Network Intrusion Detection Model based on Deep Learning Techniques

*

Aysha Bibi
Department of Cyber Security
Air University
Islamabad, Pakistan

Hidayat ur Rehman
Department of Cyber Security
Air University
Islamabad, Pakistan

Dr Naveed Bhatti
Department of Cyber Security
Air University
Islamabad, Pakistan

***Abstract*—Several network attacks increase yearly, and standard firewalls and data encryption solutions cannot constantly meet the needs of existing network safety measures. As a result, network intrusion detection technologies have been created. Detecting and managing these dangers with typical analytic methods is extremely difficult. Although it has lower detection rates and requires extensive feature engineering, machine learning is used to augment conventional intrusion detection algorithms. This research proposes a Deep Learning Model for network intrusion detection to solve the issue of low detection accuracy. The suggested IDS blends data analytics, statistical techniques, and recent advancements in Machine learning theory to get optimal and further associated qualities. A benchmarking database known as NSL-KDD is used to verify the validity of the proposed IDS. When evaluating to remaining machine learning classifiers, the proposed Intrusion Detection System gets the most remarkable accuracy of 97.7% when employing the LSTM and MLP classifiers.**

***Index Terms*—Deep learning, NSL-KDD, Long Short-Term Memory, Multi-Layer Perceptron, Network Intrusion Detection**

I. Introduction

Due to the expeditious growth in computer technology, individuals all over the globe are taking more internet services than ever. Furthermore, the diversity of cyberattacks has also increased due to the increase in internet services. For example, network worms, malevolent spying, and aggressive assaults seriously threaten people's data security and physical safety. As a result, data security and security protocols have become critical for both people and society [1].

Firewalls are extensively used and frequently installed as a fundamental security measure. However, it no more remains appropriate and requires strong security (e.g., governmental entities, military assets, etc.) [2] 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 behaviour through intrusion detection systems (IDSs) to combat these threats [3]. Annually, breaches in IT networks cost trillions of dollars, and this amount is predictable to climb ahead [4]. As a response, cybersecurity has been a primary focus in recent years. To this aim, network traffic data monitoring and analysis are critical for detecting possible attack trends. 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.

A. Motivation

A collection of approaches from computers, statistics, and information and technology, such as Machine Learning, are included in this research. Due to the huge number of massive heterogeneous data generated by numerous sources, traditional data analytics and machine learning approaches are worthless and inefficient in dealing with such security concerns. Furthermore, traditional machine learning approaches have a limited processing complexity and cannot discover complicated nonlinear relationships in large datasets. As a result, to overcome the abovementioned restrictions and thereby improve intrusion detection performance, we integrate classic data analysis and statistical approaches with current breakthroughs in Machine learning. Deep Learning technologies are mainly used to create a more advanced security IDS [5]. 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. The data preparation program eliminates outliers, scales characteristics from 0 to 1, and uses one hot-encoding approach [6] to change categorical attributes to numerical ones. Other associated variables are extracted in the feature extraction module. Conversely, an autoencoder deep classifier is presented in the classification algorithm to distinguish distinct dataset groups. Binary and multi-classification are two types of classification employed. The binary class has two labels: normal and abnormal, whereas the multi-classification includes five labels: DoS (Denial of service), Normal, Probe (Probing), R2L (Root to Local), and U2R (User to Root). DoS attacks have included 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 accessibility of services.R2L attacks have included that allow unauthorised local access to a device by delivering deceptive information to the host.U2R attacks include those that grant root privileges. In this example, the attacker discovers the internal weaknesses and begins utilising the device as a regular user. AE (AutoEncoder), Long-short-term-memory (LSTM), and some other machine learning classifiers such as MLP (Multi Layer Perceptron), L-SVM, Q-SVM, LDA, and QDA are employed.

B. Contribution

The major contributions of this research include:

- Development of a new intrusion detection system based on data analytics and Deep Learning technologies
- Development of an IDS capable of accurately distinguishing diverse cyber-attack types in the NLS-KDD dataset.
- Creation of an IDS with substantial potential for use in industrial applications

The following is the rest of the paper: Section II goes over the Literature Review, Section III goes over the Proposed Approach, Section IV goes over the Experimental Settings, and Section V goes over the Results and Discussion. Section VI concluded the paper with a conclusion and a few suggestions for further research. Finally, the references for this work are provided.

