Effective Approaches for Intrusion Detection Systems in the Face of Low-Frequency Attacks

Chadia El Asry [1],*, Ibtissam Benchaji [1], Samira Douzi [2], and Bouabid El Ouahidi [1]

[1] Intelligent Processing and Security of Systems (IPSS), Faculty of Sciences, Mohammed V University, Rabat, Morocco
[2] Faculty of Medicine and pharmacy FMPR, Mohammed V University in Rabat, Morocco
Email: chadia.elasry@um5r.ac.ma (C.E.A.); ibtissam.benchajy@um5s.net (I.B.); s.douzi@um5r.ac.ma (S.D.); b.elouahidi@um5r.ac.ma (B.E.O.)
*Corresponding author

Abstract—**This paper presents a new approach to improve the detection of network security by combining feature selection with Long-Short-Term-Memory (LSTM) approaches. The SHapley Additive exPlanations (SHAP) values approach is utilized for feature selection, in conjunction with cross-validation, to identify the most effective set of features that improve model recall for each specific sort of assault. We employ the Network Security Laboratory-Knowledge Discovery in Databases (NSL-KDD) dataset to train and assess the efficacy of our model. The suggested model exhibits greater performance in comparison to standard LSTM models when utilizing all features. Furthermore, it surpasses current leading models with an accuracy of 99.74%, precision of 95.42%, recall of 94.92%, and F1-Score of 94.90%. In addition, the model demonstrates outstanding aptitude in precisely detecting Remote-to-Local (R2L) and User-to-Root (U2R) attacks, which are complex forms of intrusions that exploit vulnerabilities to gain unauthorized access to systems or networks. Although infrequent, these assaults provide a substantial risk because they have the ability to do substantial harm and compromise confidential data.**

Keywords—**intrusion detection systems, deep learning, SHapley Additive exPlanations (SHAP) values, Long-Short-Term-Memory (LSTM), feature selection**

I. Introduction

The swift progress of computer and communication networks has enabled the global distribution of more convenient services through Internet technology. Nevertheless, the escalating quantity and assortment of cyberattacks encompassing network infections, malevolent eavesdropping, and other harmful endeavors, present substantial hazards to the security of persons' information and property. Therefore, it is of utmost importance to prioritize the protection of information and communications for both individuals and society as a whole [1, 2]. Although firewalls are commonly used and essential security mechanisms, their dependence on manual configuration and their slow response to new attack methods make them inadequate for highly secure entities, such as government and military institutions [3]. Network security researchers suggest implementing Intrusion Detection Systems (IDSs) as a method to promptly detect and respond to abnormal network intrusions.

An Intrusion Detection System (IDS) has demonstrated its effectiveness and potential as a cybersecurity solution. It functions by identifying established dangers and malevolent actions by monitoring traffic data in computer systems [4]. Upon identification of these risks, the system generates alerts to promptly inform appropriate parties about the found security vulnerabilities.

Typically, Intrusion Detection Systems (IDSs) can be classified into three primary types [5]: those employing the behavioral approach (which aims to discover anomalies), the scenario method (which focuses on detecting signatures), and the specification approach.

Behavioral analysis comprises two distinct phases: a learning phase that enables the system to comprehend and identify typical behavior, and a detection phase devoted to uncovering anomalies. This approach exhibits remarkable efficacy in pinpointing unfamiliar attacks [6, 7]. Despite its effectiveness, behavioral analysis is not without limitations. It may produce false positives or negatives, as the definition of "normal" behavior can be intricate, and attackers can adapt to evade these systems. The precision of outcomes is significantly influenced by the quality of the training data, and certain systems might experience delayed anomaly detection.

On the other hand, scenario analysis utilizes a predetermined collection of attack scenarios, considering them as distinctive patterns and producing alerts when there are matches. However, it requires regular updates to the signatures [5].

The specification technique integrates the benefits of the behavioral approach with scenario analysis. It involves manually defining requirements, which allows for the identification of previously unidentified assaults with a minimal percentage of false positives.

The deployment of Intrusion Detection Systems (IDS) that use machine learning [8] and deep learning techniques [9] is frequently hampered by class imbalance, a common issue in this field. Class imbalance occurs when the number of intrusion instances is significantly

Manuscript received March 6, 2024; revised May 9, 2024; accepted June 3, 2024; published September 25, 2024.

doi: 10.12720/jait.15.9.1070-1078

lower than the number of routine activities, which are more prevalent in real network systems. Within the Network Security Laboratory-Knowledge Discovery in Databases (NSL-KDD), the User-to-Root (U2R) and Remote-to-Local (R2L) attacks have a substantial impact on system security and data confidentiality. These attacks are classified as minority classes, with just 995 and 52 incidences, respectively, in comparison to other types of attacks like Denial of Service (DoS) attacks or regular events. As a result of this imbalance, many current approaches prefer to give more importance to the majority class and ignore the minority classes due to their little data. Frequently, this results in models exhibiting bias towards the dominant class, even though it is crucial to reliably detect intrusions. Misclassifying typical behavior as intrusive might have more severe repercussions than failing to identify an incursion. In order to tackle these issues, we propose the implementation of a customized model designed to enhance the detection and classification of R2L and U2R attacks. Our model incorporates the utilization of SHapley Additive exPlanations (SHAP) values alongside cross-validation to carefully choose features that are tailored to each form of attack. In addition, we improve the classification abilities of our model by implementing Long-Short-Term-Memory (LSTM) networks. We also utilize cross-validation throughout both the training and testing stages to optimize performance.

The document is structured into various sections, commencing with an introduction. Section II provides an overview of prior research, while Section III discusses the methodology employed in this paper. Section IV comprises the proposed approach and the accompanying empirical data. Section V functions as the culmination of the study, offering a succinct overview of the primary findings and insights.

II. Related Work

Many academics strongly recommend integrating intrusion detection and Machine Learning (ML) technologies to detect network threats by creating efficient models.

Amaizu *et al.* [10] leverage Principal Component Analysis (PCA) to extract features and employ multiple deep learning classification models. By comparing the performances of different models, it was continuously found that the Deep Neural Network (DNN) achieved the maximum accuracy across all datasets utilized in the inquiry, including the Network Security Laboratory-Knowledge Discovery in Databases (NSL-KDD), University of New South Wales-Network Based 2015 (UNSW-NB15), and Canadian Institute for Cybersecurity-Intrusion Detection Systems 2018 (CSE-CIC-IDS2018) datasets.

