EXPLAINABLE AI FOR INTRUSION DETECTION SYSTEMS:LIME AND SHAP APPLICABILITY ON MULTI- LAYER PERCEPTRON

[1]KADALI VYSHNAVI, [2]K.RAJA RAJESWARI

[1]Students, Department of MCA, B V Raju College, Bhimavaram Ap

[2]Assistant Professor, Department of MCA, B V Raju College, Bhimavaram Ap

ABSTRACT

Intrusion Detection Systems (IDS) play a critical role in identifying malicious activities in network environments. While machine learning and deep learning models such as Multi-Layer Perceptron (MLP) provide high prediction accuracy, they often act as black-box models, making it difficult to understand how predictions are made. This project proposes an Explainable Artificial Intelligence (XAI)-based IDS framework that integrates LIME (Local Interpretable Model-agnostic Explanations) and SHAP (SHapley Additive exPlanations) to interpret model decisions. The system is trained using intrusion detection datasets such as CIC IDS, and multiple algorithms including MLP, LSTM, TCN, XGBoost, and Voting Classifier are implemented to improve prediction accuracy. Among these, the Voting Classifier achieves the highest accuracy. LIME is used to provide local explanations for individual predictions, while SHAP provides both global and local feature importance insights. Experimental results demonstrate that the proposed system not only achieves high accuracy but also enhances transparency and interpretability. This approach helps in identifying important features contributing to intrusion detection, improving trust and reliability in cybersecurity systems.

Keywords: Explainable AI, Intrusion Detection System, LIME, SHAP, MLP, LSTM, XGBoost, Cyber Security, Feature Importance

I.INTRODUCTION

With the rapid growth of network-based applications, cybersecurity threats such as unauthorized access, malware, and denial-of-service attacks have increased significantly. Intrusion Detection Systems (IDS) are essential for identifying and preventing such attacks. Traditional IDS techniques rely on signature-based detection, which is limited in detecting new and unknown threats. Machine learning and deep learning models have been introduced to overcome these limitations by learning patterns from historical data and predicting potential attacks. However, these models often operate as black boxes, making it difficult to interpret their decisions.

Explainable Artificial Intelligence (XAI) has emerged as a solution to address the lack of transparency in machine learning models. Techniques such as LIME and SHAP provide insights into how models make predictions by identifying the contribution of individual features. LIME focuses on local explanations by approximating the model behavior around a specific instance, while SHAP uses game theory to provide both local and global explanations. These techniques help developers and security analysts understand the reasoning behind predictions, improving trust in the system.

This project proposes an XAI-based intrusion detection system that integrates deep learning models with explanation techniques. The system is trained using datasets such as CIC IDS and evaluated using metrics such as accuracy, precision, recall, and F1-score. Additionally, advanced models such as LSTM, TCN, XGBoost, and Voting Classifier are implemented to enhance prediction accuracy. The results demonstrate that combining high-performance models with explainable AI techniques provides both accurate and interpretable intrusion detection, making it suitable for real-world cybersecurity applications.

II SURVEY OF RESEARCH

[1] The study by Marco Tulio Ribeiro et al. (2016) introduced LIME (Local Interpretable Model-agnostic Explanations), a technique for explaining predictions of black-box models. The methodology involves generating perturbed samples around a specific instance and training a simple interpretable model to approximate the complex model locally. Results showed that LIME effectively explains individual predictions and improves model interpretability. However, it provides only local explanations and may vary with different perturbations. In the proposed system, LIME is used to explain specific intrusion detection predictions.

[2] The research by Scott Lundberg and Su-In Lee (2017) introduced SHAP (SHapley Additive exPlanations), a unified approach for interpreting machine learning models. The methodology is based on cooperative game theory, assigning importance values to each feature based on its contribution to the prediction. Results demonstrated that SHAP provides consistent and accurate feature importance explanations. However, it can be computationally expensive. In the proposed system, SHAP is used to provide both global and local explanations for IDS predictions.

[3] The study by Geoffrey Hinton et al. (1986) introduced neural network-based learning, forming the foundation for Multi-Layer Perceptron (MLP). The methodology involves multiple hidden layers to learn complex patterns in data. Results showed improved

performance in classification tasks. However, neural networks lack interpretability. In the proposed system, MLP is used as the base model for intrusion detection.

[4] The research by Sepp Hochreiter and Jürgen Schmidhuber (1997) introduced Long Short-Term Memory (LSTM), a deep learning model capable of capturing temporal dependencies. The methodology uses memory cells and gating mechanisms to retain long-term information. Results demonstrated high performance in sequence prediction tasks. However, training complexity is high. In the proposed system, LSTM is used to improve IDS accuracy.

[5] The study by Tianqi Chen and Carlos Guestrin (2016) introduced XGBoost, a powerful ensemble learning algorithm. The methodology combines multiple decision trees using gradient boosting to improve prediction accuracy. Results showed that XGBoost achieves high accuracy in classification tasks. However, it requires careful tuning. In the proposed system, XGBoost is used as an extension model to enhance IDS performance.

