Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 257 (2025) 569–574

Procedia Computer Science

www.elsevier.com/locate/procedia

The 16th International Conference on Ambient Systems, Networks and Technologies (ANT)
April 22-24, 2025, Patras, Greece

Enhancing Explainability of Artificial Intelligence for Threat Detection in SDN-based Multicast Systems

Preety Prasad*, Mohammad Tahir, Jouni Isoaho

Department of Computing, University of Turku, Turku 20014, Finland

Abstract

The increasing adoption of Artificial Intelligence (AI) based Software-Defined Networking (SDN) in multicast systems has improved network management and traffic efficiency. However, for network administrators, understanding AI outcomes and explanations for how conclusions are reached in threat detection and mitigation is essential for strengthening their overall security framework. Additionally, centralized control planes in SDN introduce new security challenges, which can complicate the detection and mitigation of various network threats. In this regard, this paper presents a novel framework that integrates Explainable AI (XAI) with SDN to detect and mitigate threats in real-time. The proposed framework leverages a hybrid machine learning model, using Convolutional Neural Networks (CNN) for analyzing network traffic features and Long Short-Term Memory (LSTM) networks for identifying patterns and anomalies. To enhance transparency and explanation for the threat detection process, the framework incorporates both LIME (Local Interpretable Model-Agnostic Explanations) and SHAP (SHapley Additive exPlanations). LIME provides local explanations by generating perturbed data instances and training a surrogate model to identify the most influential features in a specific prediction. This allows the network administrators to understand how different network features contribute to the classification decision. SHAP, on the other hand, quantifies the contribution of each feature to the overall model decision by computing Shapley values, offering a global perspective of feature importance. This approach offers a more effective and transparent solution for SDN systems in a multicast environment, improving threat detection and security.

Keywords: Explainable AI (XAI), Threat Detection, Software-Defined Networking, LIME (Local Interpretable Model-Agnostic Explanations), Multicast Traffic, SHAP (SHapley Additive exPlanations), Network Security, Intrusion Detection Systems, Convolutional Neural Networks (CNN), Long Short-Term Memory (LSTM).

1. Introduction

Software-defined networking (SDN) has transformed multicast audio and video transmissions by enabling centralized traffic control and dynamic adaptability. While ideal for high-demand, low-latency scenarios, this centralization

* Corresponding author. Tel.: +358-29-450-5000 ; fax: +358-29-450 5040.
E-mail address: preety.p.prasad@utu.fi

1877-0509
10.1016/j.procs.2025.03.073

exposes the SDN controller to critical threats like DDoS attacks, data breaches, and packet manipulation. Ensuring secure and uninterrupted multicast transmission is vital to addressing these vulnerabilities.

AI-based SDN-enabled multicast systems for threat detection face challenges such as analyzing non-linear multicast patterns, real-time group membership changes, load balancing, and limited transparency in decision-making [1]. The dynamic nature of multicast traffic and frequent membership changes complicate threat detection, while the ”black-box” nature of AI models hinders troubleshooting, regulatory compliance, and prediction validation. The lack of explainability increases susceptibility to adversarial attacks, emphasizing the need for Explainable AI. XAI enhances transparency by providing insights into AI decisions and supporting error correction and compliance. In critical scenarios, like DDoS attacks, XAI builds trust, improves fairness, and ensures accountability in SDN-enabled multicast environments [2]. Existing research focuses on neuro-symbolic AI to improve interpretability and decision-making, although its performance in real-world SDN systems remains unexplored in terms of transparency [3]. However, these studies primarily address general-purpose or unicast traffic and often rely on external systems for explainability, leading to increased latency and overhead [4]. Furthermore, multicast-specific challenges, such as dynamic group membership changes and nonlinear traffic patterns must be addressed during the network traffic flow to multiple receivers(e.g. video transmissions).

