International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 10
DOI: https://doi.org/10.17762/ijritcc.v11i10.8499
***Article Received:** 31 July 2023 **Revised:** 22 September 2023 **Accepted:** 08 October 2023*

Enhancing Intrusion Detection Systems with a Hybrid Deep Learning Model and Optimized Feature Composition

Dr. Dilip Motwani[1], Dr Vidya Chitre[2], Dr Varsha Bhosale[3], Sonaali Borkar[4], D.K.Chitre[5]
[1,2,3,4]Professor, Vidyalankar institute of Technology, Wadala, Mumbai, Maharashtra, India
[5]Terna college of Engineering, Navi Mumbai, Maharashtra, India
dilip.motwani @vit.edu.in[1], vidya.chitre@vit.edu.in[2], varsha.bhosale@vit.edu.in[3], sonaali.borkar@vit.edu.in[4], dnyanobachitre@ternaengg.ac.in[5]

Abstract: Systems for detecting intrusions (IDS) are essential for protecting network infrastructures from hostile activity. Advanced methods are required since traditional IDS techniques frequently fail to properly identify sophisticated and developing assaults. In this article, we suggest a novel method for improving IDS performance through the use of a hybrid deep learning model and feature composition optimization. RNN and CNN has strengths that the proposed hybrid deep learning model leverages to efficiently capture both spatial and temporal correlations in network traffic data. The model can extract useful features from unprocessed network packets using CNNs and RNNs, giving a thorough picture of network behaviour. To increase the IDS's ability to discriminate, we also offer feature optimization strategies. We uncover the most pertinent and instructive features that support precise intrusion detection through a methodical feature selection and engineering process. In order to reduce the computational load and improve the model's efficiency without compromising detection accuracy, we also use dimensionality reduction approaches. We carried out extensive experiments using a benchmark dataset that is frequently utilized in intrusion detection research to assess the suggested approach. The outcomes show that the hybrid deep learning model performs better than conventional IDS methods, obtaining noticeably greater detection rates and lower false positive rates. The performance of model is further improved by the optimized feature composition, which offers a more accurate depiction of network traffic patterns.

Keywords: Intrusion Detection System, GRU framework, Optimization, Deep Learning, CNN, RNN

I. INTRODUCTION

Information exchange has been changed and made seamless and practical by the worldwide rapid rise of information technology. But even with these improvements, communication networks still have a lot of problems, especially with breaches and cyberattacks. Intrusion Detection Systems (IDS), which use a variety of detection techniques, have become crucial tools for locating and categorizing potential assaults on a network or host. The two basic kinds of IDS are signature-based IDS (SIDS) and anomaly-based IDS (AIDS). Network traffic patterns are compared with pre-established attack signatures or patterns by SIDS to identify assaults. In contrast, AIDS tracks network traffic patterns and contrasts them with typical or regular patterns to spot any deviations or anomalies, successfully identifying fresh and previously undiscovered attacks.

The constraints [3] of Signature-based IDS (SIDS) can be solved using a variety of development strategies, and researchers are becoming more and more interested in AIDS. By examining statistical parameters can identify intrusions. To apply Statistical IDS, invariant, multivariate, and time-series models are used. These models offer efficient tools for examining and spotting irregularities in network traffic. The foundation of knowledge-based approaches is a set of rules developed by human expertise. These techniques make use of expert systems, finite-state machines, and description languages. Knowledge-based IDS may efficiently identify abnormalities based on specified rules and patterns by utilizing human knowledge.

The development of anomaly-based IDS is frequently done using machine learning techniques. Unsupervised learning and supervised learning are its two main subcategories. Without depending on labelled examples, unsupervised learning locates anomalies by locating patterns that differ from the system's typical behaviour. Contrarily, supervised learning needs labelled examples in order to train the model to correctly recognize anomalies. Learning from a mixture of huge numbers of unlabelled cases and a smaller collection of tagged instances is called semi-supervised learning. The usage of machine learning algorithms to detect cyber intrusions is on the rise due to the former's autonomy and rapid response times. However, due to the dynamic nature of cyber-attacks, scalable and adaptable detection systems must be developed. Deep learning (DL) techniques allow for the creation of such scalable systems as a real possibility. Intruder detection in both supervised and unsupervised systems can benefit from DL's use [6, 7].

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 10
DOI: https://doi.org/10.17762/ijritcc.v11i10.8499
***Article Received:** 31 July 2023 **Revised:** 22 September 2023 **Accepted:** 08 October 2023*

This problem has been solved by long-term dependency handling techniques included in existing deep learning solutions. We used the GRU framework, a kind of recurrent neural network (RNN), to address this problem. Long-term dependency issues can be solved by gated recurrent unit by adaptively updating and resetting its memory state. We wanted to improve the long-term contextual information acquisition and utilization of our deep learning model by integrating gated recurrent unit. Our proposed IDS gets more adept at identifying assaults by utilizing the advantages of CNN for extracting spatial information and proposed model for managing sequential dependencies. A comprehensive framework that can assess both the local patterns and the temporal dynamics present in network traffic data is created by combining these two designs, and the result is better intrusion detection performance.

II. REVIEW OF LITERATURE

Based on their findings, Achmad et al. [15] proposed a mixed approach to intrusion detection, one that makes use of both supervised feature optimisation and unsupervised data reduction strategies. Attribute Importance Decision Tree (DT) with recursive feature removal was utilised as a feature optimisation strategy to select useful and significant characteristics. They also employed the Local Outlier Factor (LOF) method to identify unusual data points. The researchers tested their methods on the NSL-KDD and UNSW-NB15 datasets, two of the most popular in the field. According to their findings, their hybrid model is more accurate than other intrusion detection systems. However, they acknowledged that the system's false acceptance rate (FAR), sensitivity, and specificity may use improvement. By integrating supervised and unsupervised methods, the hybrid approach described by Achmad et al. aimed to boost the intrusion detection system's efficiency. Attribute significance DT-based feature optimisation and the LOF technique for outlier detection increased the model's ability to accurately identify and classify incursions. The study authors emphasised the need for further development to target specific performance measures and boost the method's overall effectiveness. The experimental results validated that the interpretations generated by their framework accurately reflect the key features of the attacks. The decision-making process of the IDS and the underlying issues affecting its forecasts were made more transparent thanks to the framework's deemed-simple explanations.

