technologies

Article

A Hypertuned Lightweight and Scalable LSTM Model for Hybrid Network Intrusion Detection

Aysha Bibi [1], Gabriel Avelino Sampedro [2,3], Ahmad Almadhor [4], Abdul Rehman Javed [5] and Tai-hoon Kim [6,*]

[1] Department of Cyber Security, Air University, Islamabad 44000, Pakistan
[2] Faculty of Information and Communication Studies, University of the Philippines Open University, Los Baños 4031, Philippines
[3] College of Computer Studies, De La Salle University, 2401 Taft Ave., Malate, Manila 1004, Philippines
[4] Department of Computer Engineering and Networks, College of Computer and Information Sciences, Jouf University, Sakaka 72388, Saudi Arabia
[5] Department of Electrical and Computer Engineering, Lebanese American University, Byblos P.O. Box 36/S-12, Lebanon
[6] School of Electrical and Computer Engineering, Yeosu Campus, Chonnam National University, 50, Daehak-ro, Yeosu-si 59626, Jeollanam-do, Republic of Korea
* Correspondence: taihoonn@chonnam.ac.kr

Abstract: Given the increasing frequency of network attacks, there is an urgent need for more effective network security measures. While traditional approaches such as firewalls and data encryption have been implemented, there is still room for improvement in their effectiveness. To effectively address this concern, it is essential to integrate Artificial Intelligence (AI)-based solutions into historical methods. However, AI-driven approaches often encounter challenges, including lower detection rates and the complexity of feature engineering requirements. Finding solutions to overcome these hurdles is critical for enhancing the effectiveness of intrusion detection systems. This research paper introduces a deep learning-based approach for network intrusion detection to overcome these challenges. The proposed approach utilizes various classification algorithms, including the AutoEncoder (AE), Long-short-term-memory (LSTM), Multi-Layer Perceptron (MLP), Linear Support Vector Machine (L-SVM), Quantum Support Vector Machine (Q-SVM), Linear Discriminant Analysis (LDA), and Quadratic Discriminant Analysis (QDA). To validate the effectiveness of the proposed approach, three datasets, namely IOT23, CICIDS2017, and NSL KDD, are used for experimentation. The results demonstrate impressive accuracy, particularly with the LSTM algorithm, achieving a 97.7% accuracy rate on the NSL KDD dataset, 99% accuracy rate on the CICIDS2017 dataset, and 98.7% accuracy on the IOT23 dataset. These findings highlight the potential of deep learning algorithms in enhancing network intrusion detection. By providing network administrators with robust security measures for accurate and timely intrusion detection, the proposed approach contributes to network safety and helps mitigate the impact of network attacks.

Keywords: deep learning; machine learning; Long-short-term-memory (LSTM); cyberattacks; network intrusion detection; cyber security

Citation: Bibi, A.; Sampedro, G.A.; Almadhor, A.; Javed, A.R.; Kim, T.-h. A Hypertuned Lightweight and Scalable LSTM Model for Hybrid Network Intrusion Detection. *Technologies* **2023**, *11*, 121. https://doi.org/10.3390/technologies11050121

Academic Editors: Mohammed Mahmoud and Lipo Wang

Received: 4 June 2023
Revised: 26 August 2023
Accepted: 5 September 2023
Published: 7 September 2023

1. Introduction

Due to expeditious growth in computer technology, individuals all over the globe are adopting more internet services than ever [1–3]. Furthermore, the diversity of cyberattacks has increased due to increased internet services [4–7]. For example, network worms, malevolent spying, and aggressive assaults seriously threaten people's data security and physical safety [8]. As a result, data security and security protocols have become critical for both people and society [9–12]. Firewalls are extensively used and frequently installed as a fundamental security measure. However, it no longer remains

appropriate and requires strong security (e.g., governmental entities, military assets, etc.) [13–15] owing to the difficulties of human setup and the latency for new forms of assaults. Network security researchers have proposed a new approach for identifying and addressing anomalous behavior through intrusion detection systems (IDSs) to combat these threats [1].

Annually, breaches in IT networks cost trillions of dollars, which is predicted to climb in the future [16]. As a response, cybersecurity has been a primary focus in recent years. Monitoring and analyzing network traffic data is crucial for recognizing potential attack patterns [17–19]. Therefore, in this scenario, firms and IT organizations worldwide have been spending on data science to build increasingly sophisticated Intrusion Detection Systems (IDS) to stop hostile attacks and ensure greater cybersecurity [20,21]. A collection of approaches from computers, statistics, and information and technology, such as Machine Learning, are included in this research. Due to the massive heterogeneous data generated by numerous sources, traditional data analytics and machine learning approaches could be more valuable and efficient in dealing with such security and privacy concerns [22–25]. Furthermore, novel technologies such as federated learning are helping to preserve the privacy of users in various environmental setting settings [26–29]. Federated learning has been used for network data security [4,30], healthcare data security [31], game theory [32], vehicular data [33,34], and smart city applications [35]. Studies have found that data poisoning attacks and other cyberattacks can also compromise this technique [36–39]. Furthermore, traditional machine learning approaches have a limited processing complexity and need help discovering complicated non-linear relationships in large datasets.

As a result, to overcome the above-mentioned restrictions and improve intrusion detection performance, we integrate the classic data analysis and statistical approaches with current breakthroughs in Machine learning. Deep Learning technologies are mainly used to create more advanced security Intrusion Detection Systems (IDS) [40]. A deep learning system is suggested in this paper to distinguish between regular and abnormal network actions. Data processing, feature extraction, and classification were the primary components presented in the suggested framework.

Conversely, an auto-encoder deep classifier is presented in the classification algorithm to distinguish distinct dataset groups. Binary and multi-classification are the two types of classification employed. The binary class has two labels: normal and abnormal, whereas the multi-classification includes five labels: Denial of Service (DoS), Normal, Probing (Probe), Root to Local (R2L), and User to Root(U2R). DoS attacks have included those that force a computer to slow down or close down by delivering more data to the server than the host can manage. DoS attacks disrupt lawful network traffic or the accessibility of services. R2L attacks have included those that allow unauthorized local access to a device by delivering misleading information to the host. U2R attacks include those that grant root privileges. In this example, the attacker discovers internal weaknesses and regularly utilizes the device. AE, LSTM, and other machine learning classifiers such as MLP, L-SVM, Q-SVM, LDA, and QDA are employed.

Contribution: This research has made several significant contributions, including:

- Development of a novel intrusion detection system (IDS) that leverages data analytics and deep learning technologies. The proposed model represents a significant intrusion detection advancement and contributes to network security in commercial and industrial settings.
- The developed system can accurately distinguish a wide range of cyberattacks. This achievement is particularly noteworthy as accurately identifying and categorizing cyber threats is critical for effective network security.
- The proposed approach utilizes various classification algorithms, including the AutoEncoder (AE), Long-short-term-memory (LSTM), Multi-Layer Perceptron (MLP), Linear Support Vector Machine (L-SVM), Quantum Support Vector Machine (Q-SVM), Linear Discriminant Analysis (LDA), and Quadratic Discriminant Analysis (QDA).

- To validate the effectiveness of the proposed approach, three datasets, namely the NSL KDD, CICIDS2017, and IOT23 datasets, are used for experimentation. The results demonstrate impressive accuracy, particularly with the LSTM algorithm, achieving a 97.7% accuracy rate on the NSL KDD dataset, 99% accuracy rate on the CICIDS2017 dataset, and 98.7% accuracy on the latest IOT23 dataset.
- The research provides a comparative analysis demonstrating the superiority of the proposed model in terms of accuracy and overall performance compared to various existing models. This contribution helps advance intrusion detection by offering more accurate and efficient solutions to a specific problem.

The paper comprises several sections, each focusing on a distinct aspect of the study. Section 2 presents the relevant work to intrusion detection, while Section 3 demonstrates the Proposed Approach. Section 4 covers the Experimental Settings, and Section 5 reports the Results and Discussion. The paper is concluded in Section 6, which offers a summary and recommendations for further research.

2. Literature Review

The authors in [41] used a support vector machine (SVM) and a genetic algorithm to adjust the correctness of the Model by tweaking the SVM attributes' selection, variables, and weights. Tang et al. [42] introduced a deep neural network to detect intrusion and software to define the network context. The NSL KDD dataset was used to train a tri-layer neural network. Only six characteristics were employed, and only two-way differentiation was used. The test findings showed a 75% accuracy rate.

Ahmin et al. [43] introduced a two-phase NIDS framework that detects network assaults using numerous categorization approaches, including the REP Tree, JRip procedure, and Forest PA strategy. The algorithm they used achieved an accuracy of 96.66% on the CICIDS 2017 dataset. Faker et al. also created a geographically dispersed NIDS model with a DNN and two ensemble approaches [44]. The framework was tested on the UNSW NB15 and CICIDS2017 datasets. The authors of [45] proposed an NIDS architecture that combines CNN and LSTM. They also used category weights to optimize the model training phase, lowering the number of imbalanced instances in the data set used for training. The suggested NIDS was tested on the CICIDS2017 data set comprising seven different forms of network traffic and obtained an accuracy of 98.67% with a rate of false alarms of roughly 0.47%. This NIDS, however, proved useless in detecting Heartbleed and SSH-Patator attacks. Jiang et al. devised a NIDS approach that uses hybrid testing and a deep hierarchical network [46] to overcome this issue. They also used one-side filtering to deal with distortion on minority tags and fake minority oversampling to boost minority tag sample sizes. This method can increase detection effectiveness on skewed datasets while reducing training time. After refining the network properties, the authors retrieved spatial features using CNN and temporal data using bilateral long short-term memory. The results show that their suggested strategy surpasses previous studies.