[6] The research by Leo Breiman (2001) introduced Random Forest, an ensemble learning method that improves prediction accuracy by combining multiple decision trees. The methodology reduces overfitting and enhances generalization. Results demonstrated high performance in classification tasks. However, interpretability remains limited. This research supports the use of ensemble methods in IDS systems and is extended in the proposed work through voting classifiers.

III. WORKING METHODOLOGY

The proposed Explainable AI-based Intrusion Detection System (IDS) follows a structured pipeline consisting of data preprocessing, model training, and explanation generation. Initially, the intrusion detection dataset (CIC IDS dataset) is loaded into the system. The dataset contains both normal and attack traffic records with various network features such as source/destination ports, packet size, and protocol details. Since the dataset includes non-numeric values and missing entries, preprocessing steps such as label encoding, normalization, and handling missing values are applied. The processed dataset is then divided into training and testing sets, where 80% of the data is used for training and 20% for testing. This ensures that the model is evaluated on unseen data for reliable performance.

In the next phase, multiple machine learning and deep learning models are trained to detect network intrusions. The base model, Multi-Layer Perceptron (MLP), is implemented as proposed in the original work. To enhance prediction accuracy, additional models such as Long Short-Term Memory (LSTM) and Temporal Convolutional Network (TCN) are trained. Furthermore, ensemble techniques

such as XGBoost and Voting Classifier are implemented as extension models. Each model is evaluated using performance metrics such as accuracy, precision, recall, and F1-score. Among all models, the Voting Classifier achieves the highest accuracy, followed by XGBoost and LSTM, demonstrating the effectiveness of ensemble and deep learning approaches in IDS.

Finally, Explainable AI techniques are applied to interpret model predictions. LIME is used to provide local explanations by perturbing input features and analyzing their impact on predictions. It explains individual predictions by highlighting which features contributed most to a specific classification (attack or normal). SHAP is used to provide both local and global explanations by calculating feature importance based on game theory. SHAP summary plots and violin plots are used to visualize feature contributions across the dataset. These explanations help in identifying the most influential features and understanding the model's behavior. The system is implemented using Python, Jupyter Notebook, and libraries such as Scikit-learn, TensorFlow, SHAP, and LIME, providing a transparent, accurate, and interpretable IDS solution.

IV RESULTS EXPLANATIONS

All machine or deep learning algorithms get trained on past dataset and then perform prediction on test data. Algorithm performance is evaluated based on predicted accuracy without performing any black box testing. Black box technique helps in knowing how algorithm predicted particular class and the predicted class label is True Positive or False Positive. Black box technique will also in explaining what features in dataset contribute most for particular class label prediction. Black box techniques help developers in knowing best features and then they can trained models with best features to get better accuracy.

In propose paper author utilizing deep learning MLP algorithm for training IDS dataset and then employing two different explainable algorithms such as SHAP and LIME for Black box testing.

LIME provides local explanations by creating a simplified, interpretable model around a specific instance, while SHAP uses game theory to attribute feature contributions to predictions, offering both local and global explanations.

LIME is best suited for explaining individual predictions and is computationally less expensive. LIME generates a set of perturbed samples around the instance you want to explain. This involves slightly modifying the input features of the original instance. These perturbed samples are then fed into the complex, "black-box" model you want to explain, and their predictions are recorded. LIME will internally perturbed different features values and then call SYSTEM function to record how False Positive and True positive

prediction percentage will change based on data perturbed.

SHAP provides both local and global explanations, offering a more comprehensive understanding of the model's behaviour and is based on a more rigorous mathematical foundation.

In short both algorithms will explain how particular model utilize training features to predict particular class labels.

In propose paper author has used ADFA-LD dataset for MLP training and for explanation but this dataset not available on internet so we have CIC dataset to train IDS dataset and then perform explanation with SHAP and LIME.

Algorithms Implementation

In propose work author has used only MLP deep learning algorithm and to further enhance prediction accuracy we have experimented with multiple algorithms such as LSTM (long short term memory), TCN (temporal convolution neural network and MLP (multilayer perceptron). Each algorithm performance is evaluated in terms of accuracy, precision, recall and FSCORE. Among all propose algorithms LSTM got high accuracy.

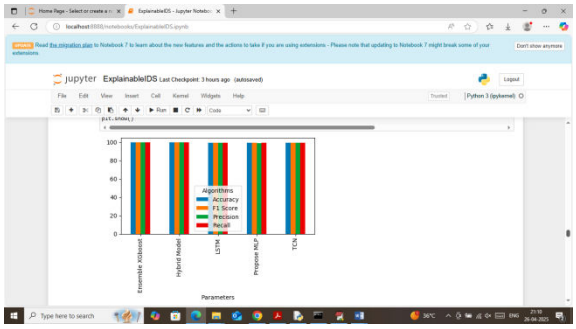
Extension Algorithms

In propose work author has concentrate only on model explanation but has not done any experiment with different algorithms which can enhance prediction accuracy and this enhance model can help explanation algorithms to decode or explain predicted class with best features.