Even though there are many ways to improve Intrusion Detection Systems (IDS), there is still room for development. In contrast to conventional approaches, there has been an increasing tendency in recent years to design IDSs based on machine learning. However, the effectiveness of IDSs might change based on the particular technique used and the settings in which they are used. Using the UGR'16 dataset, the authors assessed the performance of these models with an emphasis on four different types of attacks: Denial of Service and a botnet attacks. The results of their study offer the scientific community useful information, particularly with regard to the creation of better Network Intrusion Detection System (NIDS) solutions. It is crucial to remember that the study's main objective was to assess how well the models performed against the four chosen attacks. This research did not go into great detail about the wider landscape of potential attacks and their detection.

SVM, KNN, were among the machine learning algorithms used in a study by Iram et al. [12] to design an intrusion detection system (IDS). Dimensionality reduction was accomplished using a random selection of features from the NSL-KDD dataset. All three classifiers (DT, extra-tree, and RF) performed at or above 99% accuracy. However, the use of optimisation methods was not the focus of the research. With DT, RF, and XGBoost classifiers in mind, Abdulsalam et al. [13] focused on developing IDS for SDN. The NSL-KDD dataset was used to evaluate the performance of the models. The XGBoost classifier outperformed the competition on several metrics, including the F1 score, the precision, and the recall.

For dataset optimization [14], the GIWRF model, an embedded feature selection method, was used. In the investigation, the decision tree classifier performed better than other models. It should be emphasized, nonetheless, that this study did not evaluate multiclass classification. Some research missed optimization techniques or certain elements, such multiclass classification, while getting high accuracy scores and investigating various classifiers. To improve IDS performance and solve real-world intrusion scenarios, future research might concentrate on merging optimization techniques and tackling particular issues.

Mario et al.'s study [21] involved studies to contrast several neural-based methods with an emphasis on Artificial Neural Networks (ANN). Their model's performance was assessed using the KDD99 and CICIDS2017/2018 datasets. According to the researchers, most of the time, ANN-based approaches displayed exceptional performance. However, because ANN employs the backpropagation technique, these models have poor processing speeds. It is significant to remind that their study did not include a feature optimization step, which would have decreased the classifier's time complexity. While Mario et al.'s work focused on the excellent performance of ANN-based approaches, it disregarded the inclusion of feature optimization, which may have advantages in terms of computing effectiveness. Shi et al.'s research, on the other hand, demonstrated the application of a Semi-Supervised

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 10
DOI: https://doi.org/10.17762/ijritcc.v11i10.8499
***Article Received:** 31 July 2023 **Revised:** 22 September 2023 **Accepted:** 08 October 2023*

Deep Reinforcement approach, but it also brought to light the SSDDQN model's shortcomings in terms of optimization and spotting specific kinds of anomalous attack traffic.

As opposed to using features obtained through a classification strategy directly, Joohwa et al.'s research [24] offered an approach for deep learning classification by leveraging features acquired through a pre-processing technique. They used the Random Forest (RF) classification algorithm along with an unsupervised deep learning autoencoder model. A deep sparse autoencoder was employed to extract the features. The CICIDS 2017 dataset was used for the studies. The authors asserted that their suggested strategy outperformed current feature extraction techniques. They did observe that the approach's performance was rather subpar for the network's unusual class. A multi-stage optimised Machine Learning (ML)-based Network Intrusion Detection System (NIDS) framework was introduced by Mohammadnoor et al. [25]. In order to establish the minimum training sample size necessary, they looked into the effects of oversampling techniques on model training instance sizes. Gain-based and correlation-based feature selection methods were contrasted by the researchers. To assess their model, CICIDS 2017 and UNSW-NB 2015 datasets were employed. They asserted that their framework used just up to 50% of the features at hand and still had an accuracy rate of above 90%.

The many methods of implementing an Intrusion Detection System have been covered in the cited research. Many of these research, like KDDCUP and NSL-KDD, evaluate their models on datasets with limited attribute variability. To address this problem and guarantee a comprehensive analysis, our efforts are focused on the CICIDS-2017 dataset, which gives more features and has a more diverse variety of risks than the KDD dataset. Our suggested work focuses on reducing the quantity or size of input features by using a reliable feature optimisation method. By cherry-picking the most relevant and instructional data points from the dataset, this approach aims to strengthen the reliability and performance of the intrusion detection mechanism.

III. PUBLICALLY AVAILABLE DATASETS

The dataset was specifically created to incorporate a more varied range of attack types and traffic patterns in order to solve the shortcomings of earlier datasets like KDDCUP and NSL-KDD.

The following are some of the primary qualities and traits of the CICIDS-2017 dataset:

The dataset was created using actual network traffic that was recorded in a regulated setting, giving a true depiction of network behavior.

It has a wide range of features that were taken from network traffic, including as statistical features, flow-based features, and features based on transport layer protocols. These features record crucial data about network behavior and communication.

Class Imbalance: The dataset displays class imbalance, where some attack types are more common than others, just like in real-world network environments.

Size: The dataset is big, with millions of network traffic examples, making it appropriate for developing and testing IDS models based on machine learning.

Available Protocols: The collection contains information about the existence of widely used protocols as Email, HTTP, FTP, HTTPS and SSH. This guarantees that the dataset covers a wide range of network communication protocols.

Attack Diversity: Based on the 2016 McAfee study, the dataset includes a range of attack types. Web-based assaults, brute-force attacks, Heartbleed attacks, bot attacks, and scan attacks are among them. The evaluation and testing of intrusion detection systems under multiple assault scenarios is made possible by the comprehensive attack diversity.

Heterogeneity: The dataset records system calls and memory dumps from each victim machine as well as network activity from the main switch while the assaults were being carried out.

Feature Set: Using the CICFlowMeter, more than 80 network flow features were retrieved from the generated network traffic. The dataset's foundation is made up of several features, which offer comprehensive information about the network flows. The dataset is often offered in CSV file format, making analysis of it simple.