II. Literature Review

The NSL KDD dataset has remained used extensively in publications to estimate the execution and usefulness of intrusion detection simulations. Nehla et al. [13], proposed a machine learning classifier, Naïve Bayes, to detect abnormal networks and, in the end, examine them with the decision tree. The authors of one study [8] used SVM and a genetic algorithm to adjust the correctness of the Model by tweaking the SVM attributes' selection, variables, and weights. Authors in [14] proposed a multi-layer-perceptron for the discovery attack stack in the dataset. The classifications' accuracy is 81% and 79%, respectively. Yadav et al. [15], used a multi-layer perceptron and convolutional neural network to detect abnormality with an accuracy of 95%. Y.jin et al. [16], authors used an ensemble learning theory with a semi-supervised method on the NSL-KDD dataset with an accuracy of 84% on the assessment data. Shapoorifard et al. [7], update the classifiers by integrating them to boost detection accuracy. Lee et al. [17], developed a unique intrusion detection approach that uses a C4.5 decision tree algorithm to partition network data into subgroups before creating various SVM models for the subsections, reducing complexity and improving the detection performance of unspecified assaults. On the other hand, existing machine learning approaches focus on feature engineering, which takes a lot of computer power and only learns shallow features, resulting in less accurate identification. Many academics have shifted their attention to the actual deep learning approach to bypass the attribute selection stage by simply incorporating network traffic data into the system. One study [18] proposes a structural model based on deep neural. Tang et al. [19], the authors introduced a deep neural network to detect intrusion and software to define network context. The NSL KDD dataset was used to train a tri layers neural network. Only six characteristics were employed, and only two-way differentiation was used. The test findings showed a 75% accuracy rate. Table I presents the previous relevant research summary.

Kim et al. [20], used the KDD99 dataset to prepare a DNN. The DNN used the reweighted estimation (Adam) approach to trait four hidden layers and hundred hidden neurons. Researchers claim to have achieved excellent results, yet they only used sections of the entire KDD99 dataset. Xu et al. [21], authors 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. [22] suggested a small autoencoder. The researchers stated a 98% accuracy rate, although they sped up the experiment by scrambling and reconstructing the essential information into numerous separate datasets. Hantao et al. [23], have developed a stacked semi-deep autoencoder design for detecting cyber assaults. Yin et al. [24], 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 multi-classification accuracy of 85%. A recent publication proposes a co-model machine learning acceleration based on a sequence learning procedure that achieved 76.04% accuracy and a training duration of 144.5 seconds. Furthermore, Ahmad et al.in [25], offered a sparse AE architecture with a 79% accuracy rate. In contrast, Yadigar et al. [26], the researchers constructed a Gaussian RBM with seven layers of hundred neurons and 73.23% overall accuracy. [27] created a layered non-symmetric shallow AE. [24] reported an accuracy of 81% for an RNN with 80 hidden elements. Furthermore, network traffic data has an asymmetric distribution, and neither of the existing networks takes advantage of the correlation among traffic aspects. A deep learning model is provided in this research to handle the existing challenges, as well as an autoencoder for data dimensionality reduction. We use the NSL KDD dataset for simulation testing to train and assess the proposed Model's performance.

III. Proposed Approach

Figure 1 shows the proposed architectural design of the system. First, upload the NSL-KDD dataset and then start preprocessing the data by data-normalization using the standard scalar within the range of 0 and 1. Then, use one-hot encoding to transform category characteristics into numerical features. Finally, the detection operation is evaluated using the classifiers mentioned above.

TABLE I
EXISTING RESEARCH WORK SUMMARY

AUTHOR	YEAR	ALGORITHM	CONTRIBUTION
Shamsinejad et al. [7]	2017	k-Means	To enhance detection accuracy, a K-MEANS clustering classifier was proposed.
Sun et al. [8]	2018	SVM and Genetic Algorithm	SVM features are optimized using a genetic algorithm. It enables selection parameters and weights to be optimized.
Xinqian et al. [9]	2019	Random Forest	Detected abnormal network behavior using a multilevel random forest model.
Catania et al. [10]	2016	RNN	An RNN model is proposed to detect Botnet anomalies.
Zeng et al. [11]	2017	CNN	The data generated by the network traffic is projected into pictures by CNN.
Wang et al. [12]	2020	CNN and LSTM	To detect each attack type, a model based on CNN and LSTM is proposed.

