National College of Ireland

Improve the detection accuracy and performance of intrusion detection system using deep Bi-Directional LSTM

MSc Research Project
MSC. Cyber Security

Saifullah Sheikh
Student ID: X19216815

School of Computing
National College of Ireland

Supervisor: Imran Khan

National College of Ireland

MSc Project Submission Sheet

School of Computing

Student Name:	Saifullah Sheikh
Student ID:	X19216815
Programme:	MSc. In Cyber Security **Year:** 2020-2021
Module:	Internship
Lecturer:	Imran Khan
Submission Due Date:	16-08-2021
Project Title:	**Improve the detection accuracy and performance of intrusion detection system using deep Bi-Directional LSTM.**
Word Count:	5144 words **Page Count:** 20

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.
ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature: ……Saifullah Sheikh……

Date: ……16-08-2021……

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST

Attach a completed copy of this sheet to each project (including multiple copies)	□
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	□
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	□

Assignments that are submitted to the Programme Coordinator Office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

Improve the detection accuracy and performance of intrusion detection system using deep Bi-Directional LSTM

Saifullah Sheikh
X19216815

Abstract

Intrusion detection systems are used to monitor the network for anomalies to prevent hostile attacks on the entire network. Many firms are now having NIDS problems, which causes large quantities of false alerts when hostile activities are found and IDS raises their alarm, however, we remain mystified because the complicated environment is not especially flexible. The IDS performance was slower or dropped by considerable amounts of false alarms, which makes this sensitive duty difficult, and the management of overall network intrusion detection is costlier due to the large computational effort. The same and key properties of the entire network and computer security have been studied extensively. Intrusion classification into UNSW-NB15 by classifying network difficulties by updating and installing new, efficient technologies that may be readily categorized into a UNSW-NB15 dataset intrusion identifier. A new Deep Bi-directional LSTM strategy will be proposed to build an updated detection model to minimize or accuracy of the false alarm rate.

Keywords: IDS, Deep Bi-directional LSTM; Neural Networking, Cyberspace Security.

1 Introduction

A network suspicious activity monitoring system (IDS) is used extensively to defend the network from intruders and external invaders. This strong detection system helps prevent unwanted entry, vulnerable information leakage, and destruction in a computer network. On the contrary, the Internet of Things devices, Internet apps, and computer networks might be considered as an elite solution that helps in addressing security breaches, and essential protection solutions. In several areas, such as industrial or smart cities and health surveillance, IDS is extensively employed. Continuing cybersecurity research has given rise to various new options, including the capacity to take control of the system and operate it remotely. In addition, safety at the industry level has now become the main priority. There have been numerous newly developed and updated programmes and techniques for improving the security of the domain, for example, computers, IoT devices, and networks [1]. Intrusion detection is the only tool to detect suspicious activity when IDS identifies suspicious traffic in a network that alerts the user of possible attacks [2]. The intrusion detection system tool not only functions as a detection tool but has a monitoring function to monitor network traffic and provides important timeline information. The goal of the system network is to build efficient and efficient models for use in IDS frameworks that are based on neural network technology. Each model, on the other hand, makes precise decisions from a variety of designs, which may limit the model's ability to accomplish the main goal.

Example: In the Internet of Things and a system network, some neural Networks-based IDSs fail to recognize or attempt to apply the models on certain inconsistent data sets due to a lack of essential engineering, which harms accuracy, computer time, memory sizes, and system performance. As well as on antiquated or unconnected data sets that do not accurately represent the internet of things in real-time, network traffic, and system traffic, certain intrusion-detection systems are evaluated on outmoded or unconnected data sets. The UNSW-NB15 is used for implementing the IDS in the model provided, using the neuronal Bidirectional-Long Short-Term Memory network. The current dataset named UNSW-NB15[3] has 81 recent features for all new attacks that are conceivable, including malware and remote access Trojans (RAT).
The intrusion detection system based on anomalies is the focus of cybersecurity research. As an intrusion detection system, the most advanced tool is used in the company for network protection by scanning the network and evaluating each of the traffic to check whether the network is normal or anomalous to protect the system from probable intrusion attacks. We have chosen deep bidirectional LSTM for the current data set UNSW-NB15, which has close to 81 characteristics to attain a minimum precision of 0.91-0.98% and a false positive rate decrease. It also provides a simpler, more precise, and quantifiable approach to address all types of intrusion attacks directly. Based on this research, we have a question to investigate IDS with UNSW- NB 15.