The possibility of employing NIDS on IoT edge networks was proved in [47,48] by implementing machine learning designs, Isolation Forest (iForest), and the Local Outlier behavior Factor (LOF) on resource-constrained devices to detect network threats. Midi et al. created Kalis, a network surveillance and management tool that detects fraudulent traffic using signature-based and anomaly-based algorithms. However, Kalis has limitations, such as routing assaults and the need for specialized detection modules, which can lead to suboptimal detection performance. Similarly, in [49], a deep hierarchical model based on CNN and Gated Recurrent Units (GRU) was developed to detect abnormal network traffic at the packet level, achieving 99% accuracy with a processing rate of approximately 20,000 packets per second on three datasets. Additionally, other research has explored sub-domains of cybersecurity, such as the work in [50] that investigated time-series anomaly detection features and [51] that proposed a mutual authentication scheme with minimal complexity and easy installation for resource-constrained devices.

Xu et al. [52] created an IDS based on deep neural networks that successfully classified data from the NSL-KDD sample. They did, though, use the Tenfold cross-validation approach on the actual data to assess the implementation of the suggested approach. To identify the classes in the NSL-KDD sample, Han et al. [53] suggested a small auto-encoder. The researchers stated a 98% accuracy rate, although they sped up the experiment by scrambling and reconstructing the essential information into numerous separate datasets. Yin et al. [54] created an IDS based on a Recurrent Neural Network (RNN). The researchers utilized the NSL KDD dataset as a reference and conducted binary and multi-classification, with 83% and 81% accuracy percentages, respectively. The authors employed the NSL-KDD dataset in their investigation and reported a multi-classification accuracy of 85%. Table 1 presents the previous relevant research summary.

Table 1. A C omprehensive Overview of Existing Research in the Field.

Author	Year	Algorithm	Contribution	Limitations
Shamsinejad et al. [55]	2017	k-Means	A K-MEANS clustering classifier was proposed to enhance detection accuracy	It only considers a single feature for intrusion detection, which may not capture all relevant information.
Sun et al. [41]	2018	SVM, and Genetic Algorithm	SVM features are optimized using a genetic algorithm. It enables selection parameters and weights to be optimized	Lacks diversity, potentially affecting the generalizability of findings.
Xinqian et al. [56]	2019	Random Forest	Detected abnormal network behavior using a multilevel random forest model	Only perform experiment on conventional machine learning algorithms
Wang et al. [57]	2020	CNN and LSTM	To detect each attack type, a model based on CNN and LSTM is proposed	a small sample size and limited generalizability of the findings.
Cao et al. [58]	2022	CNN and GRU	Proposed a NIDS model using CNN and GRU, with contributions including feature selection, hybrid sampling, and the introduction of CNN, GRU, and attention mechanisms to improve model performance	Limited effectiveness in detecting unknown attacks and limited generalizability
Qazi et al. [59]	2023	CNN and RNN	To detect and classify malicious traffic, analyzes existing ML/DL techniques	Accessibility is limited by the high-end hardware used in the experiments
Mhawi et al. [60]	2022	Ensemble of ML Classifiers	Provided improved NIDS with a comparative analysis of different techniques and classifiers.	Absence of control group, self-reported data, and limited generalizability.

3. Proposed Approach

Figure 1 shows the proposed architectural design of the system. Firstly, we pre-process the data by data normalization using the standard scalar within the range of 0 and 1. Then, we use one-hot encoding to transform category characteristics into numerical features. Finally, the detection operation is evaluated using the classifiers mentioned above.

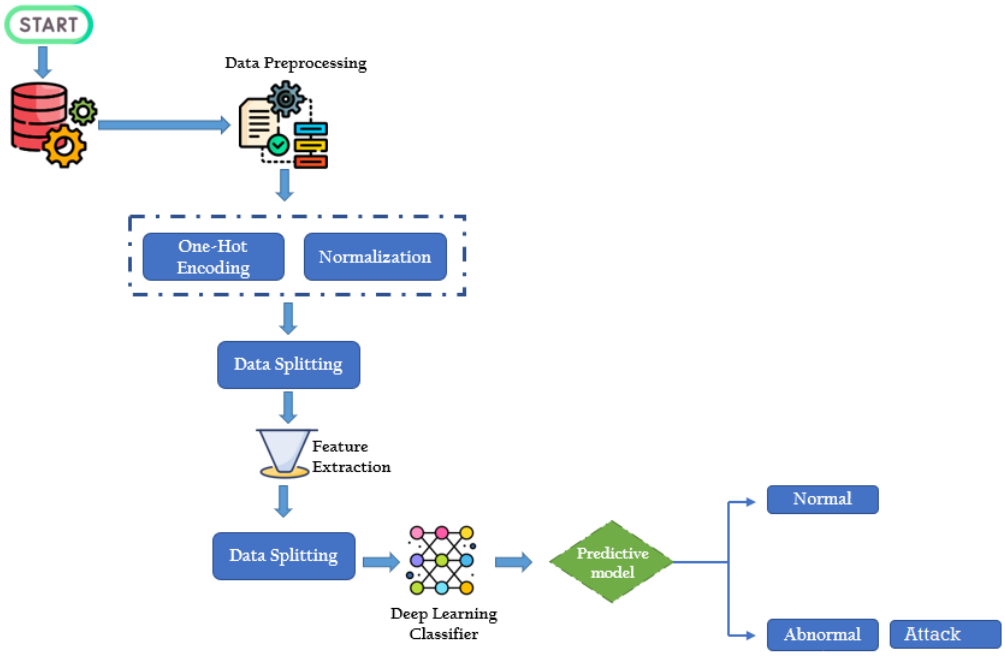


Figure 1. Illustration of the proposed Model: framework diagram with key modules.

3.1. Dataset

Three datasets, NSL KDD [61], CICIDS2017, and IOT23, are utilized in this research. The NSL KDD dataset is one of the common datasets used to assess the functioning of intrusion detection systems (IDS), and it is utilized in this paper's experimental data. It comprises 125,973 traffic examples in the training set (KDD-Train) and 22,544 traffic data in the test set (KDD-Test). There were seventeen more assault types in the test dataset that were not included in the training dataset for classification. The NSL-KDD dataset has forty-two-dimensional characteristics, one being a classification tag and the other being feature identifiers. There are two types of classification labeling for binary classification: normal and abnormal. The classification categories for multiclassification are Normal, Denial of Service (DOS), Root to Local (R2L), User to Root (U2R), and Probing (probe). The CICIDS2017 dataset is a collection of network traffic data used for detecting intrusions and analyzing network behavior. It contains information about network connections, such as the source and destination IP addresses, port numbers, protocols, packet sizes, and communication durations. The dataset includes 79 features that help researchers and analysts study network traffic patterns and identify potential security threats.

The IoT-23 [62] dataset, established in January 2020, offers real network traffic data from IoT devices. It includes 20 malware instances and three benign IoT captures. The dataset, funded by Avast Software, aids in developing IoT malware detection algorithms. It encompasses 23 scenarios, 20 with malware-infected traffic executed on Raspberry Pi devices and three with benign IoT traffic. This dataset provides valuable insights for IoT research. It contains information about malicious and benign captured packets. The IoT-23 dataset employs specific labels to categorize network traffic. These include "Attack" for exploiting vulnerabilities, "Benign" for harmless connections, "C&C" for Command and Control server links, "DDoS" for Distributed Denial of Service flows, "FileDownload" for downloading files, "HeartBeat" for C&C tracking, "Mirai" for Mirai botnet-like patterns, "Okiru" for Okiru botnet-like behaviors, "PartOfAHorizontalPortScan" for horizontal port scanning, and "Torii" for connections resembling the Torii botnet.

3.2. Pre-Processing

The datasets contain outliers or inconsistent values, and data pre-processing is necessary for building a model. Our work comprises two components: the normalization of

the data and one-hot encoding. Using the conventional scalar normalizing approach, the numeric feature values were mapped into the numeric range 0 and 1. A sample's standard score is computed as it arises in Equation (1) [63]:

$$Z = \frac{si - min(s)}{max(s) - min(s)} \quad (1)$$

where $s = (s1, \ldots, sn)$ and Z is the ith normalized data point. Three categorical elements are present in the dataset (service, flag, and protocol type). We converted these category data into numerical features using the one-hot-encoding approach. The $z2$ feature has three properties. One hot encoding approach was used to convert them into binary data: [1,0,0], [0,1,0], [0,0,1], respectively. Also transformed into one-hot-encoding matrices were the $z3$ and $z4$ attributes (service and flag).