Imrana *et al.* [11] introduced a bidirectional LSTM deep learning method for identifying different sorts of attacks, with a specific focus on U2R and R2L attacks. The suggested model outperforms the traditional LSTM in terms of accurately detecting these attack types.

Le *et al.* [12] presented a classifier for Intrusion Detection Systems (IDS) that utilizes Recurrent Neural Networks (RNN). They conducted an investigation using six different optimization algorithms for LSTM-RNN (Long Short-Term Memory-Recurrent Neural Network). Among these, Nadam demonstrated the highest level of effectiveness in identifying threats. The proposed approach exhibited enhanced capabilities in identifying each assault in comparison to LSTM-RNN with the Stochastic Gradient Descent (SGD) optimizer. Nevertheless, it is worth mentioning that although there has been progress, the performance indicators are still regarded as mediocre.

Laghrissi *et al.* [13] presented a novel Intrusion Detection System (IDS) that utilizes Long Short-Term Memory (LSTM) and an attention mechanism. They used this system to the NSL-KDD dataset, which consists of five different attack types. Although the model has generally great performance, it frequently misclassifies U2R attacks as normal. Laghrissi *et al.* [14] introduce an Intrusion Detection System (IDS) that utilizes Long Short-Term Memory (LSTM). Principal Component Analysis (PCA) and Mutual Information (MI) are used as methods to reduce the number of dimensions and pick relevant features. The model is tested on the Knowledge Discovery in Databases 1999 (KDD99) benchmark dataset, and the findings show that PCA-based models attain the maximum accuracy for both training and testing. This applies to both binary and multiclass classification, with accuracies of 99.44% and 99.39% respectively. In addition, the authors include R2L and U2R attacks together in the same categories because of the limited number of incidents connected with these attacks in comparison to others.

Dong *et al.* [15] presented Multi-Channel Attention-Long Short-Term Memory (MCA-LSTM), an Intrusion Detection System (IDS) that utilizes Multivariate Correlation Analysis (MCA) and Long Short-Term Memory (LSTM) as part of its Machine Learning approach. The model utilized the Information Gain (IG) technique for selecting features. The process involved the selection of a specific collection of characteristics, which were then converted into a matrix representing the triangle areas. This matrix, known as the Triangle Area Map (TAM), was subsequently utilized in the LSTM algorithm to predict intrusions. The authors evaluated the model's performance by utilizing the NSL-KDD and UNSW-NB15 datasets. The experimental results demonstrated that MCA-LSTM attained a test accuracy of 82.15% for 5-way classification using the NSL-KDD dataset. For the 10-way classification job, MCA-LSTM achieved a test accuracy of 77.74% in the case of UNSW-NB15. MCA-LSTM achieved an accuracy of 80.52% for binary classification using the NSL-KDD dataset and 88.11% using the UNSW-NB15 dataset. Despite the superior performance of these findings compared to earlier methods, the authors did not investigate the impact of dataset size. Furthermore, they neglected to consider a diverse range of performance measurements such as recall and F1-Score. Fu *et al.* [16] introduce a model for detecting abnormal traffic patterns in their paper, titled "A Deep Learning Model for Network Intrusion Detection" (DLNID). This model integrates an attention mechanism with the Bidirectional Long Short-Term Memory (Bi-LSTM) network. The authors propose the use of an Adaptive Synthetic Sampling (ADASYN) oversampling algorithm as a data augmentation technique to solve the issues of data imbalance and low detection accuracy in network intrusion data. In addition, they employ a stacked autoencoder with an augmented dropout structure as a technique for reducing data size, hence improving the model's capacity to generalize. The network structure is enhanced by integrating the channel attention mechanism with the bidirectional LSTM network. The network model provided attains a precision of 90.73% and an F1-Score of 89.65% on the KDD Test+ test set. Nevertheless, even with the use of data augmentation, it is seen that the U2R category had a higher probability of being misclassified.

Mohammad *et al.* [17] utilized a traditional neural network to classify network hazards within a system. A two-layer multi-layer perceptron was built using the backpropagation learning approach. The proposed technique attained a classification accuracy rating of 90.78%. The work employed the KDDCUP99, ISCX3 (ISCX FlowMeter Traffic dataset version 3), and NSL-KDD4 datasets for both training and testing the model. Nevertheless, it is important to mention that the dataset used is obsolete and does not adequately represent current attack scenarios.

Opoola *et al.* [18] devised a hybrid methodology called Layer-wise Aggregation and Embedding-Bidirectional Long Short-Term Memory (LAE-BLSTM) for the identification of botnets. The model was trained using the BotIoT6 dataset and successfully shown the capability to differentiate between attacks and normal traffic [18]. The study's evaluation results revealed a remarkable accuracy rate of 91.89%, even after reducing the bulk of the data.

Table I presents a comprehensive summary of different well-established models used for Intrusion Detection Systems (IDS).

Table I shows the intricacies associated with identifying Remote-to-Local (R2L) and User-to-Root (U2R) attacks. Many models struggle to differentiate between these two sorts of attacks, frequently misclassifying them as regular behaviors. Occasionally, the R2L and U2R categories are merged because they have relatively few examples compared to other groups. In order to enhance the detection of these specific classes, various techniques may exclude other attack types or employ methods to create more instances. This study presents a strategic approach specifically developed to address the aforementioned challenges. Through the implementation of several methodologies, we want to improve the detection capabilities for R2L and U2R attacks, enabling us to more effectively distinguish and precisely classify these types of attacks.

