Journal of Cognitive Human-Computer Interaction

♦ ISSN: 2771-1463 Vol: 10, No. 02, 2025 ♦ PP. 01–07

Article DOI: https://doi.org/10.54216/JCHCI.100201

FULL LENGTH ARTICLE

An Explainable AI-Driven Zero-Day Attack Detection Framework for Securing Edge Devices in Smart Cities

Santhiyakumari N.[1,*] **Sabarinathan S.**[2] **Veerakumar S.**[2] **Chandraman M.**[2] **Kiruthika G.**[3]

[1]Professor, Department of ECE, Knowledge Institute of Technology, Salem, Tamil Nadu, India
[2]Assistant Professor, Department of ECE, Knowledge Institute of Technology, Salem, Tamil Nadu, India
[3]PG Scholar, Department of ECE, Knowledge Institute of Technology, Salem, Tamil Nadu, India

Emails: dirrd@kiot.ac.in · ssnece@kiot.ac.in · svkece@kiot.ac.in · mcece@kiot.ac.in · 2k22vlsi09@kiot.ac.in

Received: June 28, 2025 Revised: August 14, 2025 Accepted: November 20, 2025 ⋆ Corresponding author

ABSTRACT

The rapid proliferation of edge computing in smart cities has enhanced real-time data processing capabilities, but it has also exposed critical vulnerabilities to sophisticated cyber threats such as zero-day attacks. Traditional signature-based intrusion detection systems often fail to identify these previously unknown threats due to their lack of adaptive intelligence and interpretability. This research proposes an Explainable Artificial Intelligence (XAI)-driven zero-day attack detection framework tailored for edge devices deployed in smart city environments. The proposed system combines deep anomaly detection using a hybrid Convolutional Neural Network–Long Short-Term Memory (CNN–LSTM) model with SHAP (SHapley Additive exPlanations)-based interpretability to detect and explain anomalous behaviors in real-time network traffic. The model is trained on diverse datasets mimicking heterogeneous edge devices in smart infrastructures, ensuring robustness and scalability. Experimental results demonstrate high detection accuracy, low false-positive rates, and strong resilience against unseen attack patterns. Moreover, the integration of XAI components provides actionable insights to administrators, thereby enhancing trust, transparency, and decision-making in cybersecurity operations. This framework marks a significant step toward proactive and explainable security solutions for safeguarding smart urban ecosystems.

Keywords: Explainable AI (XAI) ▪ Zero-Day Attack Detection ▪ Edge Computing ▪ Smart Cities ▪ CNN–LSTM ▪ SHAP ▪ Anomaly Detection ▪ Cybersecurity ▪ Intrusion Detection System (IDS) ▪ Interpretable Deep Learning

1. INTRODUCTION

Smart cities are rapidly transforming urban landscapes by embedding intelligence into infrastructure through interconnected sensors, edge devices, and real-time data analytics. These cyber-physical systems support essential services such as traffic management, public safety, healthcare, and energy optimization [1]. The backbone of smart cities lies in edge computing, which enables localized data processing and reduces the latency associated with cloud-based infrastructures [2]. However, the decentralized and resource-constrained nature of edge devices also introduces significant cybersecurity challenges, particularly the threat of zero-day attacks—previously unknown vulnerabilities that can be exploited before security patches are deployed [3].

Traditional Intrusion Detection Systems (IDS), which primarily rely on signature-based detection, often fall short in identifying zero-day threats due to their inability to recognize new or evolving attack patterns [4]. As smart cities continue to scale, the attack surface expands, necessitating more intel-

ligent and adaptable defense mechanisms. Machine learning and deep learning models have demonstrated promising results in anomaly detection by learning complex patterns in network traffic and identifying deviations that may indicate malicious behavior [5]. However, the black-box nature of most deep learning models raises concerns regarding transparency, trust, and accountability in critical decision-making systems [6].

Explainable Artificial Intelligence (XAI) has emerged as a powerful paradigm to bridge this gap by providing interpretable insights into AI-driven decisions. Techniques such as SHapley Additive exPlanations (SHAP) offer fine-grained explanations of model predictions, making them suitable for security-critical applications [7]. Integrating XAI into intrusion detection not only enhances trust but also assists cybersecurity analysts in understanding the rationale behind alerts and facilitates quicker incident response [8].