3.3. Extraction of Features

In feature extraction, the goal is to reduce the dimensionality of the dataset while retaining the most relevant information. The Pearson correlation matrix [64] is used for this purpose as it helps identify the most correlated elements in the dataset. The coefficient of correlation, which ranges from -1 to 1, quantifies the degree and trajectory of the linear link between the two factors. Using the Pearson correlation matrix in our feature extraction process, we identified the most correlated variables and reduced the dimensionality of our dataset while retaining the most important information. This ultimately resulted in a more efficient and effective analysis of our data. Its ability to identify the most correlated variables, the Pearson correlation matrix, can also be used to identify potential outliers or anomalies in the dataset. Outliers are data points that lie far away from the rest of the data points and can significantly impact the analysis results. By identifying these outliers, the Pearson correlation matrix can improve the accuracy and reliability of the analysis. It is worth noting that using the Pearson correlation matrix assumes that the variables in the dataset have a linear relationship. In cases where the variables have a nonlinear relationship, other feature extraction methods, such as the principal component analysis (PCA) or independent component analysis (ICA), may be more appropriate. Furthermore, while feature extraction can be a powerful tool for reducing the dimensionality of the dataset, it can also result in the loss of important information. Therefore, it is important to carefully select the extracted features and ensure they are relevant to the analysis.

3.4. Classification

In our research, we leveraged three datasets, namely NSL KDD, CICIDS2017, and IOT23, to evaluate the effectiveness of our proposed approach. The NSL KDD dataset comprises two class categories: regular class labels and abnormal classes. For the five class labels, including Denial of Service (DOS), Normal, Root to Local (R2L), and User to Root (U2R), we employed two deep learning classifiers, namely LSTM and AE, along with three other conventional KNN, L-SVM, LDA classifiers. Similarly, the CICIDS2017 dataset consists of two class labels: Benign and Intrusion. We also applied the same approach to this dataset, utilizing deep learning and conventional classifiers. The IOT23 dataset also consists of two classes labels: Benign and Malicious.

Binary Classification: In binary classification, we change the attack labels into 'NORMAL' and 'ABNORMAL.' First, we create the data frame with binary labels 'NORMAL' and 'ABNORMAL' and then encode the labels into 0 and 1. We have 53% of standard data and 47% of abnormal labels in the NSL KDD dataset for binary classification. In abnormal labels, we have four types of attacks. In CICIDS2017, we have 56% Intrusion and 43% Benign labels for binary classification.

Multi Classification: In multi-classification, we use NSL KDD, where we have five class tags: Normal, Denial of service (DoS), Probe, Root to Local (R2L), and User to Root (U2R). First, we created a data frame for multi-class labels and then performed label encoding

for multi-class such as 0, 1, 2, 3, 4. We have 53.46% normal, 36.46% DoS, 9.26% R2L, 0.79% Probe, and 0.04% U2R labels in the dataset in the case of multi-classification.

3.4.1. Auto-Encoder (AE)

In an auto-encoder, the input and output dimensions are identical. It is an unsupervised learning network [65]. It contains two modules: the first one is the encoder, and the other is the decoder module. AE uses deep learning techniques to identify the maximum accurate features from the input information while conserving as much information as feasible. The encoder reduces the data size, which the decoder reconstructs into the source data. We wanted to develop an auto-encoder that can reduce the dimensionality and boost data resilience to familiarize with complicated network situations, which can accomplish better data dimensionality reduction than previous dimensionality reduction approaches. Auto-encoder architecture is given in Figure 2.

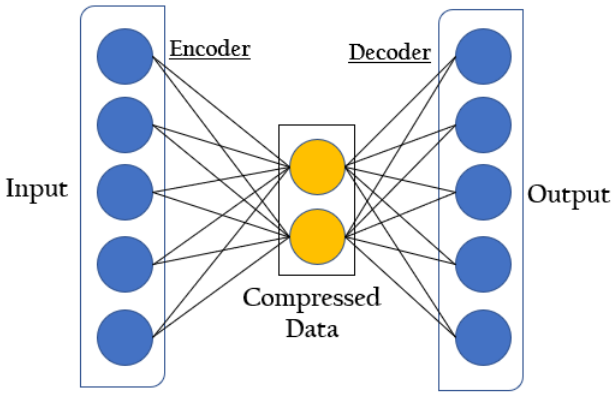


Figure 2. The architecture of Autoencoder [66].

3.4.2. Long Short-Term Memory Networks (LSTM)

LSTM [67,68] provides storage cells and cell states to solve the recurrent neural network's (RNNs) long-term reliance problem. LSTM systems are recurrent neural networks that deal with instances where RNNs fail. A recurrent Neural Network (RNN) is a network that operates on the current input while considering the previous output (feedback) and keeping it for a short moment in its memory (short-term memory). The most common applications are speech processing, non-Markovian management, and music composition, to name a few. However, RNNs have several drawbacks. Long short-term memories (LSTMs) are used to overcome lengthy time gaps in some issues, and they can also handle noise, dispersed representations, and continuous values. There is no requirement to preserve a finite number of states from the beginning with LSTMs, as in the hidden Markov model. The method teaches that input and output are biased, and other parameters are available with LSTMs. As a result, no precise modifications are required. The LSTM classifier contains 50 neurons, a batch size of 5000, 100 epochs, and a sigmoid activation layer fpr classification data.

3.4.3. Discriminant Analysis (DA)

The objective of the DA classifier is to keep dimensionality to a minimum while ensuring sufficient class distinction. It accomplishes this by translating the dataset onto a smaller space with maximal class separation and minimal diffusion of samples from a similar class. It is a numerical approach used in machine learning.

3.4.4. Support Vector Machine (SVM)

SVM is a probabilistic learning theory-based approach. The optimum hyperplane that gives the most separation across classes is found via SVM. A Support Vector Machine classifier with L-SVM and Q-SVM is developed in this work.

3.4.5. Multi-Layer Perceptron (MLP)

It is a feed-forward artificial neural network simulation that converts raw data sets into a collection of relevant results. MLP and AE designs have the same architecture. The MLP classifier contains one hidden layer, 50 neurons, and a sigmoid activation layer for classification tasks.

4. Experimental Analysis and Result

All tests were performed on Google Colab using the Python language and the SK-learn library for creating and modeling in this research. Different systems of measurement are used to measure the implementation of the proposed work, such as precision in Equation (2), recall in Equation (3), accuracy in Equation (4), and F-measure in Equation (5).

$$Precision = \frac{TP}{TP + FP} \tag{2}$$

$$Recall = \frac{TP}{TP + FN} \tag{3}$$

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN} \tag{4}$$

$$F\text{-}measure = 2 \times \frac{Precision \times Recall}{Precision + Recall} \tag{5}$$

where the True Positive (TP) is the number of abnormal instances accurately identified; the number of instances accurately identified as normal is denoted by the True Negative (TN). The amount of normal traffic patterns misclassified as abnormal is called the False Positive (FP); the amount of anomalous traffic patterns misclassified as normal is called the False Negative (FN). Precision is a measure of the accuracy of positive predictions, while recall is a measure of the completeness of positive predictions. The F1 score combines both precision and recall into a single metric.

4.1. Evaluation and Results with NSL KDD

Two types of classification are used in this case, such as binary classification and multiclassification, to check the performance of the proposed system. In binary classification, two classes are there, such as normal and abnormal, and in the case of multi-class, five classes are present in the dataset.

Table 2 shows the accuracy of binary and multi-classification. For binary classification, Long Short-term Memory (LSTM) and MLP beat all other classifiers with an accuracy of 97.7%. In the case of Multi-classification, MLP and AE outperform with an accuracy of 97%.This could be attributed to the ability of LSTM to capture long-term dependencies and patterns in sequential data, a characteristic of network intrusion detection data. Additionally, MLP is a versatile and powerful feed-forward neural network that can learn complex non-linear relationships between input and output data.

The results of binary classification with NSL KDD dataset studies are shown in Table 3. The LSTM and MLP classifiers detected the normal category with great precision, whereas the Q-SVM and QDA classifiers detected the abnormal category with a precision of 0.99. In contrast to other classifiers, the precision of the QDA classifier is as low as 0.63.This could be due to the assumption of QDA that each class follows a Gaussian distribution with a different covariance matrix. However, this assumption might not hold in practice for all classes, leading to a lower precision in some cases. In contrast, other classifiers like L-SVM and Q-SVM have a more flexible decision boundary, which can adapt better to the complex distribution of data points, leading to higher precision. The Confusion matrix of the LSTM classifier for binary classification is shown in Figure 3a. The Accuracy vs. Epoch and the Loss vs. Epoch of the LSTM classifier are shown in Figure 3b,c.

Table 2. Accuracy Comparison of Proposed Models for Binary and Multi-Classification.

Models	Binary Classification	Multi Classification
LSTM	97.7%	95%
MLP	97.7%	97%
L-SVM	96.6%	95%
Q-SVM	95.7%	92%
AE	91%	97%
LDA	96.7%	93%
QDA	68%	44%

Table 3. Evaluation of Proposed Model's Precision in Binary Classification.