TABLE I. PAPERS SUMMARY OF THE PAST WORK MENTIONED ABOVE, IN ADDITION TO THEIR RESPECTIVE LIMITATIONS

Ref.	Architecture	Dataset	Limitation
[9]	LSTM	NSL-KDD	The model provides poor performance for Remote-to-Local (R2L) and User-to-Root (U2R) attacks
[13]	LSTM-Attention mechanism	NSL-KDD	The model frequently faces difficulties in accurately classifying U2R attacks, resulting in frequent misclassifications as normal instances.
[14]	Principal Component Analysis-Long Short Memory (PCA-LSTM)	KDD99	Due to the limited number of instances associated with both R2L and U2R attacks compared to other categories, the authors combine these two attack types into the same classes.
[15]	Multi-scale Attention (MCA-LSTM)	NSL-KDD and UNSW-NB15	The strategies employed to prevent overfitting during the training process were not explicitly emphasized by the authors.
[19]	Artificial Neural Network (ANN)		High time execution
[20]	K-Nearest Neighbors (KNN), Feature Selection, Unsupervised model	Bot-IoT	Poor performance
[21]	Malicious Activity Detection using Random Forests (MAD-RF)	NSL-KDD	The detection accuracy for ICMP and UDP DDoS attacks is less and can be improved
[22]	Naïve Bayes feature embedding-Support Vector Machine (SVM)	NSL-KDD, UNSWNB15, CIC-IDS2017 (Canadian Institute for Cybersecurity), Kyoto 2006+	The model does not yield better performance for all the data sets except NSL-KDD.
[23]	KNN and Recurrent Selection Algorithm (RSL)	Industrial Control System Cyber attack Dataset	Accuracy can be improved.
[24]	Correlation-Fisher Linear Discriminant Analysis (FLDA)	KDD99	They only relied on accuracy as the sole performance evaluation metric
[25]	Principal Component Analysis (PCA)-Naïve Bayes	NSL-KDD	To enhance the detection of U2R and R2L attack classes, they excluded other types of attacks.

III. BACKGROUND

Within this part, we will initially explore the fundamental principle underlying the LSTM architecture. The structure of SHAP values is thereafter established. We provide a more comprehensive description of the NSL-KDD dataset utilized for the training and validation of our model.

A. Long Short-Term Memory (LSTM)

Long Short-Term Memory (LSTM) is a specific form of Recurrent Neural Network (RNN) architecture that was proposed by Hochreiter and Schmidhuber in 1995. It is commonly used in deep learning to represent time series data [26]. LSTM, in contrast to conventional feed-forward neural networks, integrates feedback connections among hidden units that are interconnected at predefined time intervals. This feature enables the model to acquire knowledge and make predictions about long-term relationships within a sequence by analyzing the patterns in the prior data [27].

LSTMs were specifically developed to address the difficulties encountered when training conventional RNNs, such as the problem of gradients vanishing or exploding during the learning process. The models are equipped with three separate gates: the input, forget, and output gates. These gates are responsible for controlling modifications to specific memory units called cell states (ct), as illustrated in Fig. 1. Given its capacity to efficiently regulate the dissemination of data, LSTMs have become a fundamental component in the domain of intrusion detection, as demonstrated by prominent research and our personal expertise. Due to their shown effectiveness in this field, we have chosen to incorporate them as a vital component of our strategy.

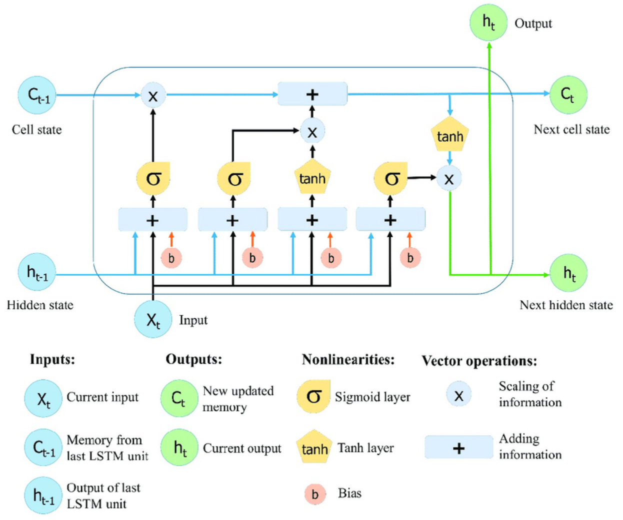

Fig. 1. The architectural configuration of the Long Short-Term Memory (LSTM) neural network [28].

B. *Feature Selection: SHapley Additive exPlanations (SHAP) Values*

The process of selecting features in an intrusion detection system is a crucial step because of the intricate nature and inherent interference of network data, particularly in specific circumstances. Hence, the utilization of feature selection techniques is crucial in mitigating these issues and subsequently diminishing the dimensionality of a dataset [25]. The SHapley Additive exPlanations (SHAP) value method, introduced in 2017 by Lundberg and Lee [29], is a technique that aims to provide an explanation for the output of machine learning models. It draws inspiration from game theory and determines the SHAP values by comparing model predictions with and without a specific feature. This is achieved through combinatorial calculation and retraining of the model on all possible combinations of attributes that involve the feature of interest [30]. The values facilitate the computation of the significance of each characteristic for every data point by taking the average of the absolute Shapley values computed for a certain dataset, so producing "overall" values for each variable.

IV. Materials and Methods

A. *NSL-KDD Dataset*

The NSL-KDD dataset is an improved version of the original KDD-99 dataset, which was developed for the International Knowledge Discovery and Data Mining Tools Competition [31]. It consists of 4,898,431 instances that were obtained from the raw data of the KDD Cup 1999. This dataset has been optimized by the removal of redundant information [32, 33]. The dataset includes a total of 42 attributes that are classified into four types: Categorical, Binary, Discrete, and Continuous, as specified in Table II. The dataset assigns a standardized identifier to each entry and encompasses around 22 distinct attack categories, including smurf, nmap, back, teardrop, neptune, Satan, ipsweep, portsweep, loadmodule, buffer_overflow, warezmaster, land, imap, rootkit, load-module, ftp_write, multihop, phf, perl, and spy. The normal activity class is the most prevalent, with 67,343 samples, while the "neptune" attack category is the most frequent, occurring 41,214 times. By comparison, the "spy" assault is exceptionally uncommon, documented just on two occasions in the dataset.

The NSL-KDD dataset is extensively employed by several academics to train and assess their suggested methodologies in the domain of intrusion detection. Consequently, we would like to integrate this dataset into our work for comparable objectives.