To address these limitations, this paper introduces a novel framework incorporating an hybrid CNN-LSTM for intrusion detection mechanism tailored specifically for SDN-based multicast systems. The hybrid model leverages Convolutional Neural Networks (CNN) for spatial feature extraction and Long Short-Term Memory (LSTM) networks for temporal pattern recognition. For explainability, the framework integrates LIME (Local Interpretable Model-agnostic Explanations) and SHAP (SHapley Additive exPlanations) to create surrogate models that clarify why certain traffic flows are flagged as threats, ensuring trust and transparency in decision-making. Embedding the IDS mechanism directly within the SDN controller improves responsiveness and ensures seamless alignment with the network architecture. This research addresses multicast-specific challenges and leverages XAI to deliver actionable, real-time insights, bridging the gap between advanced analytics and practical network security. This article aims to addresses the following key research questions:

- ***Research Question 1 (RQ1):*** How can explainable techniques LIME and SHAP both be incorporated in AI-driven SDN-based multicast environments to improve the interpretability and transparency?
- ***Research Question 2 (RQ2):*** How can the machine learning model in XAI be used to detect threats in real-time and high-volume SDN multicast traffic efficiently?

The rest of the article is organized as follows: Section 2 introduces the proposed framework, which integrates an Explainable AI model within the SDN multicast environment. Section 3 explains the integration of the ML hybrid model with the explainability engine (LIME and SHAP) to support the decision-making process. Finally, Section 4 concludes the paper and summarizes its key findings.

2. Framework: Explainable AI model integrated with SDN Multicast environment

To bridge the gap between existing research and the proposed framework, it is essential to highlight how the combination of LIME and SHAP, alongside a hybrid CNN-LSTM model, uniquely addresses the challenges in multicast traffic management. While SHAP-based approaches, such as SHAP with Pattern Dependency (SHAPPD), have been successful in enhancing explainability in LSTM-based models for DDoS classification by quantifying feature contributions and identifying interdependencies, these methods largely focus on unicast traffic or generalized threat scenarios [5]. Similarly, XAI-IDS framework enhances network intrusion detection by integrating AI models with XAI techniques to improve interpretability and performance. However, it has limitations, including the lack of SDN integration, reliance on traditional machine learning rather than advanced neural networks, no consideration for multicast traffic, potential scalability issues, limited real-time adaptation, and absence of explicit time-series data handling. Additionally, it faces challenges in integrating with existing systems and lacks parallel processing for LIME and SHAP. These gaps highlight opportunities for further research in combining XAI with advanced network architectures, particularly in SDN and multicast environments [6]. Furthermore, research on LIME and SHAP in network security has shown their potential to improve decision-making transparency, yet the integration of these methods in