Label	LSTM	AE	MLP	L-SVM	Q-SVM	LDA	QDA
Normal	0.97	0.80	0.97	0.96	0.93	0.96	0.63
Abnormal	0.96	0.88	0.98	0.97	0.99	0.97	0.99

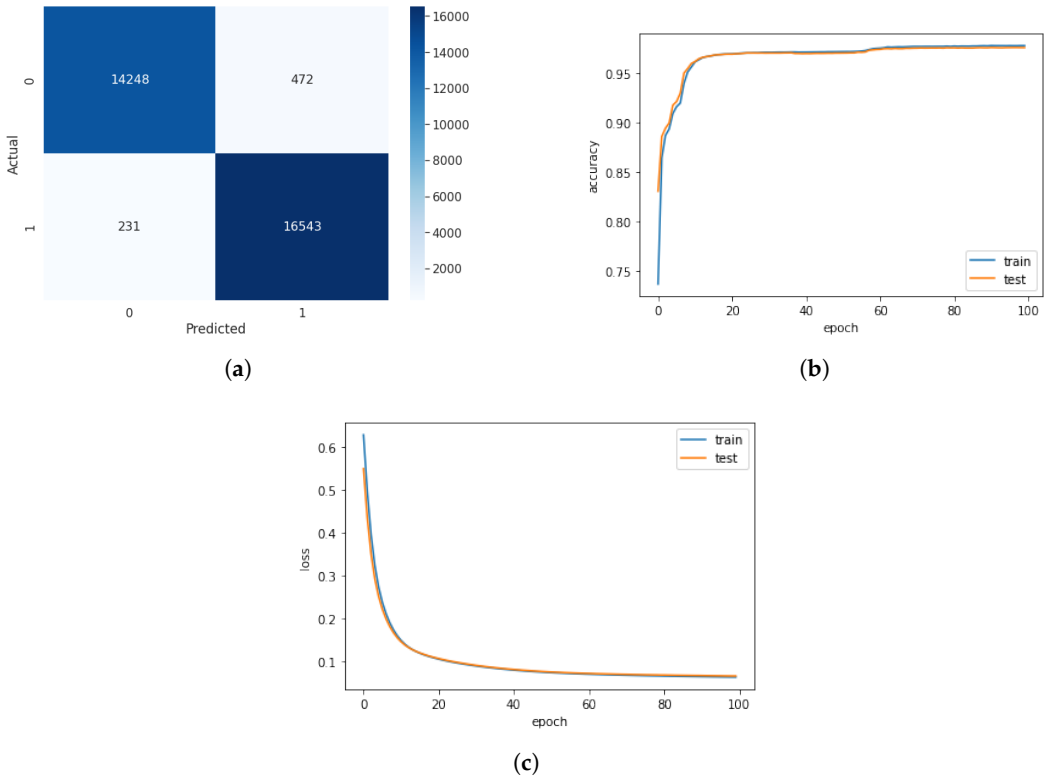


Figure 3. Illustration of the outcomes achieved by LSTM Classifier. (**a**) Confusion Matrix of LSTM for Binary Classification; (**b**) LSTM Classifier Accuracy vs. Epochs for Binary Classification; (**c**) LSTM Classifier Loss vs. Epochs for Binary Classification.

With a recall of 0.99, L-SVM and QDA exceed all other classifiers regarding the Recall because these classifiers are based on the principles of the maximum margin and quadratic discriminant analysis, respectively. These techniques have been shown to work well with imbalanced datasets, such as the NSL-KDD dataset used in this study, which contains a small number of positive instances relative to negative instances. In the case of L-SVM, it is a powerful binary classifier that tries to maximize the margin between the positive and

negative instances. This makes it less susceptible to overfitting and can help it perform well on imbalanced datasets. On the other hand, QDA is a probabilistic classifier that estimates the probability density function of each class. By modeling the distribution of each class separately, QDA can capture the differences between the two classes more accurately and provide better performance on imbalanced datasets. Q-SVM beat L-SVM in detecting the normal class using SVM classifiers. Regarding discriminant analysis, QDA surpassed LDA regarding the normal sample Recall (0.99). The LSTM classifier better recognized anomalous classes (Recall of 97%) as shown in Table 4.

Table 4. Evaluation of Proposed Model's Recall in Binary Classification.

Label	LSTM	AE	MLP	L-SVM	Q-SVM	LDA	QDA
Normal	0.98	0.97	0.98	0.97	0.99	0.98	0.99
Abnormal	0.97	0.96	0.96	0.96	0.92	0.96	0.33

The F1 score of the LSTM, MLP, and LDA classifiers in classifying the normal category was 97%. Compared to Q-SVM, the L-SVM f1-score is higher (97%). Regarding detecting standard samples, the LDA classifier outperformed the QDA. With a score of 99%, LSTM surpassed all other classifiers in recognizing abnormal samples because it is a type of RNN well-suited for sequence data processing. In the NSL-KDD dataset, abnormal samples can be viewed as network traffic sequences. LSTM models are designed to remember past events and process input sequences time-dependent, making them ideal for detecting patterns in sequence data. The LSTM model can learn complex patterns in the sequence of network traffic and accurately identify abnormal samples with a recall of 97%. The LSTM model can also detect the normal category with high precision, making it an effective binary classifier for the NSL-KDD dataset. LSTM surpassed all other classifiers in recognizing abnormal samples shown in Table 5.

Table 5. F1-Score Analysis of the Proposed Model for Binary Classification.

Label	LSTM	AE	MLP	L-SVM	Q-SVM	LDA	QDA
Normal	0.97	0.88	0.97	0.97	0.96	0.97	0.77
Abnormal	0.99	0.92	0.98	0.96	0.95	0.96	0.50

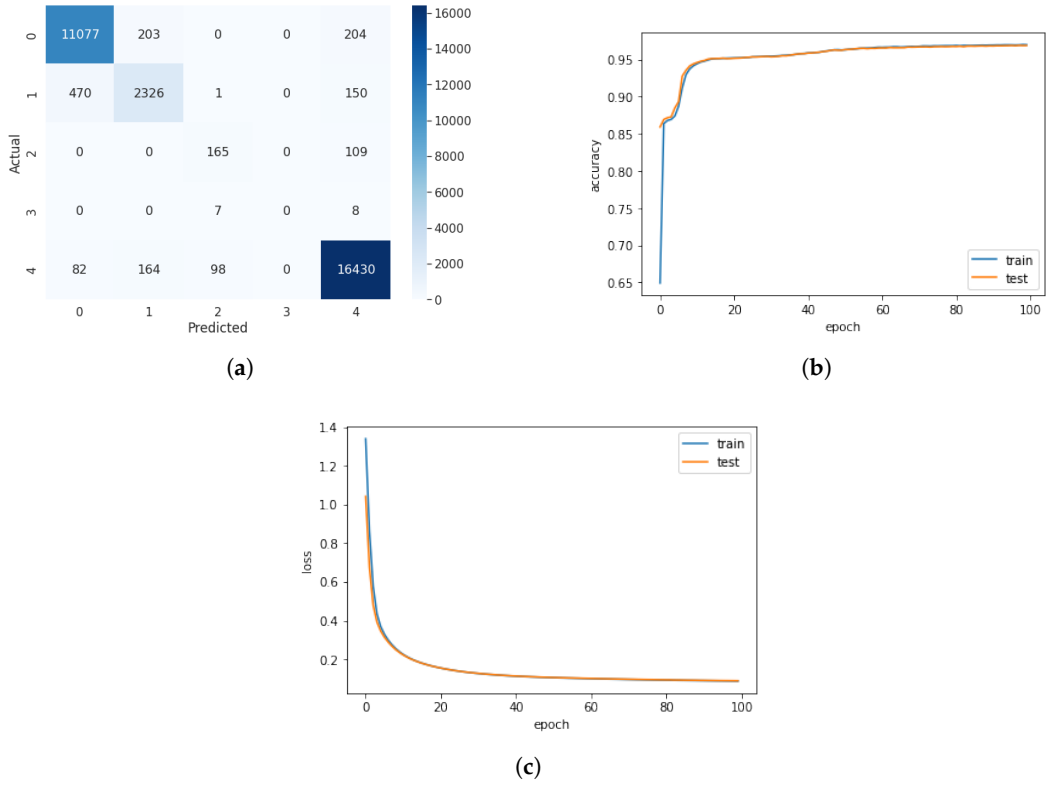
The deep L-SVM, Q-SVM, LDA, QDA, and MLP classifiers were compared similarly to the binary classification study. Table 6 shows that AE and MLP classifiers outperformed in the case of precision in all other classifiers. The confusion matrix, Accuracy vs. Epoch, and Loss vs. Epoch of the MLP classifier are given in Figure 4a–c. Recall, and f1-Score AE outperformed all other classifiers as shown in Tables 7 and 8.

Table 6. Precision of Proposed Model for Multi-Class Classification.

Label	AE	MLP	LSTM	L-SVM	Q-SVM	LDA	QDA
Normal	0.98	0.98	0.97	0.97	0.91	0.97	0.49
DoS	0.97	0.96	0.96	0.95	0.96	0.94	0.99
Probe	0.88	0.92	0.88	0.86	0.96	0.88	0.97
R2L	0.78	0.83	0.69	0.61	0.00	0.31	0.03
U2R	0.01	0.00	0.03	0.00	0.00	0.03	0.00

The standard system was used to assess the proposed IDS features and efficacy. The statistical analysis was used to extract the most correlated features and was fed into deep (AE, LSTM) and ML techniques and MLP, L-SVM, Q-SVM, LDA, and QDA. Moreover, the experimental results demonstrated that the MLP classifier attained the most acceptable performance for binary classification (97.7%) and multi-classification (97%) compared with L-SVM, Q-SVM, LDA, and QDA classifiers. The AE classifier also accomplished a high accuracy of 98% compared to the LSTM classifier.