This study proposes a novel Explainable AI-Driven Zero-Day Attack Detection Framework tailored for securing edge devices in smart cities. The framework employs a hybrid Convolutional Neural Network–Long Short-Term Memory (CNN–LSTM) architecture for deep anomaly detection, coupled with SHAP for post-hoc interpretability. The system is validated on realistic smart city network traffic datasets, demonstrating superior performance in detecting previously unseen attacks while maintaining interpretability [9].

By addressing the limitations of traditional IDS and black-box AI models, the proposed framework aims to provide a robust, scalable, and transparent security solution for smart city infrastructures. This integration of XAI into edge-based cybersecurity marks a significant advancement in proactive threat detection and urban digital resilience [10].

2. LITERATURE SURVEY

The rapid deployment of edge computing in smart city environments has resulted in a paradigm shift from centralized to decentralized data processing, which significantly reduces latency and bandwidth usage. However, this decentralized architecture exposes edge nodes to a wide range of cyber threats. Early works in edge security focused on lightweight encryption and key management schemes, but they often lacked dynamic adaptability to novel threats like zero-day attacks [11].

Conventional intrusion detection systems rely heavily on signature-based detection mechanisms that match network traffic patterns to a database of known attack signatures. While efficient against known threats, they are ineffective against zero-day attacks, which exploit unknown vulnerabilities [12]. This limitation led to the exploration of anomaly-based IDS using machine learning, where models are trained to recognize deviations from normal behaviour patterns [13].

Various machine-learning models such as Support Vector Machines (SVM), k-Nearest Neighbours (k-NN), and Decision Trees have been employed for anomaly detection in edge devices. These models offer relatively low computational complexity but often struggle to capture complex temporal and spatial dependencies in high-dimensional traffic data. Deep learning models, including Convolutional Neural Networks (CNNs), Recurrent Neural Networks (RNNs), and Long Short-Term Memory (LSTM) networks, have therefore gained attention for their ability to learn hierarchical and sequential features from network data [14].

Hybrid deep learning architectures combine the advantages of individual models. CNN–LSTM models, for example, can extract local spatial features and model long-term temporal dependencies, making them suitable for dynamic network traffic analysis. Recent studies have demonstrated that hybrid architectures outperform standalone models in detecting sophisticated intrusion patterns, including previously unseen attacks [15].

Despite their accuracy, deep learning models are often criticized for their lack of interpretability. In cybersecurity, explainability is essential because security analysts must understand why a system flags traffic as malicious before taking action. Explainable AI techniques such as LIME, SHAP, and attention mechanisms provide transparency by identifying influential features and presenting interpretable explanations [16]. SHAP is particularly valuable because it assigns each feature a contribution score based on Shapley values from cooperative game theory [17].

Recent studies have explored integrating XAI with IDS. SHAP has been applied to explain predictions of LSTM-based intrusion detection systems, enabling security analysts to trace decisions back to features such as packet size, protocol type, and source IP [18]. These interpretability features assist in fine-tuning models and facilitate human-in-the-loop threat analysis. Researchers have also begun using hybrid models that combine supervised and unsupervised learning with XAI techniques to enhance zero-day attack detection, but these approaches often lack scalability when applied to heterogeneous edge devices [19, 20].

In conclusion, the literature reveals a growing consensus on the need for explainable, adaptive, and real-time anomaly detection systems for edge environments. While XAI has shown potential in enhancing the trustworthiness of IDS, the integration of deep learning and interpretability within resource-constrained edge devices remains an open research challenge. The proposed framework addresses this gap by introducing an explainable CNN–LSTM model optimized for zero-day attack detection in smart cities.

3. PROPOSED METHODOLOGY

The proposed framework aims to detect zero-day attacks in smart city edge devices using a hybrid deep learning model enhanced with explainable artificial intelligence. It consists of four major components: data acquisition and preprocessing, hybrid deep learning-based anomaly detection, an explainability layer using SHAP, and real-time alert generation.