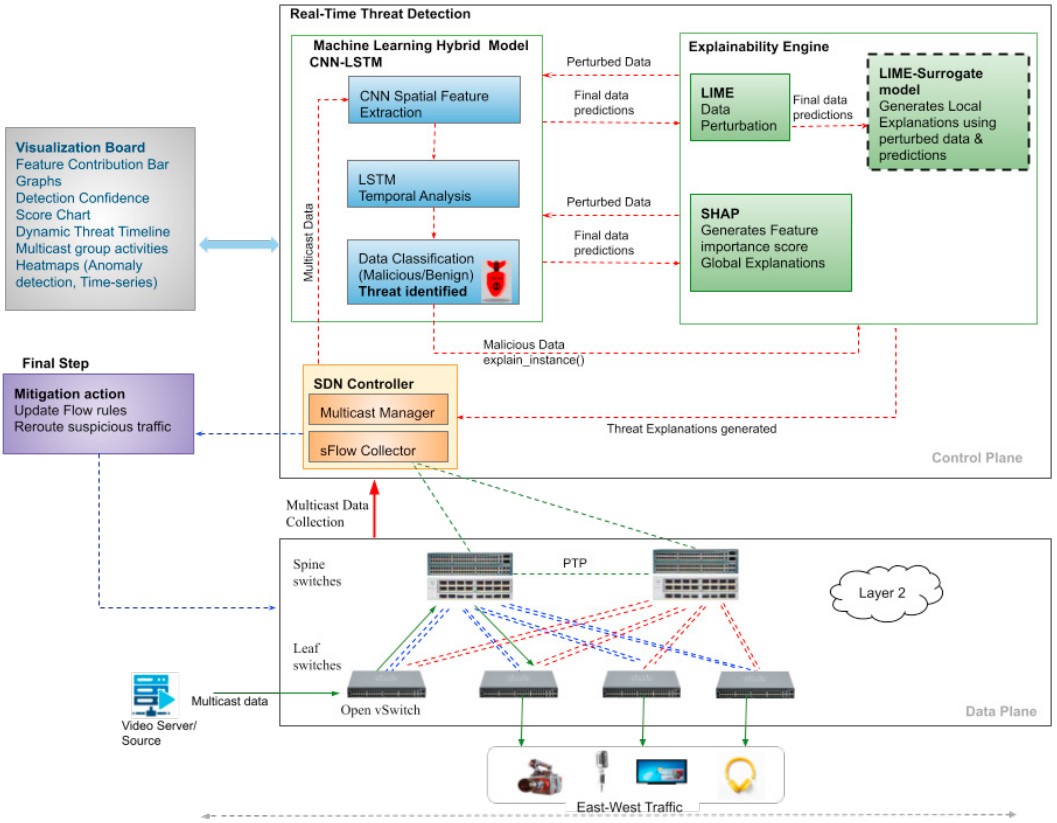

Fig. 1: Proposed XAI-Enabled SDN Framework Integrated with ML Model CNN-LSTM for Threat Detection in Multicast Systems

multicast-specific scenarios remains unexplored [7].

Multicast traffic introduces unique challenges, such as dynamic group memberships, synchronized delivery to multiple receivers, and the need for low-latency threat detection. Existing models do not adequately address these multicast-specific intricacies. Recognizing this gap, the proposed framework integrates LIME and SHAP with a CNN-LSTM model for multicast environments, balancing accuracy and interpretability while providing actionable insights into traffic behavior and anomalies.

As shown in Fig. 1, the framework enhances existing solutions by targeting multicast-specific features and enabling real-time, transparent decision-making. Furthermore, it improves threat detection, prevention, and system trustworthiness, making it ideal for modern, latency-sensitive applications like video streaming. Following are its components:

- *Data Plane (Forwarding Layer)*: The data plane in an SDN environment consists of OpenFlow switches and routers that monitor multicast traffic by capturing metadata such as packet headers, flow rates, source IPs, destination group addresses, and protocol details. Incoming multicast packets are validated using Reverse Path Forwarding (RPF) techniques to ensure they follow the expected upstream path, with invalid packets being dropped to prevent spoofing. Protocol Independent Multicast (PIM) [8] builds and maintains multicast distribution trees, while the Internet Group Management Protocol (IGMP) manages dynamic host membership [9]. The OpenFlow protocol generates traffic statistics reports, including packet counts and link utilization, and sends them to the SDN controller.
- *SDN Controller:* The SDN controller aggregates metadata from the data plane and organizes it into structured formats(feature vectors or matrices) for machine learning processing. Based on the analysis and feedback, the controller dynamically updates flow tables to optimize multicast routing and ensure efficient resource allocation.

- *Intrusion Detection Mechanism (CNN-LSTM):* This machine learning model processes structured network traffic data (e.g., packet counts, multicast group addresses, flow rates, TTL (Time-To-Live)values) by using CNN to extract spatial features, such as correlations between packet headers and multicast flow patterns. LSTM networks analyze temporal dependencies, detecting abnormal trends or sequences over time, such as sudden spikes in multicast traffic or irregular group memberships. The model subsequently classifies traffic instances as either benign or malicious, detecting threats such as Distributed Denial-of-Service (DDoS) attacks, spoofed multicast flows, or abnormal levels of multicast traffic.
- *Explainability Engine:* Suspicious traffic flagged by the CNN-LSTM model is analyzed by the Explainability Engine, where both XAI techniques can be executed simultaneously but for different purposes. SHAP provides a global explanation with a network-wide view, helping to understand the overall feature contributions to the detected anomaly, while LIME focuses on per-instance or flow-specific analysis, providing a more granular, detailed explanation for why this particular multicast flow was flagged as malicious.
- *Real-Time Monitoring and Traffic Management:* The explainability engine provides local and global explanations to the SDN controller, helping network operators and automated systems understand why specific traffic was flagged as a threat. Based on these insights, the controller refines flow rules, blocks malicious sources, or reconfigures multicast routes to mitigate the identified threats effectively. The controller dynamically applies updated QoS policies to the data plane, such as rerouting traffic, dropping malicious flows, or throttling suspect sources.