Table 9 compares the proposed Model's metrics to those of various reference models. The suggested approach outperforms other models in terms of overall performance. The suggested Model exceeds its comparable counterparts by 97.7%. The percentages are 97%, 98%, and 97%, respectively.



(**a**) (**b**)

(**c**)

Figure 4. Illustration of the outcomes achieved by the MLP Classifier. (**a**) Confusion Matrix of MLP for Multi classification; (**b**) Accuracy vs. Epochs of MLP Classifier; (**c**) Loss vs. Epochs of MLP Classifier.

Table 7. Proposed Model's Recall for Multi Classification.

Label	AE	MLP	LSTM	L-SVM	Q-SVM	LDA	QDA
Normal	0.98	0.97	0.97	0.98	1.00	0.95	0.53
DoS	0.97	0.96	0.93	0.96	0.94	0.96	0.41
Probe	0.87	0.93	0.81	0.79	0.61	0.73	0.06
R2L	0.81	0.88	0.74	0.60	0.00	0.89	1.00
U2R	0.02	0.00	0.04	0.00	0.00	0.47	0.00

Table 8. Proposed Model's F1-Score for Multi Classification.

Label	AE	MLP	LSTM	L-SVM	Q-SVM	LDA	QDA
Normal	0.98	0.95	0.96	0.98	0.95	0.96	0.51
DoS	0.92	0.93	0.90	0.96	0.95	0.95	0.58
Probe	0.91	0.88	0.83	0.82	0.74	0.80	0.11
R2L	0.79	0.81	0.81	0.61	0.00	0.52	0.06
U2R	0.04	0.00	0.05	0.00	0.00	0.06	0.00

Table 9. Performance Comparison of the Proposed Model against Existing Models.

Models	Accuracy	Precision	Recall	F1-Score
SVM-IDS [69]	82%	-	-	-
CNN [70]	80%	-	-	-
TES-IDS [71]	85%	88%	86%	85%
Autoencoder [72]	84%	87%	80%	81%
CNN & BiLSTM [46]	83%	85%	84%	85%
DLNID [73]	90%	86%	93%	89%
Proposed Model	97.7%	97%	98%	97%

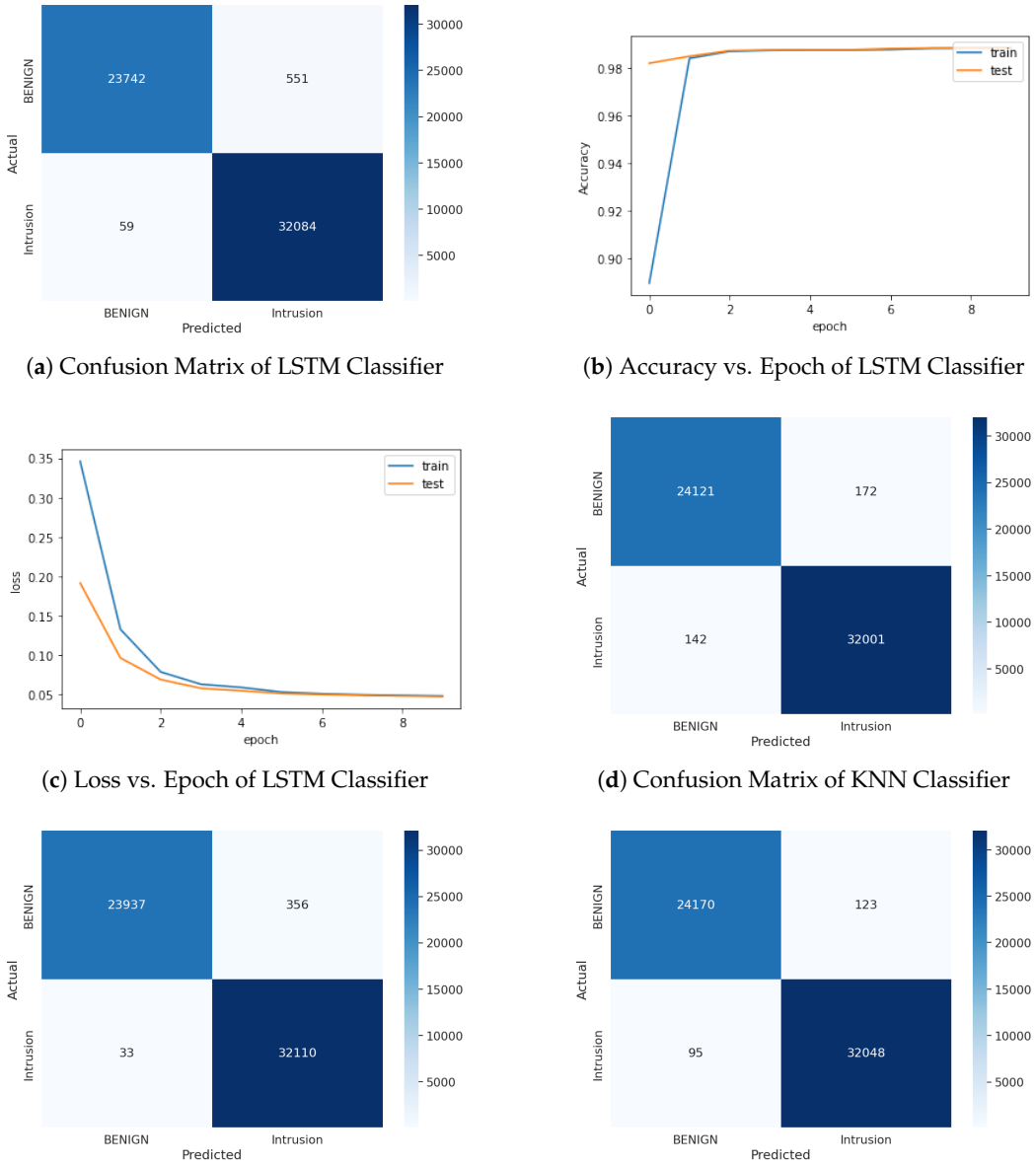
4.2. Evaluation and Results with CICIDS2017

Table 10 shows the accuracy of the binary classification. LSTM beat all other classifiers with an accuracy of 99.2%. Due to several factors, LSVM, KNN, and QDA also perform better on the CICIDS2017 dataset for classification. The dataset's characteristics, such as the linear separability and class distribution, may align well with the decision boundaries formed by LSVM and QDA. KNN's ability to capture local characteristics in the data could contribute to its effectiveness. Moreover, the suitability of these algorithms for high-dimensional datasets and non-linear decision boundaries and their ability to leverage informative features may also contribute to their superior performance.

Table 10. Accuracy Comparison of Proposed Models for Binary Classification.

Models	Accuracy	Precision	Recall	F1-Score
LSTM	99.2%	99%	99%	99%
AE	98.5%	98%	98%	98%
MLP	94.6%	94%	94%	96%
L-SVM	99%	98%	98%	98%
Q-SVM	64%	64%	64%	63%
KNN	99%	98%	99%	98%
LDA	96%	96%	96%	96%
QDA	99%	99%	98%	98%

The LSTM classifier detected the benign category with great accuracy, whereas the KNN and QDA classifiers detected the Intrusion category with an accuracy of 99%. In contrast to other classifiers, the accuracy of the Q-SVM classifier is as low as 0.64. This could be due to the assumption of Q-SVM that it relies on quantum algorithms and quantum computing resources. The Confusion matrix of the LSTM, KNN, QDA, and L-SVM classifiers for binary classification is shown in Figure 5a,d,e,f.



(**a**) Confusion Matrix of LSTM Classifier

(**b**) Accuracy vs. Epoch of LSTM Classifier

(**c**) Loss vs. Epoch of LSTM Classifier

(**d**) Confusion Matrix of KNN Classifier

(**e**) Confusion Matrix of L-SVM Classifier

(**f**) Confusion Matrix of QDA Classifier

Figure 5. Illustration of the outcomes achieved by Machine and Deep learning Classifiers.

4.3. Evaluation and Results with IOT23 Dataset

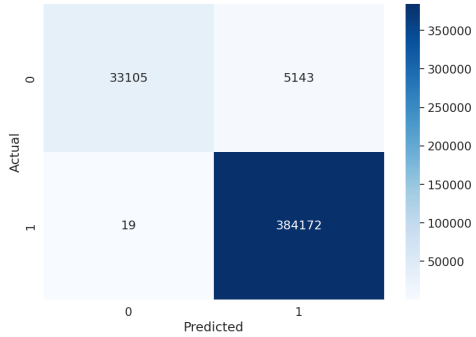
Table 11 presents the binary classification accuracy outcomes. Notably, LSTM excels over alternative classifiers, boasting an impressive 98.7% accuracy. Further, LSVM, AE, and QDA also showcase improved performances on the IOT23 dataset for classification, attributed to various contributing factors. The superior performance of LSTM on the IOT23 dataset is attributed to its ability to capture intricate temporal dependencies and patterns within the IoT network traffic data, thus enabling more accurate predictions in this context.