How can we improve the accuracy and reduce the false-positive rate in IDS using a novel method of bi-directional-LSTM using the most recent dataset UNSW-NB15, to detect potentially innovative attacks on the system?

It is consisting seven different chapters. In these seven chapters, the study structure has been defined.
In chapter 1, It introduces and defined the general overview of the study, research topic question, motivation of study on the research topic, and the aim of the research study. In chapter 2, It examines the literature selected for the study. A literature analysis is also conducted in the field of study of previous scholars. In chapter 3 & 4, We outline the method and provide a good explanation of the research methodology, in this section design of the research, flow diagram explanation, statically available tools that will be employed are well described in this section. In chapter 5, We discuss the implementation of the model and in chapter 6 & 7 we focus on the evaluation and discussion of the model.

2 Related Work

As machine learning technology, particularly deep learning technology, advances, a fast-expanding number of academics are seeking to compare deep learning to intrusion detection systems [4]. To achieve high outcomes, deep learning employs a complex neural network topology as well as iterative arithmetic to find the optimal weights. Several studies have extensively used deep learning and machine learning-based to detect intrusions, with significantly better results than traditional methods. Malware monitoring [5-7] and intrusion detection [8-10] are two applications where deep learning has been used in cybersecurity.

This section provides a high-level overview of intrusion detection systems (IDS) for deep learning-based Internet of Things networks

The internet classification framework for the Things network based on deep neural networks (DNN) is being proposed by Tama and K. H. Rhee [1]. Three modern benchmark sets were used to test the proposed framework: GPRS, CIDDS001, and UNSW-NB15. The data variance crisis also shows the disparity of the CIDDS-001 dataset, which is based on the fact that the principal class distributions of the CDDS-001 data set are substantially less than that of the additional class. The Internet of Things systems, which is centered on anomaly extraction and detection, has a unique intrusion detection approach according to Fu et al. [11]. Researchers suggest that anomalies can be identified through the Internet examination of data patterns of the sensor layer of objects, such as humidity or temperature, or anything that could be captured by an IoT device sensor. In data mining, an unattended algorithm is used to find standard patterns.
After being evaluated on the Intel Lab Project dataset, the proposed technique was compared to the expected system, but no differences in precision were discovered. The researchers could not discriminate between DNN and other machine learning approaches employed in their studies in terms of performance. Roy has just published research in which he used the BRR algorithm to investigate and explain the efficacy of DL algorithms for IoT devices. The system has been trained on UNSW-NB14 and has been able to reliably classify incursion with 97 percent accuracy [3].
The paper proposed an Internet-based technology to identify DDoS/DOS attacks on the Internet of Things, by H. Hindy, E. Hodo, E. Bayne, et al (IoT). Normal patterns and threat patterns were used in conjunction with each other to determine the identification. The ANN framework was evaluated on a simulated Things Internet network and proved more than 99 percent of the time to be reliable. It was developed to measure a broad range of attacks and has been consistent in true and false-positive results [12].

To identify the intrusion detection in conjunction with Gradient Descent Optimization, Kim. J used the long-short-term memory strategy to achieve a total record of 97.54 percent and 98.95 percent [13]. First collected intrusion detection data and a reminder, false alarm and accuracy values were 97.06 percent, 10.01 percent, and 98.65 percent respectively [14], for Gated Recurrent Units. Staudemeyer, R.C., assessed the accomplishments of the KDD 99 Cup IDS LSTM Network but significantly increased its findings by 2 22.13% and 93.82% respectively to improve network cost training and network accuracy [15].
The BoT-IoT data set for the assessment of RNN and LSTM was developed by Koroniotis et al. [16]. To verify that there is a balanced data range [0,1], the correspondence coefficient between the data set and the normalized function must be calculated. According to Aldhaheri et al.,[17] DeepDCA is an ID Self-Normalizer Network that incorporates the Dendritic cell algorithm (DCA) (NSN). The information acquisition approach is used to obtain the BoT-IoT data collection package's feature set. When the authors announced their results using a consistent data set, they left out the strategy for balancing the algorithm to improve it. They preferred the loss function because it is more efficient to improve the weights of deep learning layers.