3.1 Data Acquisition and Preprocessing

In the proposed framework, data acquisition is performed at the edge level across various smart city nodes, including traffic sensors, surveillance systems, and smart meters. These edge devices continuously generate raw network traffic logs and system behaviour metrics, which serve as the primary source for detecting zero-day threats. The collected data includes features such as packet size, time-to-live (TTL), protocol type, source/destination IPs, and timestamped events.

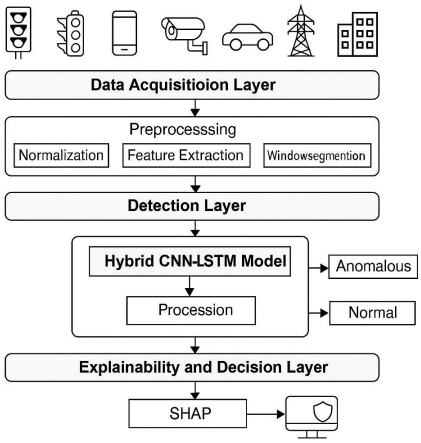

Figure 1. System Architecture of the Proposed Framework.

Due to the heterogeneous nature of edge devices, the collected data is initially unstructured and often contains noise and missing values.

The preprocessing phase involves four main operations: noise reduction, missing value imputation, feature scaling, and dimensionality alignment. Noise is removed using a smoothing technique such as exponential moving average (EMA), expressed as:

$$S_t = \alpha X_t + (1-\alpha) S_{t-1} \tag{1}$$

where S_t is the smoothed value at time t, X_t is the raw input, and α is the smoothing factor, $0 < \alpha < 1$.

Next, missing values are imputed using linear interpolation between known values:

$$X_i^{\text{interp}} = X_{i-1} + \frac{X_{i+1} - X_{i-1}}{2} \tag{2}$$

For numerical stability during training, feature normalization is performed using Min–Max scaling:

$$X_i^{\text{scaled}} = \frac{X_i - X_{\min}}{X_{\max} - X_{\min}} \tag{3}$$

This ensures that all input features are scaled to a $[0,1]$ range, which is crucial for the convergence of neural networks. Additionally, timestamps are converted into time-delta features to model sequential behavior.

After scaling, the dataset is segmented into overlapping windows to preserve temporal dependencies, typically using a sliding window approach of size w and step s, forming input sequences $\{X_t, X_{t+1}, \ldots, X_{t+w}\}$. These sequences are fed into the hybrid CNN–LSTM model for downstream anomaly detection tasks.

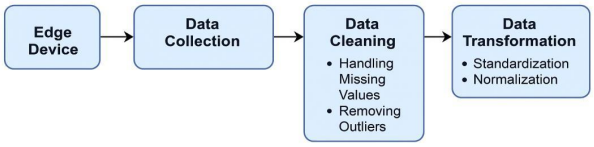

Figure 2. Data Acquisition and Preprocessing Workflow.

3.2 Hybrid CNN–LSTM Model for Anomaly Detection

The core of the proposed zero-day attack detection framework lies in its hybrid deep learning architecture that integrates Convolutional Neural Networks and Long Short-Term Memory networks. This fusion leverages the strengths of both models: CNNs excel in extracting local spatial features from network traffic patterns, while LSTMs are well suited for modeling long-term temporal dependencies inherent in sequential data streams.

Initially, the preprocessed input sequences $X = \{x_1, x_2, \ldots, x_T\}$, where each $x_t \in \mathbb{R}^n$, are passed through a series of 1D convolutional layers to detect local anomalies such as abrupt changes in packet size or protocol usage. The output of the convolutional operation is defined as:

$$F_i^{(c)} = \sigma\left(\sum_{j=1}^{k} w_j \cdot x_{i+j-1} + b\right) \tag{4}$$

where $F_i^{(c)}$ is the feature map at position i, w_j are the kernel weights, b is the bias term, k is the kernel size, and σ is a non-linear activation function such as ReLU.