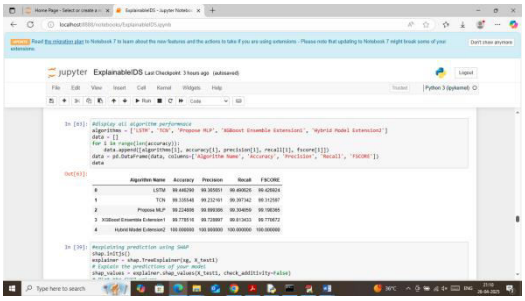
To enhance prediction accuracy we have experimented with many algorithms but accuracy got increased with Ensemble XGBOOST and Hybrid algorithm by combining multiple algorithms using Voting classifier.

Extension 1: XGBoost (eXtreme Gradient Boosting) is a powerful, open-source machine learning algorithm that utilizes gradient boosting on decision trees. It's known for its high performance, efficiency, and ability to handle large datasets. XGBoost is an ensemble learning method, combining the predictions of multiple weak learners (usually decision trees) to create a strong predictive model. Multiple decision tree help XGBOOST in enhancing accuracy.

Extension 2: The voting classifier is an ensemble learning method that combines several base models to produce the final optimum solution. The base model can independently use different algorithms such as KNN, Random forests, Regression, etc., to predict individual outputs. Combining multiple algorithms can help in finding model with best accuracy without individually experimenting with different algorithms.



In above graph x-axis represents algorithm names and y-axis represents metric values like accuracy, precision, recall in different bar colour and in all algorithms Hybrid extension2 got high accuracy



In above screen displaying all algorithms performance in tabular format

V. CONCLUSION

The proposed Explainable AI-based Intrusion Detection System successfully combines high-performance machine learning models with interpretability techniques to enhance both accuracy and transparency. By implementing models such as MLP, LSTM, TCN, XGBoost, and Voting Classifier, the system achieves excellent prediction performance, with the Voting Classifier providing the highest accuracy. The integration of explainable AI techniques such as LIME and SHAP enables detailed understanding of model decisions by highlighting the contribution of individual features. LIME provides local explanations for specific predictions, while SHAP offers both global and local insights into feature importance. This combination helps security analysts understand the reasoning behind intrusion detection, improving trust and reliability. Overall, the system provides a scalable, accurate, and interpretable solution for cybersecurity applications, making it suitable for real-world intrusion detection scenarios.

RE.FERENCES

[1] M. T. Ribeiro, S. Singh, and C. Guestrin, "Why Should I Trust You? Explaining the Predictions of Any Classifier," *Proc. ACM SIGKDD*, 2016.

[2] S. M. Lundberg and S.-I. Lee, "A Unified Approach to Interpreting Model Predictions," *Advances in Neural Information Processing Systems*, 2017.

[3] G. Hinton, D. Rumelhart, and R. Williams, "Learning Internal Representations by Error Propagation," *Nature*, 1986.

[4] S. Hochreiter and J. Schmidhuber, "Long Short-Term Memory," *Neural Computation*, 1997.

[5] T. Chen and C. Guestrin, "XGBoost: A Scalable Tree Boosting System," *Proc. ACM SIGKDD*, 2016.

[6] L. Breiman, "Random Forests," *Machine Learning*, 2001.

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

[8] C. M. Bishop, *Pattern Recognition and Machine Learning*. Springer, 2006.

[9] K. P. Murphy, *Machine Learning: A Probabilistic Perspective*. MIT Press, 2012.

[10] F. Chollet, *Deep Learning with Python*. Manning Publications, 2017.

[11] A. Géron, *Hands-On Machine Learning with Scikit-Learn and TensorFlow*. O'Reilly Media, 2017.

[12] S. Raschka and V. Mirjalili, *Python Machine Learning*. Packt Publishing, 2017.

[13] J. Brownlee, *Machine Learning Mastery with Python*. 2016.

[14] D. Jurafsky and J. Martin, *Speech and Language Processing*. Pearson, 2009.

[15] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning*. Springer, 2009.

[16] J. Han, M. Kamber, and J. Pei, *Data Mining: Concepts and Techniques*. Morgan Kaufmann, 2011.

[17] R. Kohavi, "A Study of Cross-Validation and Bootstrap for Accuracy Estimation," *IJCAI*, 1995.

[18] L. Rokach, "Ensemble-Based Classifiers," *Artificial Intelligence Review*, 2010.

[19] H. Liu and H. Motoda, *Feature Selection for Knowledge Discovery*. Springer, 1998.

[20] M. Abadi et al., "TensorFlow: Large-Scale Machine Learning," 2016.

[21] Y. LeCun, Y. Bengio, and G. Hinton, "Deep Learning," *Nature*, 2015.

[22] K. He et al., "Deep Residual Learning for Image Recognition," *CVPR*, 2016.

[23] O. Ronneberger et al., "U-Net: Biomedical Image Segmentation," *MICCAI*, 2015.

[24] A. Krizhevsky et al., "ImageNet Classification with Deep CNNs," *NIPS*, 2012.

[25] H. Greenspan et al., "Deep Learning in Medical Imaging," *IEEE TMI*, 2016.