TABLE II. Papers Summary of the Past Work Mentioned Above, in Addition to Their Respective Limitations

Feature type	Feature Number
Categorical	2, 3, 4, 42
Binary	7, 12, 14, 20, 21, 22
Discrete	8, 9, 15, 23 to 41, 43
Continuous	1, 5, 6, 10, 11, 13, 16, 17, 18, 19

B. *Methodology Proposal And Experimental Findings*

The proposed approach depicted in Fig. 2 comprises multiple steps, commencing with Dataset preparation and culminating in Classification. The model has four essential stages: data preprocessing, data partitioning, feature selection, and classification. The first phase entails performing data cleaning to improve the quality of the data. Afterwards, the features are standardized by scaling them to a certain range, and non-numeric data is transformed into numeric data. During the second stage, the data is divided into subsets according to the four distinct assault types (DoS, Probe, U2R and R2L), with each subset having occurrences exclusively from a certain attack type. The final phase involves utilizing SHAP-values in conjunction with cross-validation, using recall as a metric, to determine the most significant characteristics. Ultimately, the chosen characteristics are utilized as input for the Long Short-Term Memory (LSTM) model throughout the classification procedure. Below, a detailed elucidation of this methodical inquiry is presented. In the following section, we will provide the results of data preprocessing and give a detailed description of the implementation specifics, as well as the evaluation metrics used in this study.

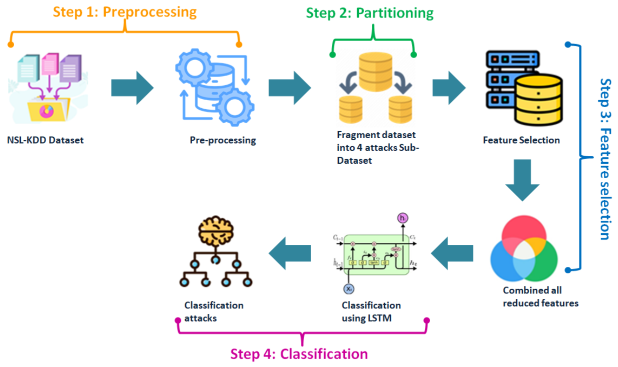

Fig. 2. Flow chart oh the proposed approach.

C. Simulation Environment

The suggested model is evaluated on a Windows 10 operating system-equipped PC (personal computer) with a 4GB GEFORCE GTX 1650 Ti graphics card, 6 GB of RAM (Random Access Memory), and an Intel Core i7–4790 processor functioning at 3.60 GHz. The model is implemented in Python 3.8. The purpose of this simulation environment is to facilitate the replication of our implementation and to guarantee that it does not necessitate a significant amount of resources. Our objective is to illustrate that our model can be effectively evaluated on a standard personal computer configuration by providing these specifications, thereby making it accessible to a broader spectrum of researchers and practitioners.

D. Datasets Preprocessing

The dataset has undergone numerous preparation methods, which are elaborated upon below:

(1) The attacks in the NSL-KDD dataset were classified into four distinct categories:

- Denial of Service (DoS) involves various assaults, including neptune, back, land, pod, smurf, teardrop, udpstorm, mail-bomb, apache2, processtable, and worm.
- The probe includes the following tools: ipsweep, nmap, portsweep, satan, mscan, and saint.
- R2L: This category includes a range of attacks, including ftp write, guess passwd, imap, multihop, phf, spy, warezcli-ent, warezmaster, sendmail, named, snmpgetattack, snmpguess, xlock, xsnoop, and httptunnel.
- U2R: The U2R attack involves the exploitation of vulnerabilities such as xterm, loadmodule, buffer overflow, perl, rootkit, ps, and sqlattack.

(2) The NSL-KDD dataset categorizes attacks into four distinct classifications, with numerical values allocated to each attack type as outlined below: 0 represents the category of "Normal," 1 represents "Denial of Service (DoS)," 2 represents "Probe," 3 represents "Remote to Local (R2L)," and 4 represents "User to Local (U2L)."

(3) The pandas. Factory function is employed to transform symbolic-valued features, such as protocol, service, and flag, into attributes with numeric values.

(4) The Standard Scaler method is utilized to standardize features using the rescaling technique.

The dataset is partitioned into four distinct subsets, with each subset only containing instances of one of the four types of attacks. The sets are named DoS-set, Probe-set, R2L-set, and U2L-set.

E. Feature Selection

The SHAP value technique with cross-validation is employed to identify the most pertinent features for each type of assault. This method aims to select a suitable subset of features that optimizes the model's recall. The use of recall as an evaluation metric to select the most optimized subset is of great importance in the field of intrusion detection. Indeed, this metric provides the detection percentage, thus making it possible to evaluate to what extent the model identifies intrusions.

Our approach utilizes SHAP values in conjunction with the XGBoost model and a cross-validation procedure to determine the optimal set of features. The main objective is to optimize recall, prioritizing the model's ability to properly identify positive instances, such as attacks. Through the use of iterations, SHAP values are utilized to evaluate different subsets of features. This procedure is executed iteratively until all features have been considered. The ultimate outcome involves demonstrating the collection of characteristics that achieve high levels of recall and sensitivity, which are crucial aspects in the classification of attacks.

This feature selection method was applied to the 4 attack subsets (DoS-set, Probe-set, R2L-set and U2R-set), in order to choose for each subset the optimal set of features that maximizes the recall and sensitivity value. Fig. 3 depicts the results obtained with SHAP values.

Table III provides a succinct summary of the unique features within each subset. Specifically, 25 features are chosen for the Dos-set, 17 features for the Probe-set, 24 features for the R2L-set, and 28 features for the U2R-set, employing the method described above.

The selected features for each subset are combined and considered relevant for NSL-KDD dataset. The combination of features yields a total of 38 important features ('src_bytes','dst_bytes','count', 'service', 'srv_count', 'protocol_type', 'dst_host_same_src_port_rate', 'dst_host_diff_srv_rate', 'dst_host_srv_count', 'logged_in', 'dst_host_same_srv_rate', 'flag', 'dst_host_count', 'same_srv_rate', 'dst_host_rerror_rate', 'dst_host_srv_diff_host_rate', 'dst_host_serror_rate', 'dst_host_srv_rerror_rate', 'num_outbound_cmds', 'land', 'srv_diff_host_rate', 'root_shell', 'dst_host_srv_serror_rate', 'num_file_creations', 'hot', 'srv_rerror_rate', 'num_access_files', 'num_root', 'wrong_fragment', 'num_compromised', 'is_host_login', ‘urgent’, ‘is_guest_login’, , ‘num_failed_logins’, ‘su_attempted’, ‘srv_serror_rate’, ‘num_shells’, ‘duration’) that will be utilized throughout the entire dataset to improve the classification of the different attacks, particularly R2L and U2R attacks.