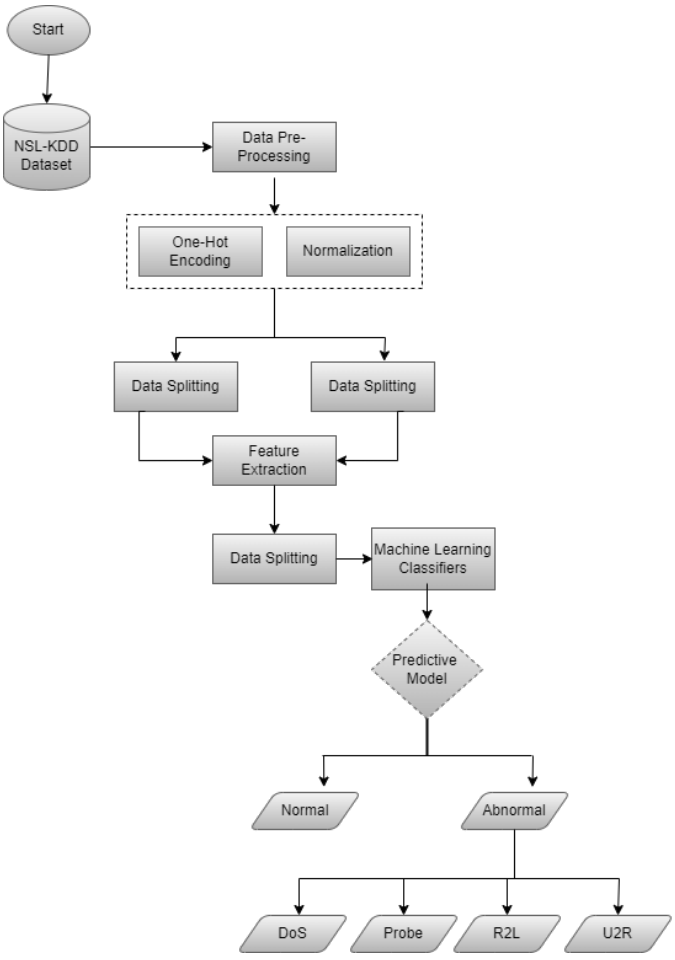

Fig. 1. Framework of the proposed Model

A. Dataset

The NSL KDD dataset is one of the common dataset used to assess the functioning of 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 fair 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 as follows: Normal, Denial of service (DOS), R2L, U2R, and probe.

B. Preprocessing

NSL-KDD dataset contains outliers or inconsistent values, and for building a model preprocessing of data is necessary. Our work comprises two components: Normalization of the data and one-hot-encoding.

1) Normalizaton: The numeric feature values were mapped into the numeric range 0 and 1 using the conventional scalar normalizing approach. A sample's standard score is computed as arises (2):

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

where s=(s1,...,sn) and Z is the ith normalised data point.

2) One Hot Encoding: Three categorical elements are present in the dataset (service, flag, and protocol type). Using the one-hot-encoding approach, we converted these category data into numerical features. The z2 feature has three properties. One hot encoding approach was used to convert them to binary data: [1,0,0], [0,1,0],[0,0,1], respectively. Also transformed into one-hot-encoding matrices were z3 and z4 attributes (service and flag).

C. Extraction of Features

The feature extraction process obtains the most correlated elements from the dataset. For the feature extraction, we used the Pearson correlation matrix [28]. The correlation coefficient has a range of values from -1 to 1.

D. Classification

The dataset that we utilized in our work is the NSL KDD dataset containing two class categories, regular class labels and abnormal classes. We have used two deep learning

classifiers, such as LSTM and Autoencoder (AE), and three other conventional for the five class labels Denial of service (DOS), Normal, R2L, and U2R.

1) Autoencoder: In an autoencoder, the input and output dimensions are identical. It is an unsupervised learning network [29]. It contains two modules, the first one is the encoder, and the other is the decoder module. Autoencoder 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 autoencoder that can conduct dimensionality reduction and boost data resilience to familiarize with complicated network situations, which can accomplish better data dimensionality reduction than previous dimensionality reduction approaches. Autoencoder architecture is given in the figure 2.

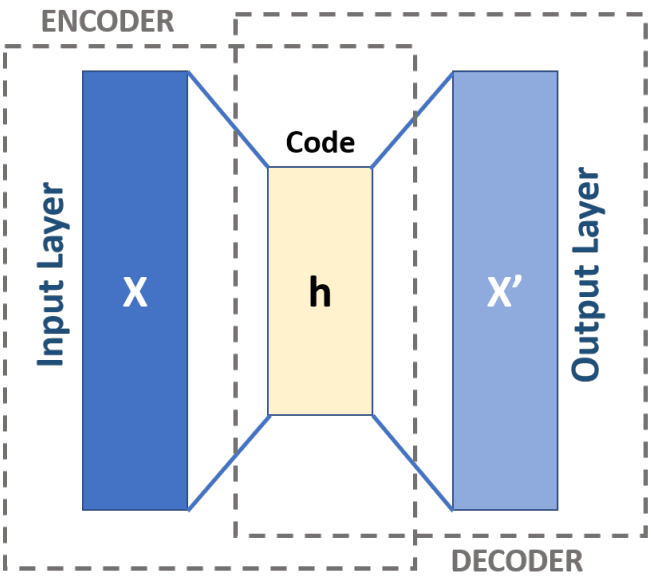

Fig. 2. The architecture of Autoencoder

2) LSTM: Long short-term memory [30], [31] provides storage cells and cell states to solve the recurrent neural network's long-term reliance problem (RNNs). LSTM systems are recurrent neural networks that deal with instances where RNNs failed. 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. 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.