The framework integrates the CNN-LSTM model with the SDN controller to achieve low-latency, real-time threat detection essential for seamless multicast audio/video streaming. While the CNN detects local anomalies using multicast-specific metadata, the LSTM captures long-term temporal trends, enabling proactive and efficient threat mitigation.

3. Decision-Making with Explainability Engine (LIME & SHAP)

Both LIME and SHAP are crucial components of the proposed framework, offering complementary insights into the model's decision-making process. LIME, a local interpretability technique, provides granular explanations for individual predictions. It highlights the specific features that influenced a particular classification, such as identifying a sudden spike in packet rate and frequent IGMP membership changes as indicators of a potential DDoS attack.
On the other hand, SHAP, a global interpretability method, calculates feature importance scores for all features across the entire dataset [10]. For example, SHAP findings can show that packet rate is the most important feature in detecting malicious traffic, while IGMP membership changes have a smaller impact. This global perspective reveals long-term trends and patterns. By combining these two methods, the framework gains a deeper understanding of the model's behavior, enabling both detailed, real-time threat detection and a broader, system-wide analysis of feature importance.
Once a packet is flagged as suspicious by the CNN-LSTM model, the Explainability Engine is activated to generate insights. The classified malicious data instance is passed to LIME, which perturbs the input data by modifying features such as packet size or traffic rate to simulate different scenarios. These perturbed instances are then processed by the CNN-LSTM model, and their predictions are collected to evaluate how the model responds to changes.
Next, LIME trains a surrogate model (linear regression or decision tree) on the perturbed data and corresponding CNN-LSTM predictions. The surrogate model approximates CNN-LSTM's decision-making for the specific instance, allowing LIME to identify the most influential features. These are presented as a ranked list or visualized using a figure. For example, if an instance is classified as malicious, it is passed to the explainability engine for interpretability. LIME generates perturbed data samples by modifying key features such as latency or TTL, helping to demonstrate how each feature (packet count, TTL, latency) influences the likelihood of the instance being malicious. For high prediction values, we infer that these features contribute to classifying instances as malicious.

SHAP provides granular insights into the CNN-LSTM model's decisions (data predictions) for classifying network traffic as malicious or benign. The generated Shapley values quantify the impact of each feature—packet count, TTL, and latency—on the model's predictions by analyzing all possible feature combinations. Among these, packet count is the most important feature, driving the model toward malicious classifications as it increases. TTL moderately supports malicious classification. Higher latency, observed in instances, further contributes to flagging traffic as mali-