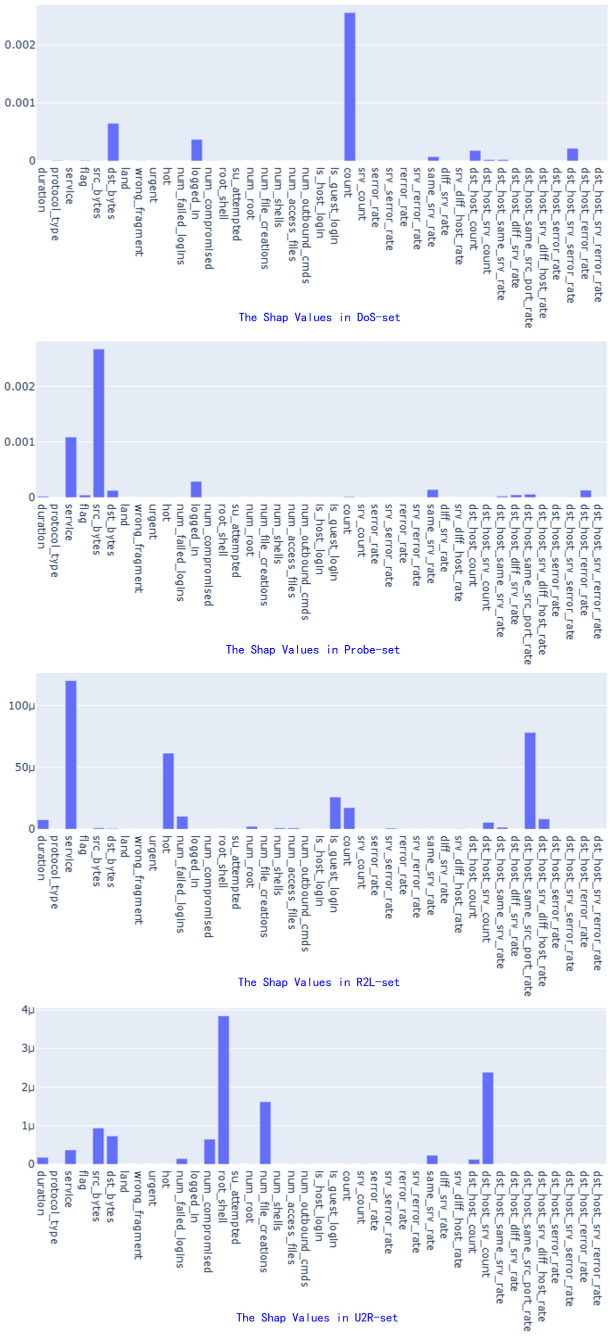

Fig. 3. SHAP scores for each feature in every subset.

The classification was done with an LSTM model using cross-validation, while evaluating the metrics of Accuracy, Recall, Precision and F1-Score.

The LSTM model is distinguished by specific parameters designed to govern its behavior during training and testing. These parameters are listed in the Table IV.

TABLE III. LIST OF SELECTED FEATURES FOR EACH SUBSET ATTACK

Subset	Selected features
DoS-set	‘count’, ‘dst_bytes’, ‘logged_in’, ‘dst_host_srv_serror_rate’, ‘dst_host_count’, ‘same_srv_rate’, ‘dst_host_same_srv_rate’, ‘dst_host_srv_count’, ‘flag’, ‘protocol_type’, ‘wrong_fragment’, ‘num_compromised’, ‘src_bytes’, ‘dst_host_srv_rerror_rate’, ‘su_attempted’, ‘root_shell’, ‘land’, ‘num_failed_logins’, ‘hot’,’urgent’, ‘num_file_creations’, ‘service’, ‘num_root’, ‘is_host_login’,’num_shells’, ‘srv_rerror_rate’, ‘dst_host_serror_rate’
Probe-set	‘src_bytes’, ‘service’, ‘logged_in’, ‘same_srv_rate’, ‘dst_host_rerror_rate’, ‘dst_bytes’, ‘dst_host_same_src_port_rate’, ‘dst_host_diff_srv_rate’, ‘flag’, ‘dst_host_same_srv_rate’, ‘duration’,’count’, ‘dst_host_srv_serror_rate’, ‘srv_count’, ‘hot’,’dst_host_count’, ‘num_compromised’, ‘num_failed_logins’
R2L-set	‘service’, ‘dst_host_same_src_port_rate’, ‘hot’, ‘is_guest_login’, ‘count’, ‘num_failed_logins’, ‘dst_host_srv_diff_host_rate’, ‘duration’,’dst_host_srv_count’, ‘num_root’, ‘dst_host_same_srv_rate’, ‘num_shells’, ‘src_bytes’, ‘num_access_files’, ‘srv_serror_rate’, ‘dst_bytes’, ‘srv_diff_host_rate’, ‘land’, ‘su_attempted’, ‘urgent’, ‘flag’, ‘protocol_type’, ‘logged_in’, ‘num_compromised’, ‘root_shell’
U2R-set	‘root_shell’, ‘dst_host_srv_count’, ‘num_file_creations’, ‘src_bytes’, ‘dst_bytes’, ‘num_compromised’, ‘service’, ‘same_srv_rate’, ‘duration’,’num_failed_logins’, ‘dst_host_count’, ‘num_root’, ‘su_attempted’,’logged_in’, ‘urgent’, ‘hot’, ‘num_access_files’, ‘wrong_fragment’,’land’, ‘flag’, ‘protocol_type’, ‘num_shells’,’dst_host_srv_rerror_rate’, ‘num_outbound_cmds’, ‘dst_host_rerror_rate’, ‘dst_host_srv_serror_rate’, ‘dst_host_serror_rate’, ‘dst_host_srv_diff_host_rate’

TABLE IV. THE LSTM MODEL PARAMETERS

Parameter	Value
Activation function	Softmax
Loss function	Sparse categorical crossentropy
Optimizer	Nadam
Learning rate	0.002
Epsilon	1e-08
Schedule decay	0.004
Epochs	10
Dropout	0.3

A. Experimental Results and Discussion

The results of LSTM using every feature of NSL-KDD, and our proposed model are presented in Tables V and VI, respectively. Furthermore, Fig. 4 demonstrates the comprehensive performance of our model and LSTM when considering all features. Accuracy, recall, precision, and F1-Score are used as performance measurements in this context. The effectiveness of our strategy has been verified by the utilization of the 5-fold cross-validation technique.