MetaData: Detailed information regarding the time of the captured network traffic, the sorts of assaults that were present, the network flows, and the accompanying labels are all included in the dataset's extensive metadata. In order to ensure transparency and promote a deeper comprehension of the dataset's contents, this metadata is often supplied in the published paper that goes along with the dataset.

Table 1: Description of Dataset

Dataset Name	**CICIDS-2017**
Used Protocol	Email protocols, HTTP, SSH, HTTPS and FTP
Diversity of Attack	DoS, Web-based, DDoS, Infiltration, Brute force, Heartbleed, and Scan
Heterogeneity	System calls, memory dumps, and captures from the main switch
Feature Set	More than 80 network flow features

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 10
DOI: https://doi.org/10.17762/ijritcc.v11i10.8499
Article Received: *31 July 2023* ***Revised:*** *22 September 2023* ***Accepted:*** *08 October 2023*

MetaData	Time, Attack, Labels, Time and flows

The analysis of network behaviour and attack patterns is made easier by these features, which give precise information about the network flows. A deeper comprehension of the dataset's contents is made possible by the inclusion of extensive metadata, such as the period during which network traffic was gathered, the sorts of attacks that were active, network flows, and associated labels.

IV. PROPOSED SYSTEM

The CNN-RNN model and the feature optimization technique make up the two main portions of the proposed work. The workflow shown in figure 1, the proposed method and it consist of two major component such as: classification and data pre-processing. The workflow's initial phase is centred on feature optimization. This entails removing redundancy and choosing the best feature set for the analysis that follows. The objective is to improve the caliber and relevance of the features utilized in the model by employing an efficient feature optimization technique.

The data must be organized and prepared for classification in this step. The data can be more effectively analysed and identified by categorizing them. The next step in the workflow entails comparing the performance of the suggested technique against that of other available algorithms. The purpose of this evaluation is to ascertain the superiority and efficacy of the suggested method for identifying and categorizing intrusions.

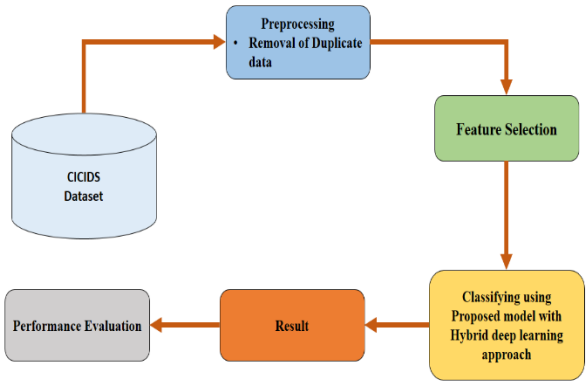

Figure 1: Proposed model for IDS

1. Data Gathering and Pre Processing :

The dataset ICIDS 2017 as assess and used for the efficacy of proposed method. This dataset allowed for a thorough investigation of 25 users' abstract behaviour based on several protocols. Five days of continuous data collection allowed for a thorough investigation of network activity. In the ICIDS 2017 dataset, a variety of attack types are covered, including DoS/DDoS attacks, brute force assaults, and web attacks, attempts at penetration, botnet activity, port scans, and more. The dataset offers a wide range of assault situations, which makes it ideal for testing and gauging the effectiveness of our approach. The attack distribution inside the ICIDS 2017 dataset, offering insightful data on the prevalence and frequency of various attack types. The ICIDS 2017 dataset is strongly advised for testing and validating intrusion detection models because of its thorough coverage of multiple attack categories.

Table 2: Different types of attacks

Sub-Dataset	Attacks
Tuesday Samples	benign, ftpPatator_Attack, sshPatator_Attack
Sample on Wednesday	Safe from GoldenEye, Hulk, Slow HTTP Test, Slow Loris, and Heartbleed Attacks
Thursday Morning Samples	benign, bruteForce_Attack, SqlInjection_Attack, XSS_Attack
Thursday Afternoon Samples	benign, infiltration_Attack
Friday Morning Samples	benign, bot_Attack
Samples on Afternoon of Friday	ddos_Attack and benign,
Samples on Afternoon of - PortScan	portscan_Attack and benign

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 10
DOI: https://doi.org/10.17762/ijritcc.v11i10.8499
Article Received: *31 July 2023* ***Revised:*** *22 September 2023* ***Accepted:*** *08 October 2023*