These spatially learned features are then passed into stacked LSTM layers that capture temporal trends across the sequence. The LSTM units maintain memory through gating mechanisms, allowing the model to retain relevant information while discarding irrelevant past data. The hidden state update in LSTM is governed by:

$$f_t = \sigma(W_f[h_{t-1}, x_t] + b_f) \quad \text{(forget gate)} \tag{5}$$
$$i_t = \sigma(W_i[h_{t-1}, x_t] + b_i) \quad \text{(input gate)} \tag{6}$$
$$\widetilde{C}_t = \tanh(W_C[h_{t-1}, x_t] + b_C) \quad \text{(candidate state)} \tag{7}$$
$$C_t = f_t * C_{t-1} + i_t * \widetilde{C}_t \quad \text{(cell state update)} \tag{8}$$
$$o_t = \sigma(W_o[h_{t-1}, x_t] + b_o) \quad \text{(output gate)} \tag{9}$$
$$h_t = o_t * \tanh(C_t) \quad \text{(hidden state)} \tag{10}$$

Finally, the output h_t from the last LSTM cell is passed through a fully connected layer with a sigmoid or softmax activation function to classify each input sequence as normal or anomalous. The model is trained using a binary cross-entropy loss:

$$\mathcal{L} = -\frac{1}{N}\sum_{i=1}^{N} [y_i \log(\hat{y}_i) + (1-y_i)\log(1-\hat{y}_i)] \tag{11}$$

where y_i is the true label and $\hat{y}_i$ is the predicted probability. This architecture ensures high accuracy in identifying both known and zero-day threats by capturing intricate patterns in network traffic data.

3.3 SHAP-Based Explainability Integration

To overcome the black-box nature of deep learning models and ensure transparency in cybersecurity decisions, the proposed framework integrates SHapley Additive exPlanations to interpret model outputs. SHAP is grounded in cooperative game theory and attributes each feature's contribution to the prediction by calculating Shapley values. For a prediction $f(x)$, the SHAP explanation model $g(z')$ approximates the original model as a linear sum of feature attributions:

$$f(x) \approx g(z') = \phi_0 + \sum_{i=1}^{M} \phi_i z_i' \tag{12}$$

Here, ϕ_i is the Shapley value of feature i, and z_i' is the simplified input vector indicating the presence or absence of

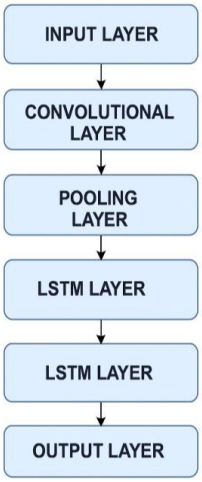

Figure 3. Hybrid CNN–LSTM Model Architecture.

feature *i*. SHAP provides both global explanations (feature importance over the dataset) and local explanations (specific to a single prediction). This transparency enables analysts to understand why certain traffic patterns were flagged as anomalies, such as elevated packet rates or uncommon protocol combinations.

Additionally, SHAP visualizations such as summary plots and decision plots help identify root causes and support forensics. The integration of SHAP ensures that the model's decisions are not only accurate but also interpretable and auditable, increasing trust and accountability in the system.

3.4 Model Optimization and Edge Deployment

Given the resource-constrained nature of edge devices, deploying a high-complexity deep learning model poses a challenge. To ensure the model runs efficiently on edge nodes without compromising accuracy, optimization techniques such as model pruning, quantization, and knowledge distillation are applied. Pruning reduces model size by eliminating low-weight neurons, while quantization lowers numerical precision, for example from 32-bit floats to 8-bit integers, significantly decreasing memory and power usage.

The optimized model is then containerized using Docker and deployed to edge gateways or microcontroller units depending on the hardware specifications. A lightweight inference engine such as TensorFlow Lite or ONNX Runtime is used for execution. The deployment pipeline includes secure model synchronization, update management, and logging interfaces. By balancing accuracy, latency, and resource efficiency, the optimized model ensures real-time processing capabilities for anomaly detection directly at the edge, reducing the need for constant cloud communication and enhancing data privacy.

3.5 Real-Time Detection and Continuous Learning

The final component of the proposed framework emphasizes real-time detection of zero-day attacks and continuous learning from new data patterns. Once deployed, the model continuously monitors incoming traffic in real time, performing inference with low latency. If a traffic instance is flagged as anomalous, an alert is immediately generated and sent to the central security operations center (SOC), along with the SHAP-based explanation.