Table V presents the classification performance for each attack type using the entire collection of features. Multiple measures, including as precision, recall, and F1-Score, were assessed for each attack category.

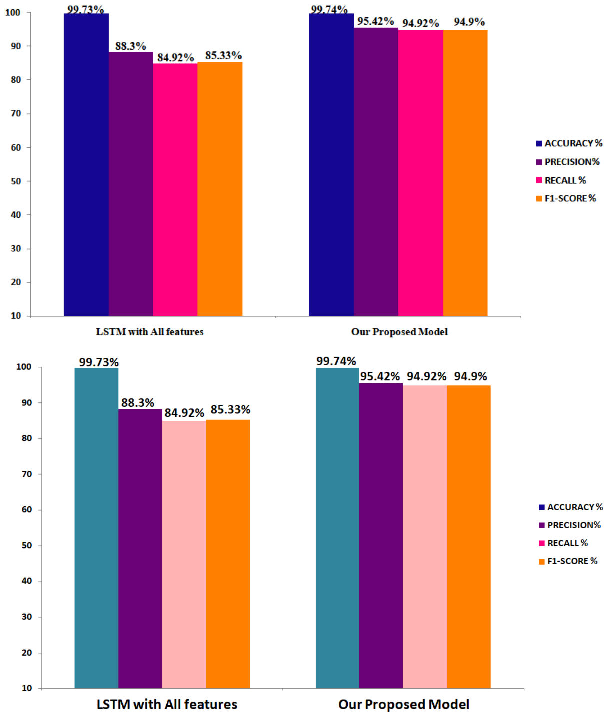

Fig. 4. Performance accuracy of proposed model and LSTM with all features.

TABLE V. CLASSIFICATION PERFORMANCE OF EACH ATTACK OF LSTM WITH ALL FEATURES

Attacks	Precision %	Recall %	F1-Score %
Normal	99.72	99.79	99.76
DoS	99.92	99.97	99.95
Probe	99.85	99.45	99.65
U2R	75.36	92.04	82.87
R2L	66.67	33.33	44.44

In the "Normal" category, LSTM exhibits outstanding classification performance, achieving a precision of 99.72%, recall of 99.79%, and an F1-Score of 99.76%. Regarding "DoS (Denial of Service)" attacks, the model demonstrates exceptional precision of 99.92%, recall of 99.97%, and an F1-Score of 99.95%, which emphasizes its efficacy in detecting DoS attacks. The "Probe" category has exceptional accuracy with a precision rate of 99.85%, however the recall rate somewhat decreases to 99.45%, resulting in an overall F1-Score of 99.65%. The model's performance in detecting "U2R (User to Root)" assaults has declined, resulting in a precision of 75.36%, recall of 92.04%, and an F1-Score of 82.87%. The LSTM model performs poorly for detecting "R2L (Remote to Local)" attacks, with a precision of 66.67%, recall of 33.33%, and an F1-Score of 44.44%. These results indicate that the LSTM model faces difficulties in accurately identifying these types of attacks.

In general, R2L and U2R demonstrate significantly worse performance metrics in comparison to other assault types.

Table VI displays the results of the proposed model. It is evident that the proposed model demonstrated superior performance compared to LSTM with all features in categorizing R2L and U2R.

TABLE VI. CLASSIFICATION PERFORMANCE OF EACH ATTACK OF OUR PROPOSED METHOD

Attacks	Precision %	Recall %	F1-Score %
Normal	99.71	99.81	99.76
DoS	99.91	99.97	99.94
Probe	99.88	99.43	99.65
U2R	77.61	92.04	84.21
R2L	100	83.33	90.91

The results of the proposed approach are presented in Table VI. The suggested model clearly exhibited higher performance in classifying R2L and U2R compared to LSTM when all features were considered. Indeed, our suggested model improves the performance marginally compared to the LSTM model with every feature in "Normal" and "DoS (Denial of Service)" assaults.

In the "U2R" category, the precision increases slightly from 75.36% to 77.61%, while the model maintains a stable recall of 92.04%. Furthermore, the F1-Score rises to 84.21%, indicating a beneficial effect on the model's capacity to precisely detect instances of U2R attacks. Our suggested model has successfully addressed and enhanced the performance in identifying U2R assaults compared to the prior LSTM model that included all features.

The proposed model's results in identifying the "R2L" category in Table VI are notably significant. The precision for "R2L" attacks is 100%, with a recall of 83.33% and an amazing F1-Score of 90.91%. This demonstrates a significant improvement in the model's capacity to reliably classify instances of R2L assaults, highlighting the efficacy of our proposed model in achieving the specific objective of enhancing the detection performance for both "U2R" and "R2L" attack types.

Fig. 4 compares the performance metrics of an LSTM model using all features with the proposed model. Although both models provide a high level of accuracy, the one proposed displays significant enhancement in precision, recall, and F1-Score. The precision experiences a notable increase from 88.3% to 95.42%, demonstrating a significant decrease in the occurrence of false positives. Simultaneously, the recall rate rises from 84.92% to 94.92%, indicating a significant decrease in the number of false negatives. The F1-Score demonstrates a significant enhancement, increasing from 85.33% to 94.9%, so emphasizing the improved equilibrium between precision and recall achieved by the suggested model. The results highlight the higher classification abilities of the proposed model in comparison to the LSTM model using all features.