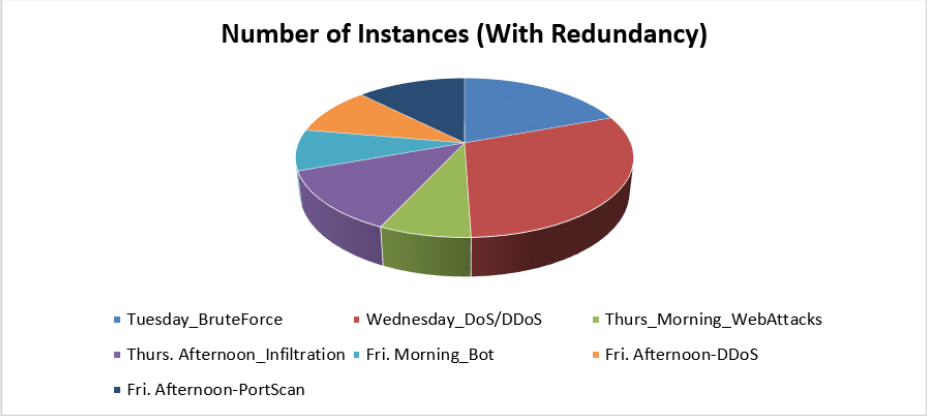

Figure 2: Representation of Dataset number of Instance with Redundancy

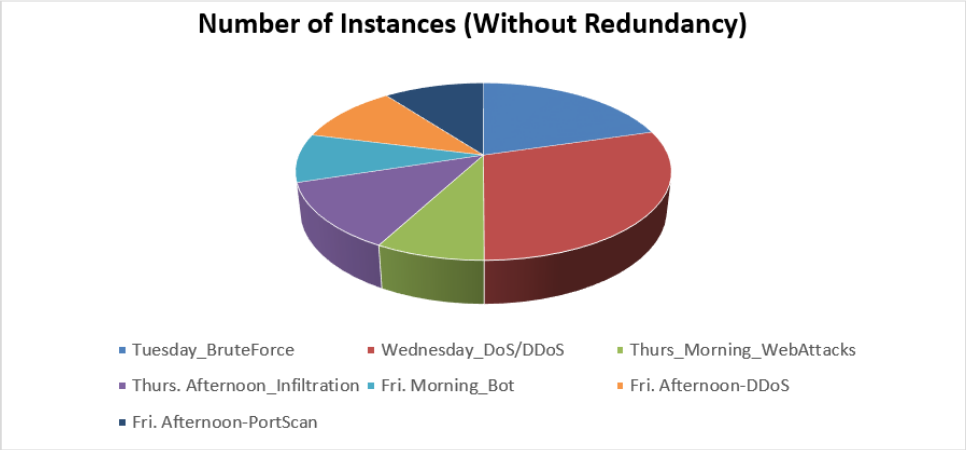

Figure 3: Representation of Dataset number of Instance with Redundancy

A feature optimization approach was used to identify the most pertinent features while lowering the input dimension. This tactic used the (PCC) Pearson's Correlation Coefficient filter approach to find out discriminative features during attribute selection and collection. Correlation by Pearson The correlation or similarity between various aspects or qualities in the dataset is measured by the coefficient. A correlation coefficient value between [-1, 1] is provided. Strong positive correlation is shown by a coefficient value of 1, whilst strong negative correlation is denoted by a value of -1. An almost-zero coefficient value denotes a poor or non-existent link between the features.

$$\rho X, Y = \frac{(X, Y)}{\sigma X \sigma Y}$$

$$PCR = \frac{\mathrm{Ei}(\mathrm{xiyi}) - \mathrm{Ej}(\mathrm{xi})\ \mathrm{Ek}(\mathrm{yi})}{\sqrt{\mathrm{Ei}(x1^2) - Ej^2(\mathrm{x2})\sqrt{\mathrm{B}(y^2) - \mathrm{E2}(y)}}}$$

In our study, we looked at many categories of network attacks and used attribute selection to find the best features for each category. BruteForce, DoS/DDoS, WebAttacks, Infiltration, Bot, DDoS, and PortScan were the classes taken into consideration. We started with a total of 77 attributes available for study for each class. The amount of attributes was nevertheless reduced by the attribute selection procedure to a more manageable and useful subset. The purpose of the attribute selection process was to determine the characteristics that are most helpful in identifying and classifying each individual attack type.

2. LSTM improved framework as Gated Recurrent Unit:

To combat the vanishing/exploding gradient problem, researchers have developed a newer architecture for recurrent neural networks called the Gated Recurrent Unit (GRU). The LSTM framework (Long Short-Term Memory) has been enhanced. GRU employs a gate structure to manage data flow in a manner analogous to that of LSTM. There are, however, notable differences between the two. In contrast to LSTM, GRU doesn't have an output gate, hence the hidden state is exposed. GRU consists of just two gates, the reset gate and the update gate. The update gate controls how much of the

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 10
DOI: https://doi.org/10.17762/ijritcc.v11i10.8499
***Article Received:** 31 July 2023 **Revised:** 22 September 2023 **Accepted:** 08 October 2023*

previous concealed state is shown, while the reset gate determines how much of it is forgotten.

$$\mathrm{RTi} = \mathrm{Sigmoid}(\mathrm{Wt\ XrXt} + \mathrm{Wt\ HrHt} - 1 + B_r)$$

$$\mathrm{ZT(l)} = \mathrm{Sigmoid}(\mathrm{Wt\ XrXt} + \mathrm{Wt\ HrHt} - 1 + B_z)$$

$$\mathrm{H(t)} = \mathrm{TanH}(\mathrm{Wt\ XrXt} + \mathrm{Wt\ HrHt}(\mathrm{RTi} \odot \mathrm{Ht} - 1) + B_h)$$

$$\mathrm{H(t)} = \mathrm{ZT(l)} \odot \mathrm{Ht} - 1 + (1 - \mathrm{ZT(l)}) \odot \mathrm{H(t)}$$

One advantage of GRU over LSTM is that it has a simpler structure and fewer parameters, both of which can improve performance. GRU's computational efficiency and propensity for overfitting are improved by the smaller number of parameters. Additionally, the lack of an output gate makes it possible for GRU to transfer information more directly, potentially enabling faster learning and greater temporal dependency capture in sequential data.

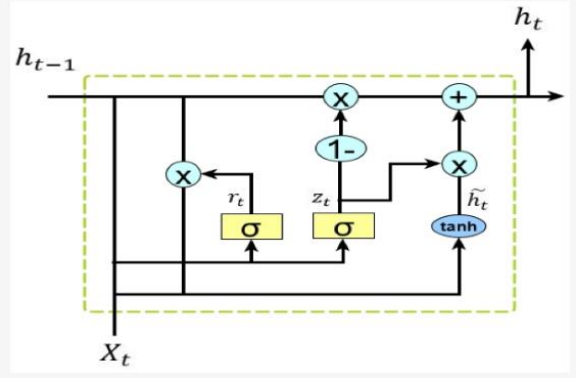

Figure 4: Gated Recurrent Architecture

3. Recurrent Neural Network (RNN):

In order to implement the algorithm, Recurrent Neural Networks (RNNs) are used in intrusion detection. An outline of the procedures for employing RNN for intrusion detection is provided below:

- Gather the dataset comprising information on network traffic, including both legitimate and malicious instances. This dataset needs to have the appropriate attack kinds annotated.
- Pre-processing the data involves cleaning it up, normalizing the features, and handling any missing values that may be there.
- Feature Extraction: Take the preprocessed dataset and extract the pertinent features. This stage tries to record the crucial aspects of network traffic that can aid in differentiating between legitimate and malicious activity.
- Create sequences from the collected features that are appropriate for RNN training. In order to do this, input sequences and corresponding target sequences must be created, where input sequences represent a succession of prior network states and target sequences represent the occurrence of an attack or regular behaviour.
- Design an RNN architecture for the model to use in intrusion detection. This normally entails deciding on the right kind of RNN cells (such as LSTM or GRU) and the network's layer and neuron count. The issue of the vanishing/exploding gradient should be taken into account.
- Model training: Apply the prepared dataset to the RNN model training. The input sequences are fed into the RNN during training, and then the output predictions are calculated and compared to the target sequences. Utilize optimization strategies such as backpropagation
- Model evaluation: Use the testing dataset to assess the trained RNN model. Determine different performance criteria, such as accuracy, precision, recall, and F1-score, to evaluate how well the model detects network intrusions.
- Optimization and fine-tuning: Boost the model's performance by fine-tuning its hyperparameters, such as learning rate, batch size, and regularization methods. To prevent overfitting, think about adopting strategies like early stopping or learning rate decay.
- Deploy the model for intrusion detection in a real-time or almost real-time scenario once you are pleased with its performance. Make predictions about whether the observed behavior is malicious or normal by continuously monitoring network traffic and feeding it into the deployed RNN model.
- Continuous Improvement: Track the model's performance over time and make periodic updates to make it flexible to changing attack