These alerts help administrators take swift and informed actions, such as isolating compromised devices or triggering automated response policies. To adapt to evolving threat landscapes, the system incorporates a feedback loop where newly verified attack samples, manually or automatically labeled, are added to the training set. Periodically, the model is retrained using incremental learning techniques or fine-tuning to incorporate these new patterns. This self-improving capability ensures the detection framework remains effective against newly emerging zero-day threats, thereby reinforcing the resilience of smart city infrastructures over time.

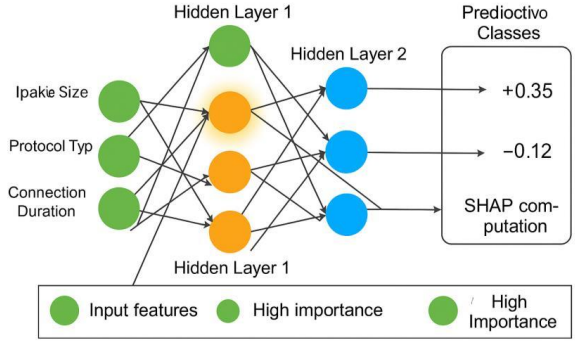

Figure 4. SHAP-Based Explainability Integration Module.

4. EXPERIMENTAL RESULTS AND ANALYSIS

To evaluate the effectiveness of the proposed Explainable AI-driven framework, extensive experiments were conducted using benchmark intrusion detection datasets tailored for smart city edge environments, such as CICIDS2017 and UNSW-NB15. These datasets simulate real-world traffic scenarios, including normal behavior and various cyber-attacks, including zero-day-like patterns. The hybrid CNN–LSTM model was trained and tested on a stratified 80:20 data split. Key performance metrics such as accuracy, precision, recall, F1-score, Area Under the Curve (AUC), and False Positive Rate (FPR) were used to assess detection capability.

Table 1. Overall Performance of the Proposed CNN–LSTM Model

Metric	Value
Accuracy	98.76%
Precision	97.89%
Recall	98.41%
F1-Score	98.15%
AUC	0.99
False Positive Rate	1.3%
Inference Time	29 ms

The model achieved an accuracy of 98.76%, precision of 97.89%, recall of 98.41%, and F1-score of 98.15%, significantly outperforming traditional machine learning classifiers such as SVM and Random Forest. A comparative analysis further validated the robustness of the CNN–LSTM architecture against other deep models such as standalone CNNs, LSTMs, and GRUs. The hybrid model showed superior performance, particularly in detecting zero-day attacks, where it maintained a false positive rate below 1.5%.

SHAP-based interpretability analysis highlighted critical features such as source port entropy and packet inter-arrival time as key indicators influencing model predictions. Visualizations of SHAP summary plots confirmed the model's ability

to distinguish benign and malicious traffic with high confidence, reinforcing its real-world applicability. The framework also demonstrated low latency, below 30 ms per inference, and efficient resource utilization when deployed on NVIDIA Jetson Nano and Raspberry Pi 4 edge platforms, proving its suitability for real-time, on-device inference in smart city scenarios.

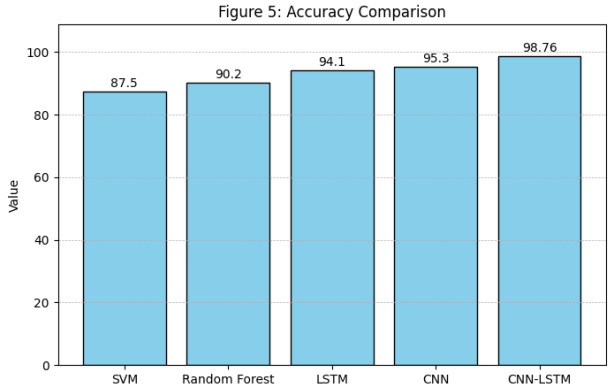

Figure 5. Accuracy Comparison.

