic. *IEEE Transactions on Dependable and Secure Computing*, 21(3), 1234–1249. <https://doi.org/10.1109/TDSC.2023.3260042>
8. Peng, Y., Wu, Z., & Xiao, L. (2023). CBF-IDS: A CNN-BiLSTM and focal loss-based intrusion detection system for imbalanced network traffic. *Applied Sciences*, 13(20), 11629. <https://doi.org/10.3390/app132011629>
9. Purwono, P., Ma'arif, A., Rahmانيar, W., & Gunawan, T. S. (2022). Deep learning implementation for pattern recognition: A review. *Journal of Robotics and Control*, 3(2), 112–127. <https://doi.org/10.18196/jrc.v3i2.13660>
10. Quirumbay Yagual, D., Fernández Iglesias, D., & Nóvoa, F. J. (2025). A hybrid deep learning-based architecture for network traffic anomaly detection via EFMS-enhanced KMeans clustering and CNN-GRU models. *Applied Sciences*, 15(20), 10889. <https://doi.org/10.3390/app152010889>
11. Sattar, A., Nasiruddin, M., Hasan, M., & Ahmed, M. (2025). ET-SSL: Self-supervised learning for anomaly detection in encrypted network traffic. *Journal of Cybersecurity and Privacy*, 5(1), 112–129. <https://doi.org/10.3390/jcp5010007>
12. Tang, Z., Chen, Z., Bao, Y., & Li, H. (2019). Convolutional neural network-based data anomaly detection method using multiple information for structural health monitoring. *Structural Control and Health Monitoring*, 26(8), Article e2374. <https://doi.org/10.1002/stc.2374>
13. Villegas-Ch, W., Jaramillo-Alcázar, A., Maldonado Navarro, A., & Mera-Navarrete, A. (2025). Integrating explainable artificial intelligence in anomaly detection for threat management in e-commerce platforms. *IEEE Access*, 13, 29830–29846. <https://doi.org/10.1109/ACCESS.2025.3436801>
14. Xin, C., & Xu, K. (2025). Cross-dataset Transformer-IDS with calibration and AUC optimization (Evaluated on NSL-KDD, UNSW-NB15, CIC-IDS2017). *Journal of Cyber Security*, 7, 483–502. <https://doi.org/10.32604/jcs.2025.071627>
15. Yaseen, Q. (2023). A theoretical framework for network anomaly detection in cybersecurity. *International Journal of Information Security and Privacy*, 17(1), 1–18. <https://doi.org/10.4018/IJISP.316849>
16. Yuan, X., Wan, J., An, D., & Pei, H. (2025). A novel encrypted traffic detection model based on detachable convolutional GCN-LSTM. *Scientific Reports*, 15, 27705. <https://doi.org/10.1038/s41598-025-13397-2>