3) DA Classifier: 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.

4) SVM Classifier: 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.

5) MLP Classifier: It is a feedforward 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 a single hidden layer, Fifty neurons, and an activation layer for classification tasks.

E. Binary Classification

In binary classification, we have changed the attack labels into ' NORMAL' and 'ABNORMAL.' First, create the data frame with binary labels' NORMAL' and 'ABNORMAL' and then encode the labels into 0 and 1. Pie-chart for the binary classification is shown in figure 3. As shown in the figure, we have 53% standard data and 47% abnormal labels in the NSL KDD dataset in the case of binary classification. In abnormal labels, we have four types of attacks.

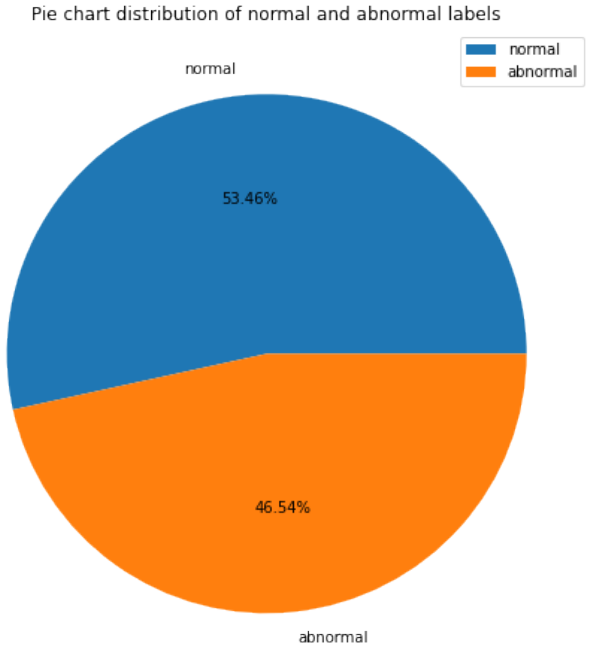

Fig. 3. Pie Chart Distribution of Binary Class Labels

F. Multi Classification

In multi-classification, we have 5 class tags: Normal, Denial of service (DoS), Probe, R2L, and 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. Pie-chart for the multi-class classification is shown in figure 4. As shown in the figure, 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.

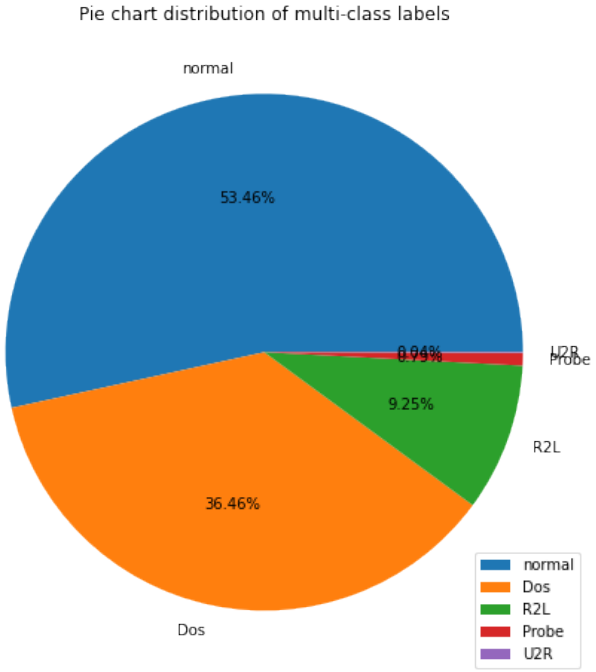

Fig. 4. Pie Chart Distribution of Multi-Class Labels

IV. Experimental Settings

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

$$Recall = TP/(TP + FN) \quad (2)$$

$$Precision = TP/(TP + FP) \quad (3)$$

$$Accuracy = (TP + TN)/(TP + FP + TN + FN) \quad (4)$$

where TP (True Positive) is the amount of abnormal cases accurately identified; The number of cases accurately identified as normal is denoted by TN (True Negative). The amount of normal traffic patterns misclassified as abnormal is referred to as FP; the amount of anomalous traffic patterns misclassified as normal is referred to as FN. Two types of classification are used in this work, such as binary classification and multiclassification, to check the presence 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.

V. Result and Discussion

Table II shows the accuracy of binary and multi-classification. For binary classification, 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%.