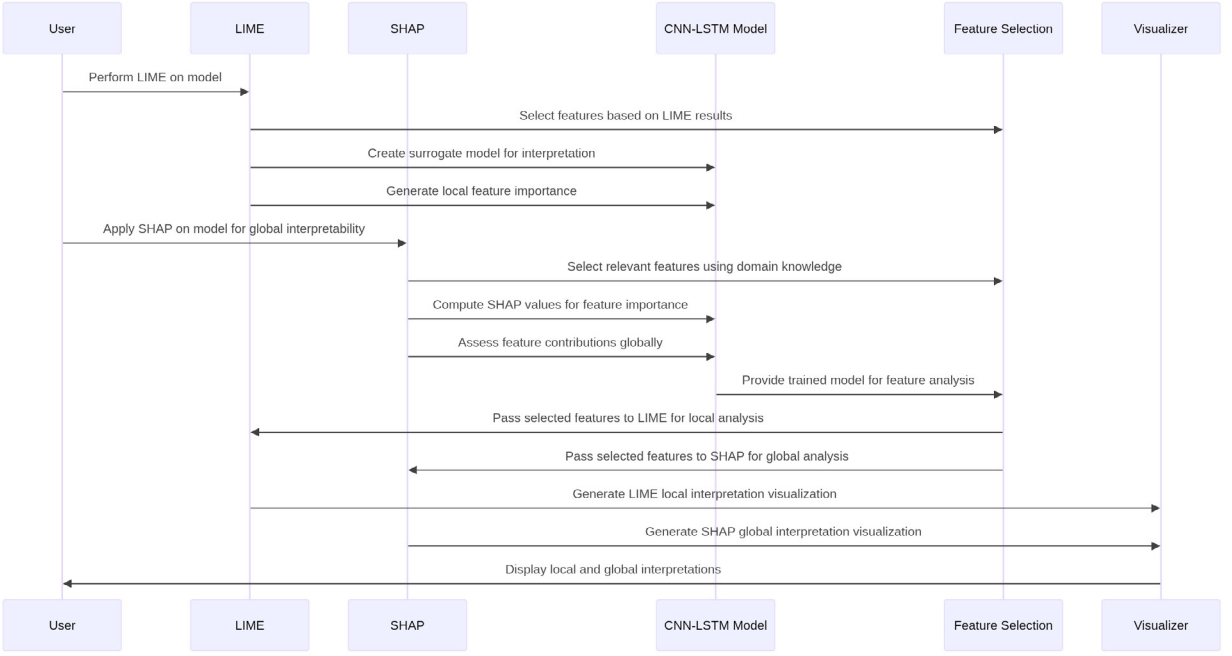

Fig. 2: LIME SHAP: Processing for threat identification Explanation

cious. SHAP *force plots* visually break down individual predictions, showing how features like packet count and TTL influence the model's output. *Summary plots* aggregate this information across multiple instances, ranking features by their overall impact.

The insights from the explainability engine are relayed to the SDN controller, enabling administrators to understand why the traffic was flagged as malicious and take corrective actions like rerouting or isolating the traffic.

The sequence diagram (Fig. 2) depicts the data flow and interactions within the framework integrating LIME and SHAP with the CNN-LSTM model for multicast traffic threat detection. Initially, the *network administrator* initiates the explainability process by requesting either local or global explanations. LIME, upon receiving the request, generates a *surrogate model* of the CNN-LSTM model and computes *local feature importance* to identify key features influencing individual predictions. Subsequently, LIME collaborates with the *FeatureSelector* to select relevant features based on its analysis. Concurrently, the network administrator applies SHAP for *global interpretability*, where SHAP calculates *Shapley values* to assess *global feature importance* across all predictions. SHAP interacts with the *FeatureSelector* to select features using domain knowledge and computes feature contributions for the entire dataset. The CNN-LSTM model, providing the trained model, collaborates with the FeatureSelector, which in turn passes the selected features to both LIME and SHAP for further analysis. Both tools then send their results to the *Visualizer*, which generates visualizations of the local and global feature importance. Finally, the administrator receives the visualized interpretations, aiding their understanding of both individual predictions and the overall feature importance. This process ensures that both local and global features importance is clearly communicated, offering transparency into the model's behaviour for network threat detection in the AI-enabled SDN framework.

The performance of the framework can be evaluated using the metrics of accuracy, explainability, and computational cost. *Accuracy metrics* are crucial in assessing the model's performance in detecting multicast anomalies. High TPR and Precision indicate effective detection, while a low FPR ensures minimal false alarms. *Explainability metrics* focus on interpretability which measures the ease of understanding explanations(e.g., "Flow size > X(threshold value) → Malicious"), fidelity (accuracy of the explanation), and sparsity (conciseness, focusing on critical features), ensuring transparency in the model's decision-making. *Computational cost metrics* assess the model's efficiency in real-time SDN environments, with Inference Time measuring response speed, Memory Usage tracking resource consumption, and Computational Complexity evaluating the operations required. These metrics ensure that the model is accurate, interpretable, and efficient in high-volume, time-sensitive multicast traffic. Integrating Explainable AI into

SDN-enabled multicast systems presents both adoption and technical challenges. From an adoption perspective, companies must invest in upgrading infrastructure to support the increased computational demands of integrating complex XAI techniques. The dynamic group membership in multicast requires real-time updates to multicast trees and SDN controllers, complicating traffic management and scalability. Technically, SHAP's high computational complexity and LIME's localized focus limit real-time decision-making and global explanation effectiveness in multicast traffic security. These factors make it challenging to deploy XAI-based solutions at scale, particularly in environments with large multicast groups and rapidly changing traffic. Future research should focus on simulating the AI-based SDN multicast framework in Mininet with OpenDaylight, using SDN datasets for intrusion detection using CNN-LSTM. Testing with redundant network setups and real-time multicast traffic can assess scalability, performance, and adaptability. Additionally, incorporating multicast orchestration will provide insights into optimizing traffic management and security in real-world SDN multicast scenarios.