Table VII presents a detailed comparison of different models' performance in identifying U2R and R2L attacks. It emphasizes the superiority of the proposed model compared to previous research in this field. The table compares the metrics of the proposed model with those of previous models such as DLNID (Distributed learning Network Intrusion Detection), LSTM, and BidLSTM. The suggested model demonstrates superior performance compared to others in classifying U2R attacks, achieving a precision of 77.61%, a recall of 92.04%, and an F1-

Score of 84.21%. This signifies a significant enhancement compared to the DLNID model, which exhibited a considerably lower recall rate of 24.00% and poor precision. The suggested model demonstrates improved accuracy and effectiveness compared to the LSTM and BidLSTM models, which achieved F1-Scores of 40.56% and 54.90% respectively. The suggested model achieves a precision of 100%, a recall of 83.33%, and an F1-Score of 90.91% for R2L attacks. This represents a significant improvement compared to the LSTM and BidLSTM models, which achieved F1-Scores of 81.69% and 84.42% respectively. The DLNID model had a recall rate of 65.76% in detecting R2L attacks, however the accuracy value was not stated. The comparison analysis highlights the superior capability of the proposed model to effectively detect and categorize U2R and R2L assaults, which is crucial for enhancing the security of computer systems against these intrusions. The model's strong precision and recall rates indicate its effectiveness in reducing false positives and assuring accurate attack detection.

TABLE VII. A COMPARISON BETWEEN OUR METHOD AND SEVERAL IDS TECHNIQUES

Approach in Ref. No.	Model	Attacks	Precision%	Recall%	F1-Score%
[16]	DLNID	U2R	-	24.00	-
		R2L	-	65.76	-
[34]	LSTM	U2R	37.99	43.50	40.56
		R2L	97.97	70.04	81.69
[34]	BidLSTM	U2R	62.42	49.00	54.90
		R2L	98.97	73.60	84.42
Proposed model		U2R	**77.61**	**92.04**	**84.21**
		R2L	**100**	**83.33**	**90.91**

The model demonstrates an amazing ability to achieve heightened sensitivity in the classification of instances of attacks, a critical aspect of this specific domain. Our approach is applicable to a variety of application domains that exhibit a substantial class imbalance, in addition to attack detection. The adaptability of our approach allows it to be effectively employed in a variety of contexts where the precise identification of uncommon occurrences is crucial, providing a robust solution to analogous challenges in other industries.

V. CONCLUSION AND FUTURE WORK

This study introduces a new approach aimed at enhancing the efficiency of network traffic classification, specifically targeting U2R and R2L attacks. The integration of SHapley Additive exPlanations (SHAP) values and Long Short-Term Memory (LSTM) networks in our method improves the effectiveness of Intrusion Detection Systems (IDS) in detecting uncommon assaults such as U2R and R2L. The experimental data demonstrate significant progress in the classification of all four types of assaults, particularly U2R and R2L. The performance of our model outperforms current techniques, achieving an F1-Score of 84.21%, accuracy of 77.61%, and recall of 92.04% for U2R attacks. For R2L attacks, our model achieves a recall of 83.33%, precision of 100%, and an F1-Score of 90.91%. These results validate the effectiveness of our method in greatly enhancing classification accuracy, particularly in successfully identifying U2R and R2L attacks.

In the future, our research will investigate additional methods for selecting features in order to improve the detection capabilities of our model. Additionally, we intend to incorporate sophisticated techniques such as transformers into our plans. Furthermore, our objective is to evaluate the efficiency of our approach by employing current intrusion detection datasets that accurately replicate real-life traffic situations, such as the CIC-IDS2017 (Canadian Institude for Cybersecurity Intrusion Detection Evaluation) dataset.

CONFLICT OF INTEREST

The authors declare no conflict of interest.

AUTHOR CONTRIBUTIONS

Chadia El Asry: Conceptualization, Data curation, Formal analysis, Funding acquisition, Investigation, Methodology, Project administration, Software, Validation, Visualization, Writing–original draft. Ibtissam Benchaji: Conceptualization, Formal analysis, Investigation, Methodology, Software, Validation, Writing–original draft. Samira Douzi: Conceptualization, Formal analysis, Investigation, Methodology, Project administration, Supervision, Validation, Visualization, Writing–original draft. Bouabid El Ouahidi: Conceptualization, Methodology, Project administration, Software, Supervision, Validation. All authors had approved the final version.

REFERENCES

[1] A. Patel, Q. Qassim, and C. Wills, "A survey of intrusion detection and prevention systems," *Inf. Manag. Comput. Secur*, vol. 18, no. 4, pp. 277–290, 2010.

[2] A. Khraisat, I. Gondal, P. Vamplew, and J. Kamruzzaman, "Survey of intrusion detection systems: Techniques, datasets and challenges," *Cybersecurity*, vol. 2, no. 1, pp. 1–22, 2019.

[3] L. Yuan, H. Chen, J. Mai, C. N. Chuah, Z. Su, and P. Mohapatra, "Fireman: A toolkit for firewall modeling and analysis," in *Proc. the 2006 IEEE Symposium on Security and Privacy*, Berkeley/Oakland, CA, USA, 2006, pp. 15–213.

[4] J. Jang-Jaccard and S. Nepal, "A survey of emerging threats in cybersecurity," *Journal of Computer and System Sciences*, vol. 80, no. 5, pp. 973–993, 2014.

[5] A. Abbas, M. A. Khan, S. Latif *et al.*, "A new ensemble-based intrusion detection system for Internet of Things," *Arab. J. Sci. Eng.*, pp. 1–15, 2022. https://doi.org/10.1007/s13369-021-06086-5

[6] R. Chaudhari and S. Patil, "Intrusion detection system: Classification techniques and datasets to implement," *International Research Journal of Engineering and Technology*, vol. 4, no. 2, pp. 1860–1866, 2017.

[7] K. Rajasekaran, "Classification and importance of intrusion detection system," *Int. J. Comput. Sci. Inf. Secur.*, vol. 10, no. 8, 44, 2020.