To assign four associated deep study classifications, Roopak et al [19] used the Convoluted Neural Network + Long-Teams Memory, MLP, LSTM, and Id-CNN models, as well as the Convolutional Neural Network + Long-Short-Team Memory, MLP, LSTM, and Id-CNN models, respectively. They have even created the dataset as a replica kit. Although the manner of balance is not specified, this time it is evident. Overfitting is addressed by using layers such as drop-out and maximum grouping, among other things. The BIPT method utilized by Ferrag and Maglaras[18] to distinguish between two Bot IoT datasets and two non-IoT datasets was the BIPT method. Before loading the features into RNN-BPTT they have to be controlled. Roopak et al. [19] used data from CICIDS-2017 to show the efficiency of a sequential planning technique that combines both long short-staff memory, CNN memory, and long short-staff memory. Different objective planning strategies, generally called NSGA [20], were utilized to maximize the number of features. They have a maximum pooling layer between the LSTM and CNN layers in their model to avoid overfitting.

Roy and Cheung [21] developed a method for detecting Internet of Things attacks by employing a bidirectional long-term memory recurring IoT attack detection neural network. The researchers used the data set UNSW-NB15 to construct a binary classifier, which they then applied to their system as part of their research. According to the researchers, their model was able to detect assaults with a 95 percent accuracy rate. Ayo et al. [22] proposed a NIDS-based deep learning model. The architecture is organized into three phases: hybrid characteristics selection, rule evaluation, and detection (or detection and evaluation). The results of experiments showed that the proposed system exceeded previous systems, for example in terms of accuracy, false alarm rate reductions, training times, and test time. A wrapper-based feature removal algorithm was utilized in a wireless intrusion detection system Feed-Forward Deep Neural Network (FFDNN) to demonstrate that it can be first performed in real-world circumstances (WFEU). The efficiency of the WFEU-FFDNN is established by analyzing data from UNSW-NB15 and AWID databases [22].

The exploratory inquiry is looking into many attack types, including binary and multi-class attacks. Choudhary and Kesswani [23] pioneered the development of a powerful neural IDS that is currently in use. They tested their approach using datasets from the UNSW-NB15, the NSL-KDD, and 99 datasets from the KDD Cup. They received positive results. They could obtain an accuracy of greater than 90 percent for each dataset. According to Khoa et al. [24], the Internet of Things has built an innovative collaborative learning-based intrusion detection system. Their goal was to include sophisticated "filters" in the system at stake that may be used on IoT gates to recognize cyber threats rapidly and prevent them. The system under consideration is now being developed. The findings of the experiments demonstrate the efficacy of the technique that has been proposed.

In short, intrusion detection methods are extremely beneficial in IoT networks. The current UNSW-NB15 security-related data set from B IoT, on the other hand, was not used in any research to analyze the efficiency of the IDS model, and so none was reported.

A. Intrusion detection System (IDS)

Intrusion detection is used for distinguishing between various kinds of attacks in computer networks. It's a technique to measure computer system efficiency. Otherwise, interference is the illegal acquisition, without their approval or consent, of access to or control of the

property of another person. Monitoring, identification, and reaction are three essential safety phenomena generated by intrusion detection. The initial stage in the identification of an incursion is monitoring. The main objective of a network intrusion detection system is to identify threats both inside and outside the network. We can presume that a hardware component of an IDS is generally present. This system [25] is also used to perform hardware component applications. A system of intrusion detection serves the same functions, as a security guard. There are two opposing hypotheses in the field of intruder detection: 1) computer systems govern incidents involving a user and devices and 2) common actions of intruders and intrusions [26,27].

B. Recurrent Neural Network (RNN)

An RNN is a feedback network that allows the use of several observations in machine learning applications that allow knowledge to transfer from the past to the present [35] [36]. The stock of information in RNN models is kept open at all times via several pathways or loops. Because the secret layers of the RNN are utilized to store information, the computer memory is comparable to that of the computer memory. Recurring neural networks (RNNs), in computer science, are networks that have the potential to process a succession of activity items via memories within their DNNs and loops [27].

C. Long Short-Term Memory (LSTM)

RNNs for long short memory was increased (LSTM). LSTM uses the concept of gates for its modules. Because they are unable to absorb context information over a lengthy period, the failure to acquire contextual knowledge, including long periods, is one of the most critical challenges for RNNs. When two events are protracted, the problem of disappearing gradients happens (i.e., time from when input is obtained to the time when the input is used to make a prediction). This means that RNNs cannot learn from dependencies that extend over significant distances [27]. One way of tackling this difficulty is to use an LSTM architecture [27]. It prevents the problem of the vanishing gradient from being preserved for a longer period in the background.

D. Bi-Directional LSTM

The bidirectional LSTM (BLSTM) is derived from bidirectional RNN [28] that uses two opaque layers to manipulate input sequences both in front and back. Bi-directional LSTM [29] predicts or marks each element's sequence by employing a limited sequence based on past and potential histories of the elements. This is because two left-to-right and right-to-left LSTMs are operating simultaneously. Composite output is a target signal estimate.