4. CNN Model:

The Convolutional Neural Network (CNN), often known as a ConvNet, is a type of deep learning system that processes input data by giving various layers of the network distinct biases and weights. After that, it performs the steps of Algorithm 1 to divide the input into its constituent parts. One of CNN's key advantages over competing algorithms is its ability to cut down on the amount of pre-processing required before data is ready to be used. This is because filters may be automatically learned and improved by CNN.

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 10
DOI: https://doi.org/10.17762/ijritcc.v11i10.8499
Article Received: 31 July 2023 *Revised:* 22 September 2023 *Accepted:* 08 October 2023

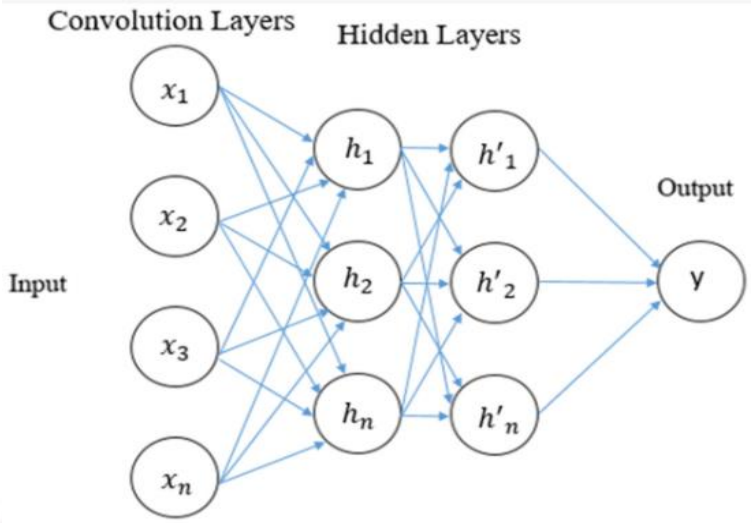

Figure 5: CNN Architecture

Proposed Model Algorithm:

$CNN - RNN\ Feature - Level\ Optimisation$

$Input$: $Data\ examples$

$Confusion\ Matrix\ (Accuracy, Precision, Recall, False\ Positive\ Rate, True\ Positive\ Rate)$

$Optimising\ Datasets$

$Get\ rid\ of\ the\ duplicates.$

$Choose\ Among\ Features$

$Calculate\ the\ attribute\ set's\ correlation\ using\ Pearson's\ Correlation\ formula.$

$If\ corr\_value\ is\ more\ than\ 0.8, then\ set\ Cf.$

$insert\ attribute\ if\ Cf\ exists, otherwise\ increase\ attribute\ count\ Cf\ C\ return$

$Classification$

$Build\ the\ dataset's\ training\ and\ test\ sets.$

$67\%\ in\ the\ training\ set$

$Data\ sample\ size$: 33%

$Model\ with\ three\ Convolution\ layers\ with\ relu\ activation$

$and\ two\ GRU\ layers\ with\ relu\ activation.$

$The\ 'categorical\_crossentropy'\ loss\ function\ was\ used\ during\ model\ compilation.$

$optimizer = 'adagrad'$

$training\ Methods\ from\ the\ CNN - GRU\ training\ set\ are\ applied\ to\ the\ CNN - GRU\ test\ set.$

$return\ The\ Matrix\ of\ Confuson\ Cm * m$

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 10
DOI: https://doi.org/10.17762/ijritcc.v11i10.8499
***Article Received:** 31 July 2023 **Revised:** 22 September 2023 **Accepted:** 08 October 2023*

5. Evaluation Metrics:

Following is a summary of the four performance evaluation measures that were utilised to evaluate the proposed illness prediction model:

- True positives (TP) are the number of precise forecasts when the model correctly recognises a patient as having a chronic illness.
- True negatives (TN) are the proportion of precise predictions in which the model correctly identifies individuals who are free of any diseases.
- False Positives (FP): The quantity of inaccurate predictions in which the model misdiagnoses a healthy person as having a condition.
- False Negatives (FN): The proportion of inaccurate predictions in which the model incorrectly classifies a patient as healthy when, in reality, they are suffering from a chronic illness.

The model's precision, recall, and overall performance in predicting the presence or absence of chronic diseases are all valuable insights revealed by these metrics.

a) Precision:

The precision or positive predictive value (PPV), a performance evaluation measure, determines the percentage of accurate forecasts to all correct values, including both true and false values. Mathematically, it is represented as follows:

$$\text{The Precision} = \frac{\text{True Positive}}{\text{True Positive} + \text{False Positive}}$$

In other words, precision describes how accurate or precise the model is at foreseeing favorable events. With a high precision, the model is less likely to misclassify negative cases as positive, or have a high rate of false positives.

b) Recall:

The quantity of true positives, or TP, in this equation represents the precision of the forecasts made by patients with chronic diseases. The number of times a healthy person was incorrectly diagnosed as having a disease is known as false positives (FP). By dividing the total number of true positives by the sum of true positives and false positives, the accuracy or positive predictive value is determined. Recall, also known as sensitivity or true positive rate (TPR), is a performance evaluation metric that calculates the ratio of correctly expected values to all correctly positive and wrongly negative projected values. The mathematical notation is as follows.

$$\text{The Recall} = \frac{\text{True Positive (FP)}}{\text{True Positive (TP)} + \text{False Negative (FN)}}$$

The number of true positives, or TP, in this equation denotes the accuracy of patients with chronic conditions' forecasts. False negatives, or FP, are the number of instances where a patient was incorrectly classified as healthy.

c) F1 Score:

The performance evaluation statistic known as the F-measure (F) combines the precision and recall criteria through a weighted average. It is especially useful when the distribution of classes is uneven or the ranges of false positives and false negatives are wide. When recall and precision are equally important, the F1-Score, a particular variation of the F-measure, is frequently employed. It is denoted mathematically as follows:

$$F1 \text{ Value score} = \frac{2 * \text{The Recall Value} \times \text{The Precision Value}}{(\text{The Recall Value} + \text{The Precision Value})}$$

This equation uses precision to stand in for accuracy, also known as positive predictive value, and recall, also known as true positive rate. Precision defines the relative weights of accuracy and recall.