TABLE II
Proposed Models Accuracies for Binary & 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%

The results of binary classification studies are shown in Table III, where the abnormal class contains Denial of service (DoS), R2L, U2R, and Probe types. The LSTM and MLP classifiers detected the normal category with great precision, whereas the Q-SVM and DQA 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. The Confusion matrix of the LSTM classifier for binary classification is shown in figure 5. The Accuracy vs. Epoch and the Loss vs. Epoch of the LSTM classifier are shown in figure 6 and figure 7.

TABLE III
Proposed Model's Precision for 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

With a recall of 0.99, L-SVM and QDA exceed all other classifiers regarding the Recall. Q-SVM beat L-SVM in detecting the normal class when using SVM classifiers. Regarding discriminant analysis, QDA surpassed LDA in terms of Normal Sample Recall (0.99). The LSTM classifier better recognized anomalous classes (Recall of 97%) as shown in TABLE IV.

TABLE IV
Proposed Model's Recall for 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 LSVM f1-score is higher (97%). Regarding

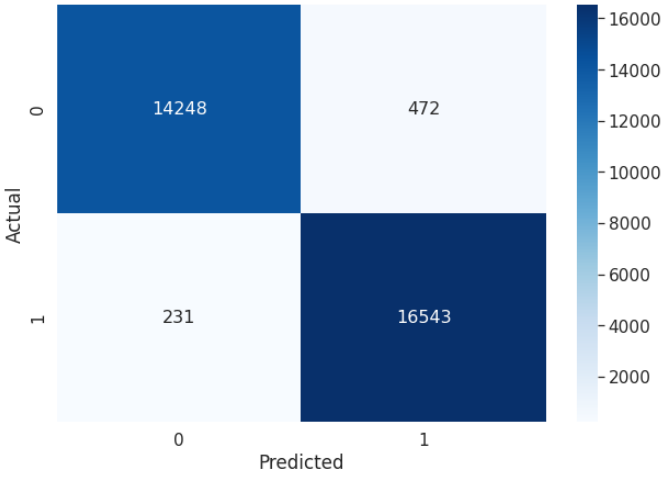

Fig. 5. Confusion Matrix of LSTM for Binary Classification

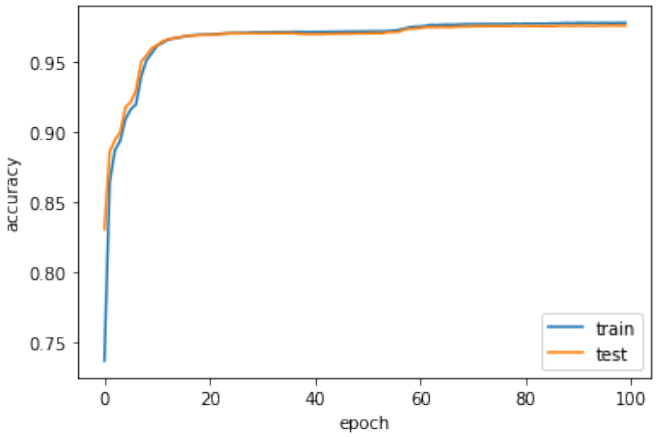

Fig. 6. Accuracy vs. Epochs of LSTM Classifier for Binary Classification

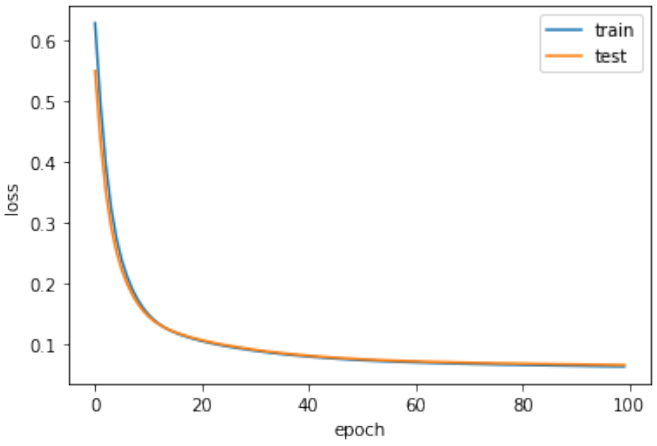

Fig. 7. Loss vs. Epochs of LSTM Classifier for Binary Classification

detecting standard samples, the LDA classifier outperformed the QDA. With a score of 99%, LSTM surpassed all other classifiers in recognizing abnormal samples shown in TABLE V.

TABLE V
PROPOSED MODEL'S F1-SCORE 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 Linear Support Vector Machine (L-SVM), Q-SVM, LDA, QDA, and MLP classifiers were compared similarly to the binary classification study. Table VI shows that Autoencoder and MLP classifier outperformed in the case of precision in all other classifiers. The confusion matrix, Accuracy vs. Epoch, and Loss vs. Epoch of MLP classifier are given in figure 8, figure 9 and figure 10.