In Table 11, the results for the binary classification accuracy are highlighted. Remarkably, LSTM stands out among the alternate classifiers, achieving a remarkable accuracy of 98.7%. Moreover, LSVM, AE, and QDA also exhibit enhanced performances when classifying the IOT23 dataset, which can be attributed to multiple underlying factors. The exceptional performance of LSTM on the IOT23 dataset can be attributed to its adeptness in capturing complex temporal relationships and patterns within the network traffic data of IoT devices, leading to more precise predictions in this specific context. The Confusion matrix, Accuracy, and Loss diagram with the LSTM classifier for binary classification are

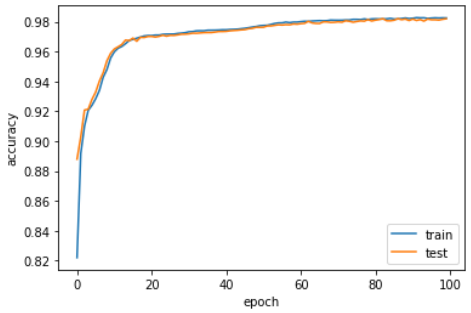
shown in Figure 6a–c. Experimental results demonstrated that the LSTM classifier attained the most acceptable performance for binary classification (99.2%).

Table 11. Accuracy Comparison of Proposed Models for Binary Classification.

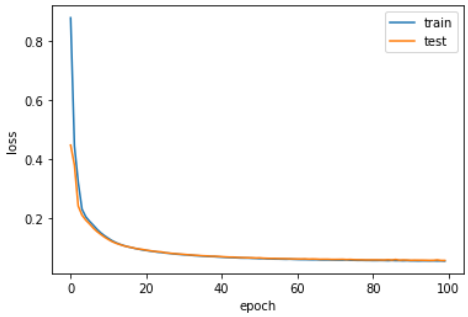
MODELS	Accuracy	Precision	Recall	F1-Score
LSTM	98.7%	98%	98.2%	98%
AE	97.5%	97.5%	97%	98%
MLP	96%	96%	96%	96%
L-SVM	98%	98%	98%	98%
Q-SVM	78%	72%	72%	72%
LDA	96.5%	96.5%	96.5%	96.5%
QDA	97.5%	97%	97%	97%



(**a**) Confusion Matrix of LSTM Classifier



(**b**) Accuracy vs. Epoch of LSTM Classifier



(**c**) Loss vs. Epoch of LSTM Classifier

Figure 6. Illustration of the outcomes achieved by LSTM Classifier on IOT23 dataset.

Table 12 compares the proposed Model's metrics to those of various reference models. The suggested approach outperforms other models in terms of overall performance. The suggested Model exceeds its comparable counterparts by 99%. The percentages are 99%, 99.9%, and 99%, respectively.

Table 12. Performance Comparison of Proposed Model against Existing Models.

Models	Accuracy	Precision	Recall	F1-Score
Hierarchical-IDS [43]	96%	-	-	-
CNN-LSTM [45]	98%	-	-	-
Proposed Model	99.2%	99%	99%	99%

5. Discussion

This study aimed to conduct a statistical investigation and develop a deep learning-based Intrusion Detection System (IDS) technique for detecting network intrusions. To achieve this, we utilized NSL KDD, CICIDS2017, and IOT23. The performances of various machine learning and deep learning classifiers, including L-SVM, Q-SVM, LDA, QDA, MLP, AE, and LSTM, were compared for binary and multi-classification tasks. Our analysis revealed that the proposed approach outperformed the reference models regarding overall performance on all datasets. Specifically, the LSTM classifier achieved the highest accuracy of 99.2% for binary classification with the CICIDS2017 dataset, 98.7% accuracy with the IOT23 dataset, and 97.7% for both binary classification and multi-classification tasks on the NSL KDD dataset. Notably, the LSTM classifier demonstrated superior performance in identifying abnormal samples from the NSL KDD dataset, with a recall of 97%. This can be attributed to the LSTM's ability to model non-linear associations between input features and the target variable. With multiple layers of neurons, the LSTM classifier effectively learns complex patterns in the data. On the other hand, the AE classifier, as an unsupervised learning algorithm, performs dimensionality reduction by encoding input data into a lower-dimensional space. The AE classifier can identify patterns and anomalies by reconstructing the input data from the encoded representation. The excellent performance of the LSTM classifier in recognizing abnormal samples can be attributed to its ability to process input sequences in a time-dependent manner. Since the NSL KDD dataset can be viewed as a sequence of network traffic, the LSTM classifier is designed to capture temporal dependencies and effectively model complex patterns and correlations within the sequence data. The L-SVM classifier also demonstrated a strong performance, achieving an F1 score of 97% in classifying the normal category. Its strength lies in its capability to maximize the margin between positive and negative instances, making it less susceptible to overfitting and suitable for imbalanced datasets.

Conversely, the QDA classifier, which models the distribution of each class separately, exhibited a superior performance in detecting the normal class, achieving a recall of 0.98. QDA's probabilistic nature allows it to estimate the probability density function of each class, making it well-suited for imbalanced datasets where one class significantly outweighs the other. The CIC-IDS2017 dataset, while potentially outdated in representing current network traffic, remains a significant benchmark due to its widespread adoption and the valuable insights it offers. Despite limitations, utilizing this dataset enables a comparison of our proposed method against established approaches on a well-known platform, facilitating meaningful performance evaluations. Acknowledging that network traffic evolves, this dataset still serves as a foundational tool to assess the effectiveness of our approach. Similarly, using the NSL-KDD dataset aims to demonstrate methodological evolution and showcase our approach's adaptability over time rather than implying incorrect assumptions. Including results on the NSL-KDD dataset illustrates the performance progression in varying network security contexts, offering insights into the field's development and newer techniques' advantages on older datasets.

Moreover, our analysis extends to the latest network traffic dataset, IOT23, which contains the most recent network attack captures. Our framework has been applied across all three datasets, ranging from the older NSL-KDD dataset to the more contemporary IOT23, ensuring a comprehensive assessment of its performance across diverse scenarios. Performing experiments on both older and newer datasets serves a twofold purpose. Firstly, it highlights the method's robustness by demonstrating its adaptability to different eras of network security challenges. Secondly, it provides insights into how the approach performs on the latest network attack captures, thus showcasing its relevance and efficacy in addressing the ever-evolving landscape of cybersecurity threats. In conclusion, the findings of this study demonstrate the effectiveness of the proposed deep learning-based IDS technique in detecting network intrusions. The LSTM classifier, in particular, showcases remarkable performance in recognizing abnormal samples, while the L-SVM and QDA classifiers also exhibit strong capabilities in classifying the normal category.

6. Conclusions

This paper proposes a novel approach combining statistical analysis and deep learning techniques for intrusion detection in network security. The model demonstrates significant progress in detecting intrusions in commercial and industrial networks. The effectiveness of the proposed IDS was evaluated using conventional measurement systems. A statistical analysis was utilized to extract highly correlated features, which were fed into deep learning models like AE, LSTM, and traditional machine learning techniques. The experiments were conducted on two datasets: NSL KDD, CICIDS2017, and IOT23, considering binary and multi-classification scenarios. The results showed exceptional accuracy, with a 99% accuracy achieved on the CICIDS2017 dataset, 98.7% accuracy on the IOT23 dataset, and 98% accuracy achieved on the NSL KDD dataset with the LSTM classifier. In future work, one can extend the research by applying deep learning classifiers to detect intrusions in additional datasets available online and in real-time.

Author Contributions: Conceptualization, A.B. and A.R.J.; Methodology, A.B., A.A., G.A.S. and A.R.J.; Software, G.A.S. and A.R.J.; Formal analysis, A.B., A.A. and T.-h.K.; Investigation, A.A. and A.R.J.; Writing—original draft, A.B., G.A.S., A.A., A.R.J. and T.-h.K.; Writing—review & editing, A.B., A.R.J. and G.A.S.; Visualization, A.B., G.A.S. and A.A.; Supervision, G.A.S. and A.R.J.; Project administration, T.-h.K.; Funding acquisition, T.-h.K. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors share no conflict of interest.