V. RESULT AND DISCUSSION

To evaluate the trained RNN model, use the testing dataset. To assess how well the model detects network intrusions, determine various performance parameters like accuracy, specificity, sensitivity, and F1-score. Optimization and fine-tuning, boost the model's performance by fine-tuning its hyperparameters, like different batch size, the different learning rate, and regularization methods. To prevent overfitting, think about adopting strategies like early stopping or learning rate decay. Deploy the model for intrusion detection in a real-time or almost real-time scenario once you are pleased with its performance. Make predictions about whether the observed behaviour is malicious or normal by continuously monitoring network traffic and feeding it into the deployed RNN model. Continuous Improvement: Track the model's performance over time and make periodic updates to make it flexible to changing attack

Table 3: Evaluation parameters metrics of different algorithm

Algorithm	Model Accuracy in (%)	Model Precision in (%)	Model Recall in (%)	Model F1-Score in (%)
CNN	93.78	93.23	91.76	92.55
RNN	90.21	91.33	90.31	91.87
Hybrid Approach (CNN and RNN)	99.73	94.22	99.16	98.92

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 10
DOI: https://doi.org/10.17762/ijritcc.v11i10.8499
Article Received: 31 July 2023 Revised: 22 September 2023 Accepted: 08 October 2023

As intrusion detection algorithms, CNN, RNN, and a hybrid CNN-RNN technique (shown in table 3) were assessed based on measures for accuracy, precision, recall, and F1-score. CNN had a 93.78% accuracy rate, 93.23% precision rate, 91.76% recall rate, and a 92.55% F1-score, according to the results. RNN achieved a 91.87% F1-score, 90.21% accuracy, 91.33% precision, and 90.31% recall. The hybrid technique combining CNN and RNN outperformed both individual algorithms, achieving accuracy of 99.73%, precision of 94.22%, recall of 99.16%, and an F1-score of 98.92%. The higher classification accuracy of the hybrid technique for intrusion detection highlighted the synergistic advantages of integrating CNN and RNN.

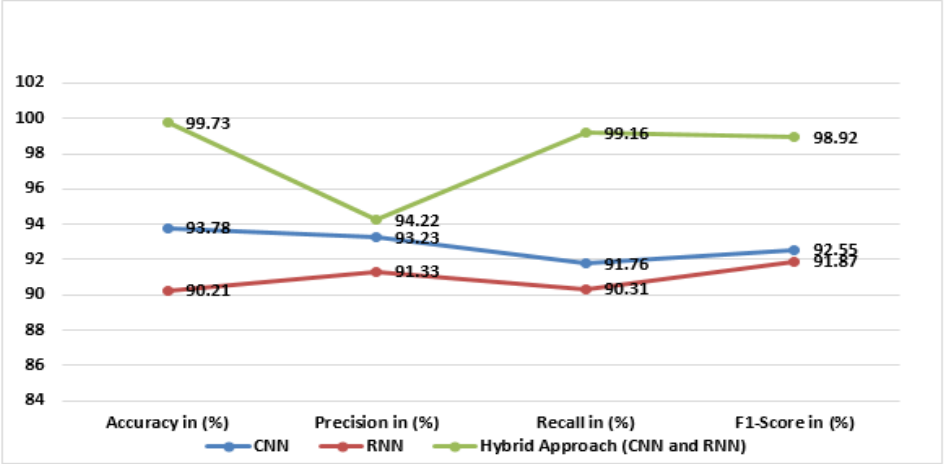

Figure 6: Representation for Comparison of Performance metrics for Deep learning method

Table 4: Statistical Analysis of Attacks (DDoS)

Evaluation metrics	BENIGN	Goldeneye	Hulk	Slowhttptest	Slowloris	Heartbleed
The Precision	94.4	77.02	96.77	98.88	80.96	0
The Recall	95	74	95	88	79	0
True Negative	59202.98	198336	142083	200466.1	200366	202377
False Positive	5019.98	622.96	2950.02	223.14	273.97	4
True Positive	134433	2660.96	53510.02	1252.14	1434.97	0
False Negative	3728.98	764.96	3842.02	444.14	309.97	4
(FPR)	58	02	4	14.1	3	0
(TPR)	95.3	73.67	95.3	87.82	79.23	0

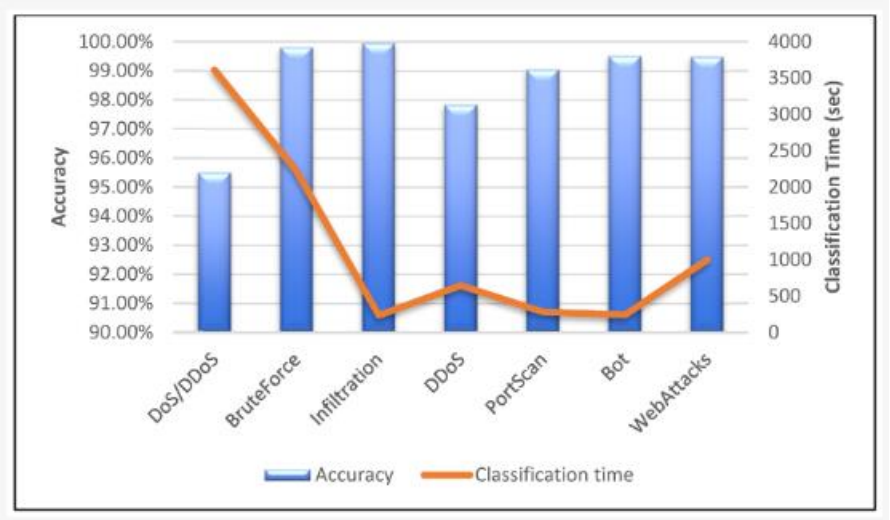

Figure 7: Accuracy and classification

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 10
DOI: https://doi.org/10.17762/ijritcc.v11i10.8499
***Article Received:** 31 July 2023 **Revised:** 22 September 2023 **Accepted:** 08 October 2023*