Recall and f1-Score AE outperformed in all other classifiers as shown in table VII & Table VIII.

TABLE VI
PROPOSED MODEL'S PRECISION FOR MULTI 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. Statistical analysis was used to extract the most correlated features, which were then fed into deep (AE, LSTM) and deep ML techniques and Multi-layer-perceptron (MLP), L-SVM, Q-SVM, LDA, and QDA.

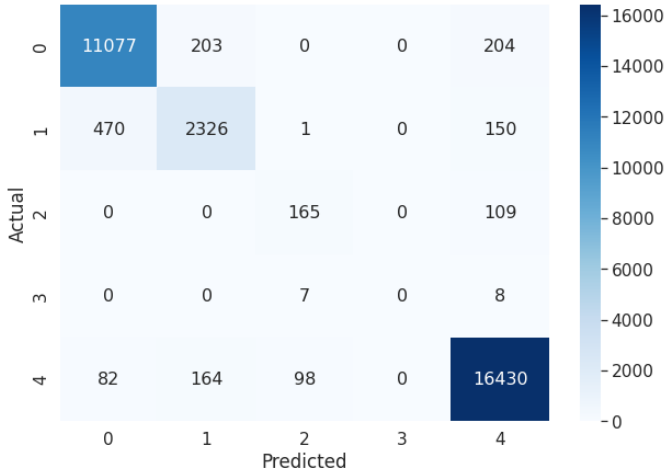

Fig. 8. Confusion Matrix of MLP for Multi classification

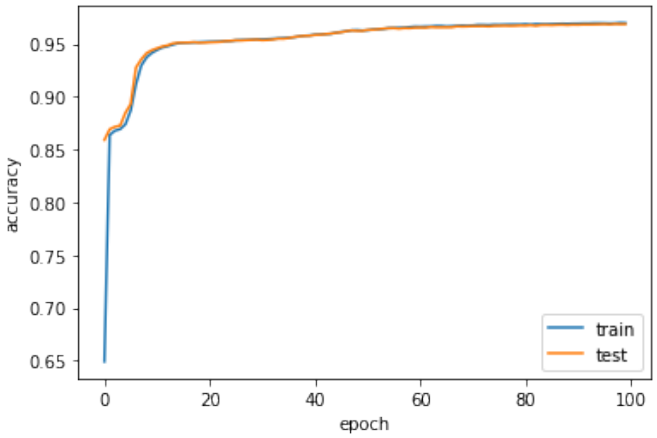

Fig. 9. Accuracy vs. Epochs of MLP Classifier

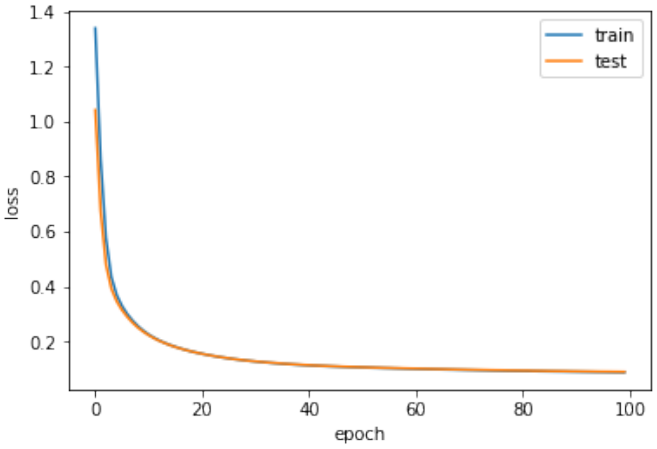

Fig. 10. =Loss vs. Epochs of MLP Classifier

TABLE VII
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 VIII
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

Moreover, 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. AE classifier also accomplished high accuracy of 98% compared to the LSTM classifier.

TABLE IX
OVERALL PERFORMANCE COMPARISON OF PROPOSED MODEL WITH EXISTING MODELS

Models	Accuracy	Precision	Recall	F1-score
SVM-IDS [32]	82%	-	-	-
CNN [33]	80%	-	-	-
TES-IDS [34]	85%	88%	86%	85%
Autoencoder [35]	84%	87%	80%	81%
CNN & BiLSTM [36]	83%	85%	84%	85%
DLNID [37]	90%	86%	93%	89%
Proposed Model	97.7%	97%	98%	97%

In Table IX, we equate our proposed Model's different 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, shown in the table.