References

1. Nasir, M.; Javed, A.R.; Tariq, M.A.; Asim, M.; Baker, T. Feature engineering and deep learning-based intrusion detection framework for securing edge IoT. *J. Supercomput.* **2022**, *78*, 8852–8866. [CrossRef]
2. Zhang, J.; Peng, S.; Gao, Y.; Zhang, Z.; Hong, Q. APMSA: Adversarial Perturbation Against Model Stealing Attacks. *IEEE Trans. Inf. Forensics Secur.* **2023**, *18*, 1667–1679. [CrossRef]
3. Mourad, A.; Tout, H.; Wahab, O.A.; Otrok, H.; Dbouk, T. Ad hoc vehicular fog enabling cooperative low-latency intrusion detection. *IEEE Internet Things J.* **2020**, *8*, 829–843. [CrossRef]
4. Rahman, S.A.; Tout, H.; Talhi, C.; Mourad, A. Internet of things intrusion detection: Centralized, on-device, or federated learning? *IEEE Netw.* **2020**, *34*, 310–317. [CrossRef]
5. Abbas, N.; Nasser, Y.; Shehab, M.; Sharafeddine, S. Attack-specific feature selection for anomaly detection in software-defined networks. In Proceedings of the 2021 3rd IEEE Middle East and North Africa COMMunications Conference (MENACOMM), Agadir, Morocco, 3–5 December 2021; pp. 142–146.
6. Kaddoura, S.; Haraty, R.A.; Al Kontar, K.; Alfandi, O. A parallelized database damage assessment approach after cyberattack for healthcare systems. *Future Internet* **2021**, *13*, 90. [CrossRef]
7. Li, B.; Zhou, X.; Ning, Z.; Guan, X.; Yiu, K.F.C. Dynamic event-triggered security control for networked control systems with cyber-attacks: A model predictive control approach. *Inf. Sci.* **2022**, *612*, 384–398. [CrossRef]
8. Cao, K.; Wang, B.; Ding, H.; Lv, L.; Dong, R.; Cheng, T.; Gong, F. Improving physical layer security of uplink NOMA via energy harvesting jammers. *IEEE Trans. Inf. Forensics Secur.* **2020**, *16*, 786–799. [CrossRef]
9. Wan, Z.; Hazel, J.W.; Clayton, E.W.; Vorobeychik, Y.; Kantarcioglu, M.; Malin, B.A. Sociotechnical safeguards for genomic data privacy. *Nat. Rev. Genet.* **2022**, *23*, 429–445. [CrossRef] [PubMed]
10. Borkar, T.; Heide, F.; Karam, L. Defending against universal attacks through selective feature regeneration. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 709–719.
11. Zhang, X.; Wen, S.; Yan, L.; Feng, J.; Xia, Y. A Hybrid-Convolution Spatial–Temporal Recurrent Network For Traffic Flow Prediction. *Comput. J.* **2022**, bxac171. [CrossRef]
12. Han, Z.; Yang, Y.; Wang, W.; Zhou, L.; Gadekallu, T.R.; Alazab, M.; Gope, P.; Su, C. RSSI map-based trajectory design for UGV against malicious radio source: A reinforcement learning approach. *IEEE Trans. Intell. Transp. Syst.* **2022**, *24*, 4641–4650. [CrossRef]

13. Schufrin, M.; Lücke-Tieke, H.; Kohlhammer, J. Visual Firewall Log Analysis-At the Border Between Analytical and Appealing. In Proceedings of the 2022 IEEE Symposium on Visualization for Cyber Security (VizSec), Oklahoma City, OK, USA, 19 October 2022; pp. 1–11.
14. Xue, B.; Warkentin, M.; Mutchler, L.A.; Balozian, P. Self-efficacy in information security: A replication study. *J. Comput. Inf. Syst.* **2023**, *63*, 1–10. [CrossRef]
15. Yu, J.; Lu, L.; Chen, Y.; Zhu, Y.; Kong, L. An indirect eavesdropping attack of keystrokes on touch screen through acoustic sensing. *IEEE Trans. Mob. Comput.* **2019**, *20*, 337–351. [CrossRef]
16. Alsharif, M.; Mishra, S.; AlShehri, M. Impact of Human Vulnerabilities on Cybersecurity. *Comput. Syst. Sci. Eng.* **2022**, *40*, 1153–1166. [CrossRef]
17. Margossian, H.; Sayed, M.A.; Fawaz, W.; Nakad, Z. Partial grid false data injection attacks against state estimation. *Int. J. Electr. Power Energy Syst.* **2019**, *110*, 623–629. [CrossRef]
18. Wahab, O.A.; Bentahar, J.; Otrok, H.; Mourad, A. Resource-aware detection and defense system against multi-type attacks in the cloud: Repeated bayesian stackelberg game. *IEEE Trans. Dependable Secur. Comput.* **2019**, *18*, 605–622. [CrossRef]
19. Wahab, O.A.; Bentahar, J.; Otrok, H.; Mourad, A. Optimal load distribution for the detection of VM-based DDoS attacks in the cloud. *IEEE Trans. Serv. Comput.* **2017**, *13*, 114–129. [CrossRef]
20. Kavitha, C.; Gadekallu, T.R.; Kavin, B.P.; Lai, W.C. Filter-Based Ensemble Feature Selection and Deep Learning Model for Intrusion Detection in Cloud Computing. *Electronics* **2023**, *12*, 556. [CrossRef]
21. Shaikh, S.; Rupa, C.; Srivastava, G.; Gadekallu, T.R. Botnet Attack Intrusion Detection In IoT Enabled Automated Guided Vehicles. In Proceedings of the 2022 IEEE International Conference on Big Data (Big Data), Osaka, Japan, 17–20 December 2022; pp. 6332–6336.
22. Dbouk, T.; Mourad, A.; Otrok, H.; Tout, H.; Talhi, C. A novel ad-hoc mobile edge cloud offering security services through intelligent resource-aware offloading. *IEEE Trans. Netw. Serv. Manag.* **2019**, *16*, 1665–1680. [CrossRef]
23. Rani, S.; Babbar, H.; Srivastava, G.; Gadekallu, T.R.; Dhiman, G. Security Framework for Internet of Things based Software Defined Networks using Blockchain. *IEEE Internet Things J.* **2022**, *10*, 6074–6081. [CrossRef]
24. Kong, H.; Lu, L.; Yu, J.; Chen, Y.; Tang, F. Continuous authentication through finger gesture interaction for smart homes using WiFi. *IEEE Trans. Mob. Comput.* **2020**, *20*, 3148–3162. [CrossRef]
25. Nagasree, Y.; Rupa, C.; Akshitha, P.; Srivastava, G.; Gadekallu, T.R.; Lakshmanna, K. Preserving privacy of classified authentic satellite lane imagery using proxy re-encryption and UAV technologies. *Drones* **2023**, *7*, 53. [CrossRef]
26. Shamseddine, H.; Nizam, J.; Hammoud, A.; Mourad, A.; Otrok, H.; Harmanani, H.; Dziong, Z. A novel federated fog architecture embedding intelligent formation. *IEEE Netw.* **2020**, *35*, 198–204. [CrossRef]
27. Srivastava, G.; K, D.R.R.; Yenduri, G.; Hegde, P.; Gadekallu, T.R.; Maddikunta, P.K.R.; Bhattacharya, S. Federated Learning Enabled Edge Computing Security for Internet of Medical Things: Concepts, Challenges and Open Issues. In *Security and Risk Analysis for Intelligent Edge Computing*; Springer: Berlin/Heidelberg, Germany, 2023; pp. 67–89.
28. AbdulRahman, S.; Tout, H.; Mourad, A.; Talhi, C. FedMCCS: Multicriteria client selection model for optimal IoT federated learning. *IEEE Internet Things J.* **2020**, *8*, 4723–4735. [CrossRef]
29. AbdulRahman, S.; Tout, H.; Ould-Slimane, H.; Mourad, A.; Talhi, C.; Guizani, M. A survey on federated learning: The journey from centralized to distributed on-site learning and beyond. *IEEE Internet Things J.* **2020**, *8*, 5476–5497. [CrossRef]
30. Wahab, O.A.; Mourad, A.; Otrok, H.; Taleb, T. Federated machine learning: Survey, multi-level classification, desirable criteria and future directions in communication and networking systems. *IEEE Commun. Surv. Tutor.* **2021**, *23*, 1342–1397. [CrossRef]
31. Sarkar, S.; Agrawal, S.; Gadekallu, T.R.; Mahmud, M.; Brown, D.J. Privacy-Preserving Federated Learning for Pneumonia Diagnosis. In *International Conference on Neural Information Processing, Proceedings of the 29th International Conference, ICONIP 2022, Virtual Event, 22–26 November 2022*; Proceedings, Part VII; Springer: Berlin/Heidelberg, Germany, 2023; pp. 345–356.
32. Hammoud, A.; Otrok, H.; Mourad, A.; Dziong, Z. Stable federated fog formation: An evolutionary game theoretical approach. *Future Gener. Comput. Syst.* **2021**, *124*, 21–32. [CrossRef]
33. Hammoud, A.; Otrok, H.; Mourad, A.; Dziong, Z. On demand fog federations for horizontal federated learning in IoV. *IEEE Trans. Netw. Serv. Manag.* **2022**, *19*, 3062–3075. [CrossRef]
34. Fawaz, W. Effect of non-cooperative vehicles on path connectivity in vehicular networks: A theoretical analysis and UAV-based remedy. *Veh. Commun.* **2018**, *11*, 12–19. [CrossRef]
35. Pandya, S.; Srivastava, G.; Jhaveri, R.; Babu, M.R.; Bhattacharya, S.; Maddikunta, P.K.R.; Mastorakis, S.; Piran, M.J.; Gadekallu, T.R. Federated learning for smart cities: A comprehensive survey. *Sustain. Energy Technol. Assess.* **2023**, *55*, 102987. [CrossRef]
36. Gupta, P.; Yadav, K.; Gupta, B.; Alazab, M.; Gadekallu, T.R. A Novel Data Poisoning Attack in Federated Learning based on Inverted Loss Function. *Comput. Secur.* **2023**, *130*, 103270. [CrossRef]
37. Arafeh, M.; Ceravolo, P.; Mourad, A.; Damiani, E.; Bellini, E. Ontology based recommender system using social network data. *Future Gener. Comput. Syst.* **2021**, *115*, 769–779. [CrossRef] [PubMed]
38. Sharma, R.K.; Issac, B.; Xin, Q.; Gadekallu, T.R.; Nath, K. Plant and Salamander Inspired Network Attack Detection and Data Recovery Model. *Sensors* **2023**, *23*, 5562. [CrossRef] [PubMed]
39. Sakhnini, J.; Karimipour, H.; Dehghantanha, A.; Yazdinejad, A.; Gadekallu, T.R.; Victor, N.; Islam, A. A Generalizable Deep Neural Network Method for Detecting Attacks in Industrial Cyber-Physical Systems. *IEEE Syst. J.* **2023**. [CrossRef]