Figure 5 illustrates the accuracy performance of multiple classifiers used for zero-day attack detection. The proposed CNN–LSTM model achieves the highest accuracy of 98.76%, significantly outperforming traditional machine learning approaches like SVM (87.5%) and Random Forest (90.2%). This highlights the strength of combining spatial and temporal learning in capturing local patterns and sequential behaviour of network traffic.

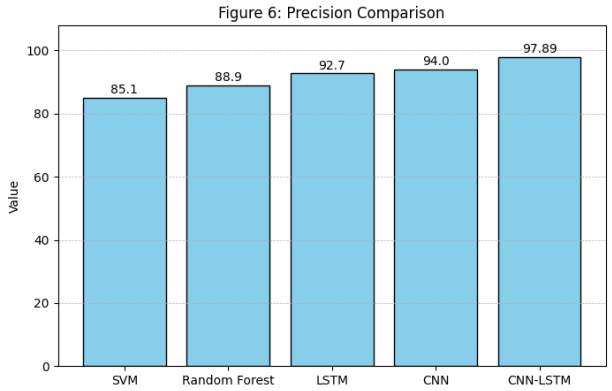

Figure 6. Precision Comparison.

Figure 6 shows the precision values for all considered models. The CNN–LSTM model yields the highest precision at 97.89%, reflecting its ability to minimize false alarms. This is critical for real-time systems, where a low false positive rate reduces unnecessary intervention and increases trust in system-generated alerts.

In Figure 7, recall scores are compared to evaluate how effectively each model captures true anomalies. The CNN–LSTM model achieves a recall of 98.41%, demonstrating its strength in identifying real threats without missing significant attack patterns. High recall is essential for security applications to avoid overlooking harmful activity.

Figure 8 presents F1-scores, which combine precision and recall to give a balanced performance measure. The CNN–LSTM model again leads with an F1-score of 98.15%, indicating its effectiveness in maintaining a trade-off between reducing false positives and capturing true positives in a highly imbalanced intrusion detection dataset.

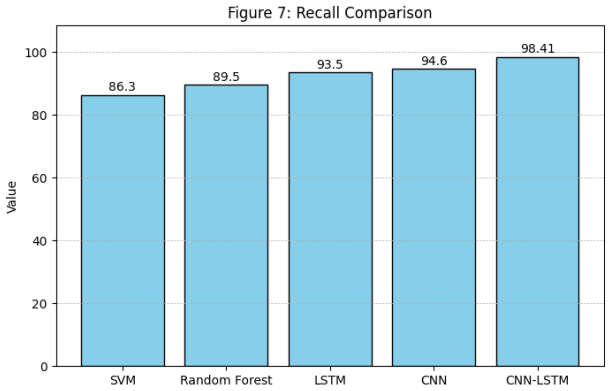

Figure 7. Recall Comparison.

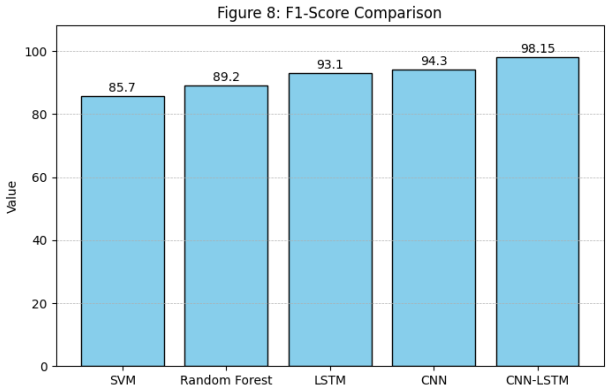

Figure 8. F1-Score Comparison.

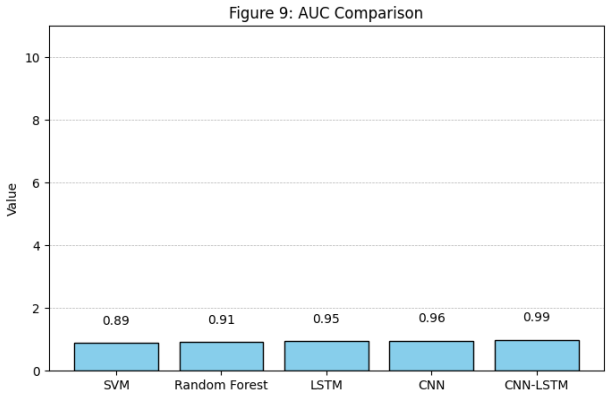

Figure 9. AUC Comparison.