VI. CONCLUSION

We introduced a new statistical analysis and deep learning-dependent IDS technique in this analysis. As a reference, the suggested Model is examined using the NSL-KDD dataset. The conventional measurement system was used to assess the proposed IDS' capabilities and efficacy. Statistical analysis was used to extract the most correlated features, which would then be fed into deep (Autoencoder, LSTM) and deep ML techniques. Furthermore, both binary and multi-classification were carried out. When equated to existing machine learning classifiers, the proposed Intrusion Detection System gets the most remarkable accuracy of 97.7% when employing the LSTM and MLP classifiers. We will employ deep learning classifiers in the future to detect intrusions on additional datasets that are available online, as well as real-time data.

REFERENCES

[1] Sam Roweis. Em algorithms for pca and spca. *Advances in neural information processing systems*, 10, 1997.

[2] Lihua Yuan, Hao Chen, Jianning Mai, Chen-Nee Chuah, Zhendong Su, and Prasant Mohapatra. Fireman: A toolkit for firewall modeling and analysis. In *2006 IEEE Symposium on Security and Privacy (S&P'06)*, pages 15–pp. IEEE, 2006.

[3] Muneeba Nasir, Abdul Rehman Javed, Muhammad Adnan Tariq, Muhammad Asim, and Thar Baker. Feature engineering and deep learning-based intrusion detection framework for securing edge iot. *The Journal of Supercomputing*, 78(6):8852–8866, 2022.

[4] Sanjay Goel, Kevin Williams, and Ersin Dincelli. Got phished? internet security and human vulnerability. *Journal of the Association for Information Systems*, 18(1):2, 2017.

[5] Sara Afzal, Muhammad Asim, Abdul Rehman Javed, Mirza Omer Beg, and Thar Baker. Urldeepdetect: A deep learning approach for detecting malicious urls using semantic vector models. *Journal of Network and Systems Management*, 29(3):1–27, 2021.

[6] Sparsh Mittal and Subhrajit Nag. A survey of encoding techniques for reducing data-movement energy. *Journal of Systems Architecture*, 97:373–396, 2019.

[7] Hossein Shapoorifard and Pirooz Shamsinejad. Intrusion detection using a novel hybrid method incorporating an improved knn. *Int. J. Comput. Appl*, 173(1):5–9, 2017.

[8] Peiying Tao, Zhe Sun, and Zhixin Sun. An improved intrusion detection algorithm based on ga and svm. *Ieee Access*, 6:13624–13631, 2018.

[9] JD Ren, XQ Liu, Qian Wang, Haitao He, and Xiaolin Zhao. An multi-level intrusion detection method based on knn outlier detection and random forests. *Journal of Computer Research and Development*, 56(3):566–575, 2019.

[10] Pablo Torres, Carlos Catania, Sebastian Garcia, and Carlos Garcia Garino. An analysis of recurrent neural networks for botnet detection behavior. In *2016 IEEE biennial congress of Argentina (ARGENCON)*, pages 1–6. IEEE, 2016.

[11] Wei Wang, Ming Zhu, Xuewen Zeng, Xiaozhou Ye, and Yiqiang Sheng. Malware traffic classification using convolutional neural network for representation learning. In *2017 International conference on information networking (ICOIN)*, pages 712–717. IEEE, 2017.

[12] Tongtong Su, Huazhi Sun, Jinqi Zhu, Sheng Wang, and Yabo Li. Bat: Deep learning methods on network intrusion detection using nsl-kdd dataset. *IEEE Access*, 8:29575–29585, 2020.

[13] Nahla Ben Amor, Salem Benferhat, and Zied Elouedi. Naive bayes vs decision trees in intrusion detection systems. In *Proceedings of the 2004 ACM symposium on Applied computing*, pages 420–424, 2004.

[14] Bhupendra Ingre and Anamika Yadav. Performance analysis of nsl-kdd dataset using ann. In *2015 international conference on signal processing and communication engineering systems*, pages 92–96. IEEE, 2015.

[15] Hany Mohamed, Hesham Hefny, and Assem Alsawy. Intrusion detection system using machine learning approaches. *Egypt. Comput. Sci. J*, 42(3):1–13, 2018.

[16] Ying Gao, Yu Liu, Yaqia Jin, Juequan Chen, and Hongrui Wu. A novel semi-supervised learning approach for network intrusion detection on cloud-based robotic system. *IEEE Access*, 6:50927–50938, 2018.

[17] Gisung Kim, Seungmin Lee, and Sehun Kim. A novel hybrid intrusion detection method integrating anomaly detection with misuse detection. *Expert Systems with Applications*, 41(4):1690–1700, 2014.