40. Afzal, S.; Asim, M.; Javed, A.R.; Beg, M.O.; Baker, T. Urldeepdetect: A deep learning approach for detecting malicious urls using semantic vector models. *J. Netw. Syst. Manag.* **2021**, *29*, 21. [CrossRef]
41. Tao, P.; Sun, Z.; Sun, Z. An improved intrusion detection algorithm based on GA and SVM. *IEEE Access* **2018**, *6*, 13624–13631. [CrossRef]
42. Tang, T.A.; Mhamdi, L.; McLernon, D.; Zaidi, S.A.R.; Ghogho, M. Deep learning approach for network intrusion detection in software defined networking. In Proceedings of the 2016 International Conference on Wireless Networks and Mobile Communications (WINCOM), Fez, Morocco, 26–29 October 2016; pp. 258–263.
43. Ahmim, A.; Maglaras, L.; Ferrag, M.A.; Derdour, M.; Janicke, H. A novel hierarchical intrusion detection system based on decision tree and rules-based models. In Proceedings of the 2019 15th International Conference on Distributed Computing in Sensor Systems (DCOSS), Santorini Island, Greece, 29–31 May 2019;pp. 228–233.
44. Faker, O.; Dogdu, E. Intrusion detection using big data and deep learning techniques. In Proceedings of the 2019 ACM Southeast Conference, Kennesaw, GA, USA, 18–20 April 2019; pp. 86–93.
45. Sun, P.; Liu, P.; Li, Q.; Liu, C.; Lu, X.; Hao, R.; Chen, J. DL-IDS: Extracting features using CNN-LSTM hybrid network for intrusion detection system. *Secur. Commun. Netw.* **2020**, *2020*, 8890306. [CrossRef]
46. Jiang, K.; Wang, W.; Wang, A.; Wu, H. Network intrusion detection combined hybrid sampling with deep hierarchical network. *IEEE Access* **2020**, *8*, 32464–32476. [CrossRef]
47. Eskandari, M.; Janjua, Z.H.; Vecchio, M.; Antonelli, F. Passban IDS: An intelligent anomaly-based intrusion detection system for IoT edge devices. *IEEE Internet Things J.* **2020**, *7*, 6882–6897. [CrossRef]
48. Kherraf, N.; Sharafeddine, S.; Assi, C.M.; Ghrayeb, A. Latency and reliability-aware workload assignment in IoT networks with mobile edge clouds. *IEEE Trans. Netw. Serv. Manag.* **2019**, *16*, 1435–1449. [CrossRef]
49. Wang, B.; Su, Y.; Zhang, M.; Nie, J. A deep hierarchical network for packet-level malicious traffic detection. *IEEE Access* **2020**, *8*, 201728–201740. [CrossRef]
50. Shaukat, K.; Alam, T.M.; Luo, S.; Shabbir, S.; Hameed, I.A.; Li, J.; Abbas, S.K.; Javed, U. A review of time-series anomaly detection techniques: A step to future perspectives. In *Advances in Information and Communication: Proceedings of the 2021 Future of Information and Communication Conference (FICC), Vancouver, BC, Canada, 29–30 April 2021*; Springer: Berlin/Heidelberg, Germany, 2021; Volume 1, pp. 865–877.
51. Kalaria, R.; Kayes, A.; Rahayu, W.; Pardede, E. A Secure Mutual authentication approach to fog computing environment. *Comput. Secur.* **2021**, *111*, 102483. [CrossRef]
52. Xu, C.; Shen, J.; Du, X.; Zhang, F. An intrusion detection system using a deep neural network with gated recurrent units. *IEEE Access* **2018**, *6*, 48697–48707. [CrossRef]
53. Yan, B.; Han, G. Effective feature extraction via stacked sparse autoencoder to improve intrusion detection system. *IEEE Access* **2018**, *6*, 41238–41248. [CrossRef]
54. Yin, C.; Zhu, Y.; Fei, J.; He, X. A deep learning approach for intrusion detection using recurrent neural networks. *IEEE Access* **2017**, *5*, 21954–21961. [CrossRef]
55. Shapoorifard, H.; Shamsinejad, P. Intrusion detection using a novel hybrid method incorporating an improved KNN. *Int. J. Comput. Appl.* **2017**, *173*, 5–9. [CrossRef]
56. Ren, J.; Liu, X.; Wang, Q.; He, H.; Zhao, X. An multi-level intrusion detection method based on KNN outlier detection and random forests. *J. Comput. Res. Dev.* **2019**, *56*, 566–575.
57. Su, T.; Sun, H.; Zhu, J.; Wang, S.; Li, Y. BAT: Deep learning methods on network intrusion detection using NSL-KDD dataset. *IEEE Access* **2020**, *8*, 29575–29585. [CrossRef]
58. Cao, B.; Li, C.; Song, Y.; Qin, Y.; Chen, C. Network Intrusion Detection Model Based on CNN and GRU. *Appl. Sci.* **2022**, *12*, 4184. [CrossRef]
59. Qazi, E.U.H.; Faheem, M.H.; Zia, T. HDLNIDS: Hybrid Deep-Learning-Based Network Intrusion Detection System. *Appl. Sci.* **2023**, *13*, 4921. [CrossRef]
60. Mhawi, D.N.; Aldallal, A.; Hassan, S. Advanced feature-selection-based hybrid ensemble learning algorithms for network intrusion detection systems. *Symmetry* **2022**, *14*, 1461. [CrossRef]
61. Tavallaee, M.; Bagheri, E.; Lu, W.; Ghorbani, A.A. A Detailed Analysis of the KDD CUP 99 Data Set. In Proceedings of the Second IEEE Symposium on Computational Intelligence for Security and Defense Applications (CISDA), Ottawa, ON, Canada, 8–10 July 2009.
62. Garcia, S.; Parmisano, A.; Erquiaga, M.J. *IoT-23: A Labeled Dataset with Malicious and Benign IoT Network Traffic*; Technical Reports; Stratosphere Lab.: Praha, Czech Republic, 2020.
63. Hastie, T.; Tibshirani, R.; Friedman, J.H.; Friedman, J.H. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*; Springer: Berlin/Heidelberg, Germany, 2009; Volume 2.
64. Chicco, D. Siamese neural networks: An overview. In *Artificial Neural Networks*; Springer: Berlin/Heidelberg, Germany, 2021; pp. 73–94.
65. Meng, Q.; Catchpoole, D.; Skillicom, D.; Kennedy, P.J. Relational autoencoder for feature extraction. In Proceedings of the 2017 International Joint Conference on Neural Networks (IJCNN), Anchorage, AK, USA, 14–19 May 2017; pp. 364–371.
66. Sublime, J.; Kalinicheva, E. Automatic post-disaster damage mapping using deep-learning techniques for change detection: Case study of the Tohoku tsunami. *Remote Sens.* **2019**, *11*, 1123. [CrossRef]

67. Greff, K.; Srivastava, R.K.; Koutník, J.; Steunebrink, B.R.; Schmidhuber, J. LSTM: A search space odyssey. *IEEE Trans. Neural Netw. Learn. Syst.* **2016**, *28*, 2222–2232. [CrossRef] [PubMed]
68. Gui, Z.; Sun, Y.; Yang, L.; Peng, D.; Li, F.; Wu, H.; Guo, C.; Guo, W.; Gong, J. LSI-LSTM: An attention-aware LSTM for real-time driving destination prediction by considering location semantics and location importance of trajectory points. *Neurocomputing* **2021**, *440*, 72–88. [CrossRef]
69. Pervez, M.S.; Farid, D.M. Feature selection and intrusion classification in NSL-KDD cup 99 dataset employing SVMs. In Proceedings of the 8th International Conference on Software, Knowledge, Information Management and Applications (SKIMA 2014), Dhaka, Bangladesh, 18–20 December 2014; pp. 1–6.
70. Ding, Y.; Zhai, Y. Intrusion detection system for NSL-KDD dataset using convolutional neural networks. In Proceedings of the 2018 2nd International Conference on Computer Science and Artificial Intelligence, Shenzhen, China, 8–10 December 2018; pp. 81–85.
71. Tama, B.A.; Comuzzi, M.; Rhee, K.H. TSE-IDS: A two-stage classifier ensemble for intelligent anomaly-based intrusion detection system. *IEEE Access* **2019**, *7*, 94497–94507. [CrossRef]
72. Ieracitano, C.; Adeel, A.; Morabito, F.C.; Hussain, A. A novel statistical analysis and autoencoder driven intelligent intrusion detection approach. *Neurocomputing* **2020**, *387*, 51–62. [CrossRef]
73. Fu, Y.; Du, Y.; Cao, Z.; Li, Q.; Xiang, W. A Deep Learning Model for Network Intrusion Detection with Imbalanced Data. *Electronics* **2022**, *11*, 898. [CrossRef]