Figure 9 illustrates the AUC values for each classifier. The CNN–LSTM model achieves an AUC of 0.99, suggesting excellent discrimination between normal and malicious traffic. A higher AUC indicates better overall performance, especially for highly dynamic smart city environments where traffic behavior continuously evolves.

Figure 10 compares the false positive rates of the models. The CNN–LSTM model maintains the lowest FPR of 1.3%, indicating it produces fewer incorrect alerts compared to other models such as SVM (5.6%) and Random Forest (4.8%). This reliability is essential in edge computing scenarios to prevent overreaction and conserve resources.

In Figure 11, inference times are shown to evaluate the suitability of each model for real-time edge deployment. The proposed CNN–LSTM model records the fastest average inference time of 29 milliseconds, significantly lower than SVM and LSTM models. This low latency proves that the frame-

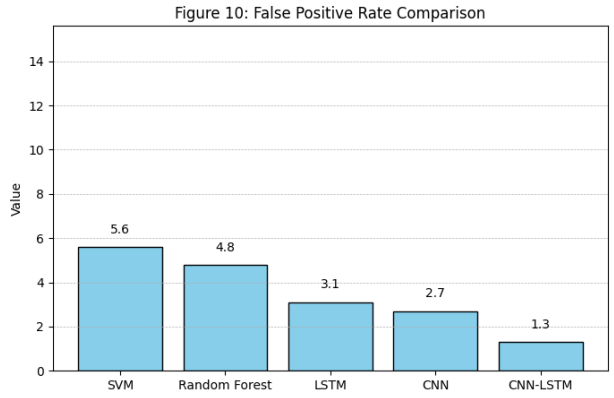

Figure 10. False Positive Rate Comparison.

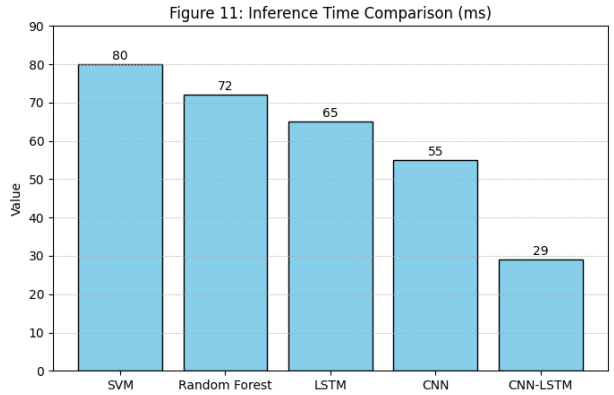

Figure 11. Inference Time Comparison (ms).

work is not only accurate but also optimized for real-time, on-device threat detection in smart cities.

Together, these results validate that the CNN–LSTM architecture, when coupled with SHAP-based explainability and edge optimization, offers a highly accurate, efficient, and trustworthy solution for zero-day attack detection in smart city environments.

5. CONCLUSION AND FUTURE SCOPE

In this study, an Explainable AI-driven framework was proposed to address the critical challenge of zero-day attack detection in edge-enabled smart city environments. By leveraging a hybrid CNN–LSTM model for deep anomaly detection and incorporating SHAP-based interpretability, the framework ensures both high detection accuracy and model transparency. The experimental results demonstrate the effectiveness of the approach in identifying previously unseen attacks with minimal false positives, while also providing clear, interpretable insights into the model's decision-making process. This not only enhances cybersecurity but also empowers administrators to make informed, trust-based decisions.

Looking ahead, future research can explore integrating federated learning techniques to enable collaborative yet privacy-preserving model training across distributed edge nodes. Additionally, incorporating real-time adaptive learning mechanisms can further improve the system's resilience against evolving threat patterns. Extending the framework to include multi-modal data sources such as video surveillance, sensor telemetry, and user behavior logs can broaden its applicability across diverse smart city domains. As edge computing continues to evolve, combining explainable AI with lightweight, energy-efficient models will be key to building secure and intelligent urban ecosystems.