The accuracy of the CNN model [18], which used 77 attributes, was 95.26 percent. This shows that it can correctly categorize incursions using the supplied dataset. With 78 attributes and a greater accuracy of 98.67%, the CNN-AE model [36] showed increased performance in intrusion detection. A second CNN model [19] used 77 attributes and had a 98.72% accuracy rate. This shows that it was successful in correctly categorizing incursions in the dataset. While using 74 attributes, the LSTM model [20] achieved a slightly better accuracy of 98.87%, demonstrating its capability to efficiently detect intrusions.

Table 5: Performance of proposed model with Existing model

Model	No. of Attributes	Accuracy
CNN [18]	77	95.26%
CNN-AE [36]	78	98.67%
CNN [19]	77	98.72%
LSTM [20]	74	98.87%
Proposed CNN-RNN	41	99.73%

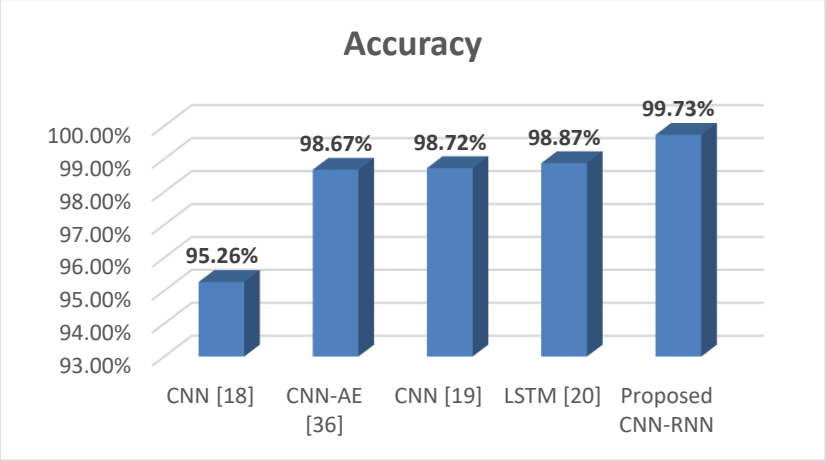

Figure 8: Comparison Performance of proposed model with Existing model

The proposed CNN-RNN model had the best accuracy of 99.73% among the tested models and used a smaller set of 41 attributes. This shows that, when compared to previous models, the suggested approach performs better at reliably classifying intrusions. Overall, great accuracy in intrusion detection was shown by the CNN-AE, CNN [19], LSTM, and suggested CNN-RNN models. The proposed CNN-RNN model outperformed the previous models despite having less features, demonstrating how well it can detect intrusions. These results demonstrate how deep learning models can be applied to intrusion detection systems to improve security and defend against online attacks.

VI. CONCLUSION

IDSs (intrusion detection systems) are essential for defending organizational borders against online dangers. In order to successfully detect and categorize intrusions, new techniques are becoming necessary given the complexity and frequency of attacks. In this study, we suggested an optimal feature composition method and a hybrid deep learning model to improve IDS performance. Using their respective advantages in feature extraction and sequence modeling, Convolutional Neural Networks (CNN) and Recurrent Neural Networks (RNN) were merged in our method. We attempted to capture both local and temporal dependencies inside network traffic data using the CNN and RNN architectures, resulting in enhanced detection accuracy. The hybrid model performed exceptionally well, attaining an accuracy of 99.73%. We used feature optimization approaches to minimize the dimensionality of the input data in order to further increase the model's efficacy. We determined and chose the most pertinent features by using Pearson's correlation coefficient, which also increased the model's efficiency and calculation speed while keeping high accuracy. The experimental examination of our suggested strategy against a number of benchmark datasets demonstrated its superiority to earlier approaches. In terms of accuracy, the hybrid CNN-RNN model outperformed the standalone CNN and RNN models as well as other cutting-edge algorithms. This demonstrates the beneficial synergy between the CNN and RNN designs in capturing complex dependencies and patterns in network traffic data.

Future research directions might include examining various deep learning architectures, adding further data sources and characteristics, and examining how generalizable the suggested strategy is in various network contexts. Additionally, work can be done to incorporate real-time monitoring tools and create defences against emerging and zero-day assaults.

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 10
DOI: https://doi.org/10.17762/ijritcc.v11i10.8499
***Article Received:** 31 July 2023 **Revised:** 22 September 2023 **Accepted:** 08 October 2023*

REFERENCES

[1] G. De Carvalho Bertoli et al., "An End-to-End Framework for Machine Learning-Based Network Intrusion Detection System," in IEEE Access, vol. 9, pp. 106790-106805, 2021, doi: 10.1109/ACCESS.2021.3101188.

[2] Z. A. El Houda, B. Brik and S. -M. Senouci, "A Novel IoT-Based Explainable Deep Learning Framework for Intrusion Detection Systems," in IEEE Internet of Things Magazine, vol. 5, no. 2, pp. 20-23, June 2022, doi: 10.1109/IOTM.005.2200028.

[3] A. Pandit, A. Gupta, M. Bhatia and S. C. Gupta, "Filter Based Feature Selection Anticipation of Automobile Price Prediction in Azure Machine Learning," 2022 International Conference on Machine Learning, Big Data, Cloud and Parallel Computing (COM-IT-CON), Faridabad, India, 2022, pp. 256-262, doi: 10.1109/COM-IT-CON54601.2022.9850615.

[4] C. -M. Ou, "Host-based Intrusion Detection Systems Inspired by Machine Learning of Agent-Based Artificial Immune Systems," 2019 IEEE International Symposium on INnovations in Intelligent SysTems and Applications (INISTA), Sofia, Bulgaria, 2019, pp. 1-5, doi: 10.1109/INISTA.2019.8778269.