3 Research Methodology

The UNSW-NB15 data set was used in the suggested context as a benchmark. This area provides an integrated rules-based detection (ADR) and false alarm rate improvement model (FAR). The diagram for the task proposed is divided into the following, as seen in Fig.1. It

proposes the IDS Model, which begins by providing an overview of the UNSW-NB15, then moves to pre-processing and then to propose an optimized rule-based model, which is then checked using a reference benchmark and divided into the train and tested, so that the trained set is classified as Deep B-Directional LSTM Neural Network.

A. Proposed Model: -

Fig.1. Flow Diagram

B. Research Architecture: -

The following stages should be used as a guide to implement the design. The methods the model works are as follows:

1. **Data load:** Load Dataset for UnSw-NB15 Instruction Detection.
2. **Pre-processing**: In this stage., the possible insufficiency and noise in the dataset will be deleted
3. **Features Selection**: Based on the correlation methodologies, the best data set features in respect to the target variables is picked.
4. **Train and Test split:** The complete dataset for testing purposes is to be dumped into the train model.
5. **Classification:** The trained set is categorized by means of a neural Bi-Directional LSTM network in this phase

6. **Trained Model**: the model is then trained and saved.
7. **Prediction Model:** Use the trained model to make the prediction.
8. **Analysis of performance**: Performance measurements shall at least be undertaken. Calculation of metrics such as precision, recall and FI score.

C. Model Architecture: -

As it can be observed from the design shown in figure 2, the proposed model comprises two LSTM layers with intermediate form and batch normalization layers. The objective is to share its features, spatial layout, and local perceptions by using a maximum pooling layer. The Sharing of Parameters enables the use of less processed features to extract a smaller set of parameters and free variables. The spatial structure allows for the development of a sparse matrix of features that have been shown to recognize the relationship between attributes more effectively. Local perception also allows for a smaller number of parameters and reduces the training period significantly. With a max-pooling layer of the Bi LSTM layer that can select the two main advantages of quick training time and prevention through the sample of distinct parameters. A batch normalization layer after Max Pooling is used to equalize parameters between intermediate levels, to avoid short training periods. For the pair of batch normalization layers, Reshape Layers reshape the earlier level output.

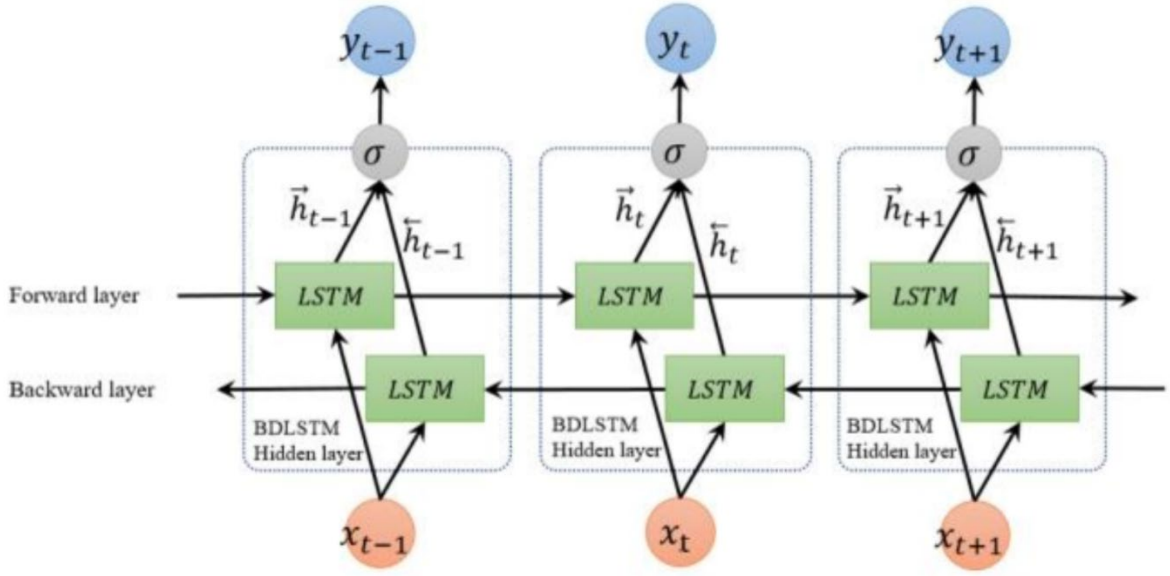


Figure 2: Architecture of the Bi-LSTM layers.