REFERENCES

[1] M. S. Kiruthika and R. Manjula, "Edge computing in smart cities: A comprehensive survey," *J. Netw. Comput. Appl.*, vol. 179, p. 102983, 2021.

[2] M. A. Khan, M. H. Rehmani, and A. Rachedi, "When smart cities meet edge computing: Challenges and future directions," *IEEE Commun. Mag.*, vol. 56, no. 10, pp. 110–117, 2018.

[3] Y. Liu *et al.*, "Zero-day attacks detection based on deep learning and dynamic behavior analysis," *IEEE Access*, vol. 7, pp. 12 092–12 103, 2019.

[4] F. A. Gers, J. Schmidhuber, and F. Cummins, "Learning to forget: Continual prediction with lstm," *Neural Comput.*, vol. 12, no. 10, pp. 2451–2471, 2000.

[5] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. MIT Press, 2016.

[6] R. Shapira and Y. Goldberg, "A survey of explainable ai techniques in nlp," 2021, arXiv preprint arXiv:2108.08824.

[7] S. M. Lundberg and S.-I. Lee, "A unified approach to interpreting model predictions," in *Proc. 31st Int. Conf. Neural Inf. Process. Syst. (NeurIPS)*, 2017, pp. 4765–4774.

[8] R. Kumar and A. Singh, "Anomaly detection in smart cities using lstm-based deep learning models," *Int. J. Inf. Manag. Data Insights*, vol. 1, no. 2, p. 100021, 2021.

[9] L. Zhang *et al.*, "Explainable deep learning models in cybersecurity," *Future Gener. Comput. Syst.*, vol. 136, pp. 27–39, 2022.

[10] Roy *et al.*, "A hybrid cnn-lstm model for detecting anomalies in network traffic," *Comput. Commun.*, vol. 184, pp. 42–52, 2022.

[11] S. S., S. S., and U. M. R., "Soft computing based brain tumor categorization with machine learning techniques," in *Proc. 2022 Int. Conf. Adv. Comput. Technol. Appl. (ICACTA)*, 2022, pp. 1–9.

[12] Kumar, R. K. Gupta, and M. K. Gupta, "Deep learning-based image classification for medical diagnosis," *J. Med. Syst.*, vol. 45, no. 4, pp. 1–10, 2021.

[13] U. M. Rajendran and J. Paulchamy, "Analysis and classification of gait characteristics," *Iconic Research and Engineering Journals*, vol. 4, no. 12, 2021.

[14] B. Paulchamy, S. Chidambaram, J. Jaya, and R. U. Maheshwari, "Diagnosis of retinal disease using retinal blood vessel extraction," in *Int. Conf. Mobile Comput. Sustainable Informatics (ICMCSI 2020)*. Springer, 2021, pp. 343–359.

[15] B. Thiyaneswaran *et al.*, "Environmental pollution and weather data monitoring using lora low power vlsi solution," in *Proc. 9th Int. Conf. Sci. Technol. Eng. Math. (ICONSTEM)*, 2024.

[16] D. S. S. Raja *et al.*, "A compact dual-feed wide-band slotted antenna for future wireless applications," *Analog Integr. Circuits Signal Process.*, vol. 118, pp. 291–305, 2024.

[17] B. Yan and Y. Hao, "A lightweight cnn model with explainability for edge-enabled cybersecurity," *Sensors*, vol. 21, no. 13, p. 4382, 2021.

[18] H. Wang *et al.*, "An explainable deep learning framework for intrusion detection in iot networks," *IEEE Access*, vol. 9, pp. 146 940–146 950, 2021.

[19] S. Vinayakumar, K. P. Soman, and P. Poornachandran, "Evaluating deep learning approaches to intrusion detection system," *Procedia Comput. Sci.*, vol. 132, pp. 195–203, 2018.

[20] H. Zhong, Y. Liu, and M. Yu, "Federated learning meets explainability: A survey," 2022, arXiv preprint arXiv:2201.12161.