[8] C. E. Asry, I. Benchaji, S. Douzi, and B. Ouahidi, "A robust intrusion detection system based on a shallow learning model and feature extraction techniques," *PloS One*, vol. 19, no. 1, e0295801, 2024. doi: 10.1371/journal.pone.0295801

[9] C. E. Asry, B. Ouahidi, and S. Douzi, "A deep learning model for intrusion detection with imbalanced dataset," *The International Conference on Intelligent System and Smart Technologies*, pp. 261–271, 2023. doi: 10.1007/978-3-031-47672-3_26

[10] G. C. Amaizu, C. I. Nwakanma, J. M. Lee, and D. S. Kim, "Investigating network intrusion detection datasets using machine learning," in *Proc. 2020 International Conference on Information and Communication Technology Convergence (ICTC)*, 2020, pp. 1325–1328.

[11] Y. Imrana, Y. Xiang, L. Ali, and Z. Abdul-Rauf, "A bidirectional LSTM deep learning approach for intrusion detection," *Expert Systems with Applications*, vol. 185. 115524, 2021. doi: 10.1016/j.eswa.2021.115524

[12] T. Le, J. Kim, and H. Kim, "An effective intrusion detection classifier using long short-term memory with gradient descent optimization," in *Proc. 2017 International Conference on Platform Technology and Service (PlatCon)*, Busan, 2017, pp. 1–6.

[13] F. Laghrissi, S. Douzi, D. Khadija, and B. Hssina, "IDS-attention: An efficient algorithm for intrusion detection systems using attention mechanism," *Journal of Big Data*, vol. 8, no. 1, 149, 2021. doi: 10.1186/s40537-021-00544-5

[14] F. Laghrissi, and S. Douzi, D. Khadija, and B. Hssina, "Intrusion detection systems using Long Short-Term Memory (LSTM)," *Journal of Big Data*, vol. 8, no. 1, 65, 2021. doi: 10.1186/s40537-021-00448-4

[15] R. H. Dong, X. Y. Li, Q. Y. Zhang, and H. Yuan, "Network intrusion detection model based on multivariate correlation analysis–long short-time memory network," *IET Inf. Secur.*, vol. 14, no. 2, pp. 166–174, 2019.

[16] Y. Fu, Y. Du, Z. Cao, Q. Li, and W. Xiang, "A deep learning model for network intrusion detection with imbalanced data," *Electronics*, vol. 11, no. 6, 898, 2022. doi: 10.3390/electronics11060898

[17] M. R. Norouzian and S. Merati, "Classifying attacks in a network intrusion detection system based on artificial neural networks," in *Proc. 13th International Conference on Advanced Communication Technology (ICACT2011)*, 2011, pp. 868–873.

[18] S. I. Popoola, B. Adebisi, M. Hammoudeh, G. Gui, and H. Gacanin, "Hybrid deep learning for botnet attack detection in the internet-of-things networks," *IEEE Internet Things J.*, vol. 8, no. 6, pp. 4944–4956, 2020. doi: 10.1109/JIOT.2020.3034156

[19] S. Li, F. Bi, W. Chen *et al.*, "An improved information security risk assessments method for cyber-physical-social computing and networking," *IEEE Access*, vol. 6, pp. 10311–10319, 2018.

[20] M. Habib, I. Aljarah, H. Faris *et al.*, "Multi-objective particle swarm optimization for botnet detection in internet of things," *Evolutionary Machine Learning Techniques*, pp. 203–229, 2020.

[21] P. Verma, S. Tapaswi, and W. W. Godfrey, "An adaptive threshold-based attribute selection to classify requests under DDoS attack in cloud-based systems," *Arab. J. Sci. Eng.*, vol. 45, no. 4, pp. 2813–2834, 2020.

[22] J. Gu and S. Lu, "An effective intrusion detection approach using SVM with naïve Bayes feature embedding," *Computers & Security*, vol. 103, 102158, 2021.

[23] A. Derhab, M. Guerroumi, A. Gumaei, L. Maglaras, M. A. Ferrag, M. Mukherjee, and F. A. Khan, "Blockchain and random subspace learning-based IDS for SDN-enabled industrial IoT security," *Sensors*, vol. 19, no. 14, 3119, 2019

[24] P. G. Jeya, M. Ravichandran, and C. S. Ravichandran, "Efficient classifier for R2L and U2R attacks," *Int. J. Comput. Appl.*, vol. 45, no. 21, 29, 2012.

[25] R. Fauzi and R. Al-Shammari, "New approach for classification R2L and U2R attacks in intrusion detection system," *International Journal of Biology, Pharmacy and Allied Sciences,* vol. 7, no. 4, pp. 1–14, 2018.

[26] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[27] I. Benchaji, S. Douzi, and B. Ouahidi, "Credit card fraud detection model based on LSTM recurrent neural networks," *Journal of Advances in Information Technology*, vol. 12, pp. 113–118, 2021. doi: 10.12720/jait.12.2.113-118

[28] S. Yan. Understanding LSTM and its diagrams. [Online]. Available: https://blog.mlreview.com/understanding-lstm-and-its-diagrams-37e2f46f1714

[29] S. M. Lundberg and S. I. Lee, "A unified approach to interpreting model predictions," *Advances in Neural Information Processing Systems*, vol. 30, pp. 4768–4777, 2017.

[30] P. R. Magesh, R. D. Myloth, and R. J. Tom, "An explainable machine learning model for early detection of Parkinson's disease using LIME on DaTSCAN imagery," *Computers in Biology and Medicine*, vol. 126, 104041, 2020. https://doi.org/10.1016/j.compbiomed.2020.104041

[31] NSL-KDD dataset. [Online]. Available: http://nsl.cs.unb.ca/nsl-kdd/

[32] S. lakhina, S. Joseph and B. Verma, "Feature reduction using principal component analysis for effective anomaly–based intrusion detection on NSL-KDD," *International Journal of Engineering Science and Technology*, vol. 2, no. 6, pp. 1790–1799, 2010.

[33] C. E. Asry, I. Benchaji, S. Douzi, and B. Ouahidi, "A robust intrusion detection system based on a shallow learning model and feature extraction techniques," *PloS One*. vol. 19. e0295801, 2024. doi: 10.1371/journal.pone.0295801

[34] Y. Imrana, Y. Xiang, L. Ali, Z. Abdul-Rauf, Y.-C. Hu, S. Kadry, and S. Lim, "χ^2-BidLSTM: A feature driven intrusion detection system based on χ^2 statistical model and bidirectional LSTM," *Sensors*, vol. 22, no. 5, 2022. doi: 10.3390/s22052018