[5] P. Widulinski and K. Wawryn, "Parameter Efficiency Testing for an Intrusion Detection System Inspired by the Human Immune System," 2022 29th International Conference on Mixed Design of Integrated Circuits and System (MIXDES), Wrocław, Poland, 2022, pp. 208-212, doi: 10.23919/MIXDES55591.2022.9838210.

[6] Khraisat, A.; Gondal, I.; Vamplew, P.; Kamruzzaman, J., "Survey of intrusion detection systems: Techniques, datasets and challenges", Cybersecurity **2019**, 2, 20

[7] D. Dal, S. Abraham, A. Abraham, S. Sanyal and M. Sanglikar, "Evolution Induced Secondary Immunity: An Artificial Immune System Based Intrusion Detection System," 2008 7th Computer Information Systems and Industrial Management Applications, Ostrava, Czech Republic, 2008, pp. 65-70, doi: 10.1109/CISIM.2008.31.

[8] V. Hnamte and J. Hussain, "An Extensive Survey on Intrusion Detection Systems: Datasets and Challenges for Modern Scenario," 2021 3rd International Conference on Electrical, Control and Instrumentation Engineering (ICECIE), Kuala Lumpur, Malaysia, 2021, pp. 1-10, doi: 10.1109/ICECIE52348.2021.9664737.

[9] J. Zhang, M. Zulkernine and A. Haque, "Random-Forests-Based Network Intrusion Detection Systems," in IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews), vol. 38, no. 5, pp. 649-659, Sept. 2008, doi: 10.1109/TSMCC.2008.923876.

[10] Kwon, D.; Kim, H.; Kim, J.; Suh, S.C.; Kim, I.; Kim, K.J. A survey of deep learning-based network anomaly detection. Clust. Comput. **2017**, 22, 949–961.

[11] W. Cao, H. Zhang, W. He, H. Chen and E. H. Tat, "Autoencoder in Autoencoder Network Based on Low-Rank Embedding for Anomaly Detection in Hyperspectral Images," IGARSS 2022 - 2022 IEEE International Geoscience and Remote Sensing Symposium, Kuala Lumpur, Malaysia, 2022, pp. 3263-3266, doi: 10.1109/IGARSS46834.2022.9884142.

[12] J. Lansky et al., "Deep Learning-Based Intrusion Detection Systems: A Systematic Review," in IEEE Access, vol. 9, pp. 101574-101599, 2021, doi: 10.1109/ACCESS.2021.3097247.

[13] X. Wang, L. Wang and Q. Wang, "Local Spatial–Spectral Information-Integrated Semisupervised Two-Stream Network for Hyperspectral Anomaly Detection," in IEEE Transactions on Geoscience and Remote Sensing, vol. 60, pp. 1-15, 2022, Art no. 5535515, doi: 10.1109/TGRS.2022.3196409.

[14] Meryem, A.; Ouahidi, B.E.L. Hybrid intrusion detection system using machine learning. Netw. Secur. **2020**, 2020, 8–19.

[15] S. A. Bajpai and A. B. Patankar, "A Study on Self-Configuring Intrusion Detection Model based on Hybridized Deep Learning Models," 2023 7th International Conference on Computing Methodologies and Communication (ICCMC), Erode, India, 2023, pp. 303-309, doi: 10.1109/ICCMC56507.2023.10084290.

[16] Abrar, I.; Ayub, Z.; Masoodi, F.; Bamhdi, A.M. A machine learning approach for intrusion detection system on NSL-KDD dataset. In Proceedings of the 2020 International Conference on Smart Electronics and Communication (ICOSEC), Trichy, India, 10–12 September 2020.

[17] Alzahrani, A.O.; Alenazi, M.J. Designing a network intrusion detection system based on machine learning for software defined networks. Future Internet **2021**, 13, 111.

[18] Disha, R.A.; Waheed, S. Performance analysis of machine learning models for intrusion detection system using Gini impurity-based weighted random forest (GIWRF) feature selection technique. Cybersecurity **2022**, 5, 1.

[19] Megantara, A.A.; Ahmad, T. A hybrid machine learning method for increasing the performance of Network Intrusion Detection Systems. J. Big Data **2021**, 8, 142.

[20] Ho, S.; Jufout SAl Dajani, K.; Mozumdar, M. A Novel Intrusion Detection Model for Detecting Known and Innovative Cyberattacks Using Convolutional Neural Network. IEEE Open J Comput Soc. **2021**, 2, 14–25.

[21] Priyanka, V.; Gireesh Kumar, T. Performance Assessment of IDS Based on CICIDS-2017 Dataset. In Information and Communication Technology for Competitive Strategies (ICTCS 2020); Lecture Notes in Networks and Systems; Joshi, A., Mahmud, M., Ragel, R.G., Thakur, N.V., Eds.; Springer: Singapore, 2022; Volume 191.

[22] 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.

[23] Mauro, M.D.; Galatro, G.; Liotta, A. Experimental Review of Neural-based approaches for network intrusion management. IEEE Trans. Netw. Serv. Manag. **2020**, 17, 2480–2495.

[24] Dong, S.; Xia, Y.; Peng, T. Network abnormal traffic detection model based on semi-supervised Deep Reinforcement Learning. IEEE Trans. Netw. Serv. Manag. **2021**, 18, 4197–4212.

[25] Pelletier, C.; Webb, G.I.; Petitjean, F. Deep learning for the classification of sentinel-2 Image time series. In Proceedings of the IGARSS 2019—2019 IEEE International Geoscience and Remote Sensing Symposium, Yokohama, Japan, 28 July–2 August 2019.

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 10
DOI: https://doi.org/10.17762/ijritcc.v11i10.8499
Article Received: 31 July 2023 Revised: 22 September 2023 Accepted: 08 October 2023

[26] Lee, J.; Pak, J.G.; Lee, M. Network intrusion detection system using feature extraction based on deep sparse autoencoder. In Proceedings of the 2020 International Conference on Information and Communication Technology Convergence (ICTC), Jeju, Korea, 21–23 October 2020.