4. Conclusion

AI-driven SDN has improved multicast network management through centralized control and flexibility, making it suitable for high-demand, low-latency scenarios. However, this centralization can lead to risks like DDoS attacks and data breaches, emphasizing the need for secure multicast transmission. Machine learning-based intrusion detection systems struggle in multicast environments with multiple receivers, often missing critical factors such as dynamic membership and synchronized delivery. In this regard, this paper proposed a novel framework that combines CNN-LSTM with XAI tools specifically designed for SDN-enabled multicast systems. By integrating a hybrid deep learning architecture—CNN for spatial feature extraction and LSTM for temporal pattern recognition—the framework enhances the detection of complex, time-varying patterns in multicast traffic. Incorporating LIME and SHAP generates surrogate models to explain flagged traffic flows, enabling real-time threat detection with transparency, trust, and clarity in decision-making.The proposed framework enhances transparency in AI-SDN solutions, providing network administrators with clear insights into network threats for effective and understandable security measures. Finally, the proposed conceptual framework can deliver a comprehensive, transparent, and efficient solution for real-time threat detection and mitigation in SDN, specifically within multicast environments. This research lays the groundwork for future advancements in AI-based network security, where high performance and explainability are crucial for maintaining secure and reliable network infrastructures.

References

[1] Ahmed Hazim Alhilali and Ahmadreza Montazerolghaem. "Artificial intelligence based load balancing in SDN: A comprehensive survey". In: *Internet of Things* 22 (2023), p. 100814.

[2] Nirvikar Katiyar et al. "AI and Cyber-Security: Enhancing threat detection and response with machine learning." In: *Educational Administration: Theory and Practice* 30.4 (2024), pp. 6273–6282.

[3] Mahmoud Said Elsayed et al. "A hybrid CNN-LSTM based approach for anomaly detection systems in SDNs". In: *Proceedings of the 16th International Conference on Availability, Reliability and Security, Vienna, Austria*. 2021, pp. 17–20.

[4] Naveed Ahmed et al. "Network threat detection using machine/deep learning in sdn-based platforms: a comprehensive analysis of state-of-the-art solutions, discussion, challenges, and future research direction". In: *Sensors* 22.20 (2022), p. 7896.

[5] Basil AsSadhan, Abdulmuneem Bashaiwth, and Hamad Binsalleeh. "Enhancing Explanation of LSTM-Based DDoS Attack Classification Using SHAP With Pattern Dependency". In: *IEEE Access* (2024).

[6] Osvaldo Arreche, Tanish Guntur, and Mustafa Abdallah. "XAI-IDS: Toward Proposing an Explainable Artificial Intelligence Framework for Enhancing Network Intrusion Detection Systems". In: *Applied Sciences* 14.10 (2024), p. 4170.

[7] Abdulmuneem Bashaiwth, Hamad Binsalleeh, and Basil AsSadhan. "An explanation of the LSTM model used for DDoS attacks classification". In: *Applied Sciences* 13.15 (2023), p. 8820.

[8] Josh Loveless, Arvind Durai, and Ray Blair. "IP Multicast, Volume 1: Cisco IP Multicast Networking". In: vol. 1. Cisco Press, 2016. Chap. 4.

[9] J. Aweya. *IP Multicast Routing Protocols: Concepts and Designs*. CRC Press, 2024. Chap. 5-6.

[10] K. Ho, S. Tan, and J. Lee. "Enhancing Explainability of Machine Learning-based Intrusion Detection Systems". In: *Journal of Cybersecurity and Privacy* 4.1 (2022), pp. 45–58.