[18] Guangzhen Zhao, Cuixiao Zhang, and Lijuan Zheng. Intrusion detection using deep belief network and probabilistic neural network. In *2017 IEEE international conference on computational science and engineering (CSE) and IEEE international conference on embedded and ubiquitous computing (EUC)*, volume 1, pages 639–642. IEEE, 2017.

[19] Tuan A Tang, Lotfi Mhamdi, Des McLernon, Syed Ali Raza Zaidi, and Mounir Ghogho. Deep learning approach for network intrusion detection in software defined networking. In *2016 international conference on wireless networks and mobile communications (WINCOM)*, pages 258–263. IEEE, 2016.

[20] Jin Kim, Nara Shin, Seung Yeon Jo, and Sang Hyun Kim. Method of intrusion detection using deep neural network. In *2017 IEEE international conference on big data and smart computing (BigComp)*, pages 313–316. IEEE, 2017.

[21] Congyuan Xu, Jizhong Shen, Xin Du, and Fan Zhang. An intrusion detection system using a deep neural network with gated recurrent units. *IEEE Access*, 6:48697–48707, 2018.

[22] Binghao Yan and Guodong Han. Effective feature extraction via stacked sparse autoencoder to improve intrusion detection system. *IEEE Access*, 6:41238–41248, 2018.

[23] Hantao Huang, Rai Suleman Khalid, Wenye Liu, and Hao Yu. Work-in-progress: a fast online sequential learning accelerator for iot network intrusion detection. In *2017 International Conference on Hardware/Software Codesign and System Synthesis (CODES+ ISSS)*, pages 1–2. IEEE, 2017.

[24] Chuanlong Yin, Yuefei Zhu, Jinlong Fei, and Xinzheng He. A deep learning approach for intrusion detection using recurrent neural networks. *Ieee Access*, 5:21954–21961, 2017.

[25] Ahmad Javaid, Quamar Niyaz, Weiqing Sun, and Mansoor Alam. A deep learning approach for network intrusion detection system. *Eai Endorsed Transactions on Security and Safety*, 3(9):e2, 2016.

[26] Yadigar Imamverdiyev and Fargana Abdullayeva. Deep learning method for denial of service attack detection based on restricted boltzmann machine. *Big data*, 6(2):159–169, 2018.

[27] Nathan Shone, Tran Nguyen Ngoc, Vu Dinh Phai, and Qi Shi. A deep learning approach to network intrusion detection. *IEEE transactions on emerging topics in computational intelligence*, 2(1):41–50, 2018.

[28] Davide Chicco. Siamese neural networks: An overview. *Artificial Neural Networks*, pages 73–94, 2021.

[29] Qinxue Meng, Daniel Catchpoole, David Skillicom, and Paul J Kennedy. Relational autoencoder for feature extraction. In *2017 International Joint Conference on Neural Networks (IJCNN)*, pages 364–371. IEEE, 2017.

[30] Klaus Greff, Rupesh K Srivastava, Jan Koutník, Bas R Steunebrink, and Jürgen Schmidhuber. Lstm: A search space odyssey. *IEEE transactions on neural networks and learning systems*, 28(10):2222–2232, 2016.

[31] Zhipeng Gui, Yunzeng Sun, Le Yang, Dehua Peng, Fa Li, Huayi Wu, Chi Guo, Wenfei Guo, and Jianya Gong. Lsi-lstm: An attention-aware lstm for real-time driving destination prediction by considering location semantics and location importance of trajectory points. *Neurocomputing*, 440:72–88, 2021.

[32] Muhammad Shakil Pervez and Dewan Md Farid. Feature selection and intrusion classification in nsl-kdd cup 99 dataset employing svms. In *The 8th International Conference on Software, Knowledge, Information Management and Applications (SKIMA 2014)*, pages 1–6. IEEE, 2014.

[33] Yalei Ding and Yuqing Zhai. 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*, pages 81–85, 2018.

[34] Bayu Adhi Tama, Marco Comuzzi, and Kyung-Hyune Rhee. Tse-ids: A two-stage classifier ensemble for intelligent anomaly-based intrusion detection system. *IEEE access*, 7:94497–94507, 2019.

[35] Cosimo Ieracitano, Ahsan Adeel, Francesco Carlo Morabito, and Amir Hussain. A novel statistical analysis and autoencoder driven intelligent intrusion detection approach. *Neurocomputing*, 387:51–62, 2020.

[36] Kaiyuan Jiang, Wenya Wang, Aili Wang, and Haibin Wu. Network intrusion detection combined hybrid sampling with deep hierarchical network. *IEEE Access*, 8:32464–32476, 2020.

[37] Yanfang Fu, Yishuai Du, Zijian Cao, Qiang Li, and Wei Xiang. A deep learning model for network intrusion detection with imbalanced data. *Electronics*, 11(6):898, 2022.