Bi-LSTM layers are used for the reverse and forward time series data training by using two units of the same input and output from the same set of training. One unit handles a time series, while a second unit handles the other time series. As its name suggests, this so-called approach seeks to provide future layer data to reduce training time and increase long-term data accuracy.

To increase the kernel size by 2 factors per iteration the two Bi-LSTM layers of the model are programmed. The first Bi-LSTM layer, beginning at 64 units, moves to a 128-unit block which shows the next and last Bi-LSTM layers. The rationale is because the usage of soil grain is emulated in finely seeded learning so that long-term quality relationships can be more clearly understood and work out quicker. Between each Bi-LSTM layer, max-pooling layers are used to reject the less significant features, and batch standards are employed between each Bi-LSTM layer to standardize the previous intermediate layer output data to enhance efficiency and minimize training time. Next to the full linked layer is the output layer and the drop-out layer. Even if the model employs the maximum pooling of all layers, the dropout layer is over fitted.

D. Research Methodology: -

There are two sorts of research evaluations: qualitative and quantitative. There contains both quantitative and qualitative data. This is a qualitative study. Qualitative data is obtained by observation and judgment such as Tama and K. H. Rhee [1] and Roy and Cheung [3]. An intrusion detection system and deep Bidirectional LSTM Neural Network experiments were compared. These scores are compared for each data set. We suggest a design and technique based on comparative research. These are exploratory, explanatory, and descriptive studies. This article's research method is explanatory and descriptive. This study used explanatory and descriptive analyses because it was based on prior research and the performance of IDs with a different dataset. Comparing the efficacy and efficiency of different deep learning algorithms in isolation is not fair. There are several reasons for this. First, there is a wide variation in the datasets and dataset parts used. To compare deep learning models, many research studies using a single computing system are required.

4 Design Specification

The intrusion detection system design flows are classified by a majority in three parts, and are as follows;

Data preprocessing: The data set used with UnSw-NB15 and data pre-processing in bidirectional LSTM is carried out using CSV file retrieved. The functionality of the data set is processed using the Python numerical framework and the jupyter notebook. This processing removes from the data set all null values and the reduction record. The arrays which are the main input for the LSTM layers are then converted to the matrix. In this process it will be prepared.

Modelling: This is the significant stage of implementation, during which the neural networking method with bidirectional LSTM layers is carried out. This is accomplished using the Tensorflow and Keras frameworks. Graphics driver is used in conjunction with the juypter frame work notebook to improve efficiency and reduce processing time. The overall procedure is carried out using both GPU

and CPU, utilizing the entire system resource and providing the evaluation metrics. Accuracy, FI-score, recall, and precision.

Visualization: For simple interpretation by the reader, the metrics obtained in the second phase are converted as graphs and a confusion matrix into visual representation.

Hardware and Software Requirement-

Dataset	UNSW-NB15 (BOT)
Language	Python
Computer Machine	High-performance Gaming Laptop
RAM	12 GB
Software	Python / Google Colaboratory
Function	Relu activation
Training Set (256 GB)	Keras, Tensorflow

5 Implementation

This part describes the datasets used, the implementation of the proposed system, the experimental environment, and the additional results gained, which are all provided and explored in further detail.

A. Framework:

In order to evaluate the performance of our data network threats similar to real-world network data, we used UNSW 2018(advance version of UNSW 15), the latest information set on our suggested technique. [30]. In this study, we introduce two-way LSTM (Bi-LSTM) model [31]. This section discusses the Siamese neural network layers of the

proposed model together with the datasets and the pre-processing processes. The data in this data set is changed so that it was more equitable in terms of attacks and normal data, the number of instances was limited to 81 characteristics, and data are divided into training and test data.

B. UNSW-NB15 Dataset:

The University of New South Wales compiled this data collection in 2015. UNSW dataset and supplies have been used since it was launched. There is a wide variety of malware families represented in the UNSWNB15 dataset, as well as a large number of extracted functions, a large number of IP addresses for testing, and data collection. This data collection is a combination of modern and up-to-date traffic attacks by the network. Table 1 and figure 3 show a list of features accessible via UNSW-NB15 data sets [34].

Table 2: UNSW-NB15 Dataset Attack Categories

Category	Count
Normal	55000
Analysis	2500
Backdoor	2000
DoS	16353
Exploit	44525
Fuzzers	1800
Generic	40000
Reconnaissance	10000
Shellcode	10500
Worms	0000
Total	133178

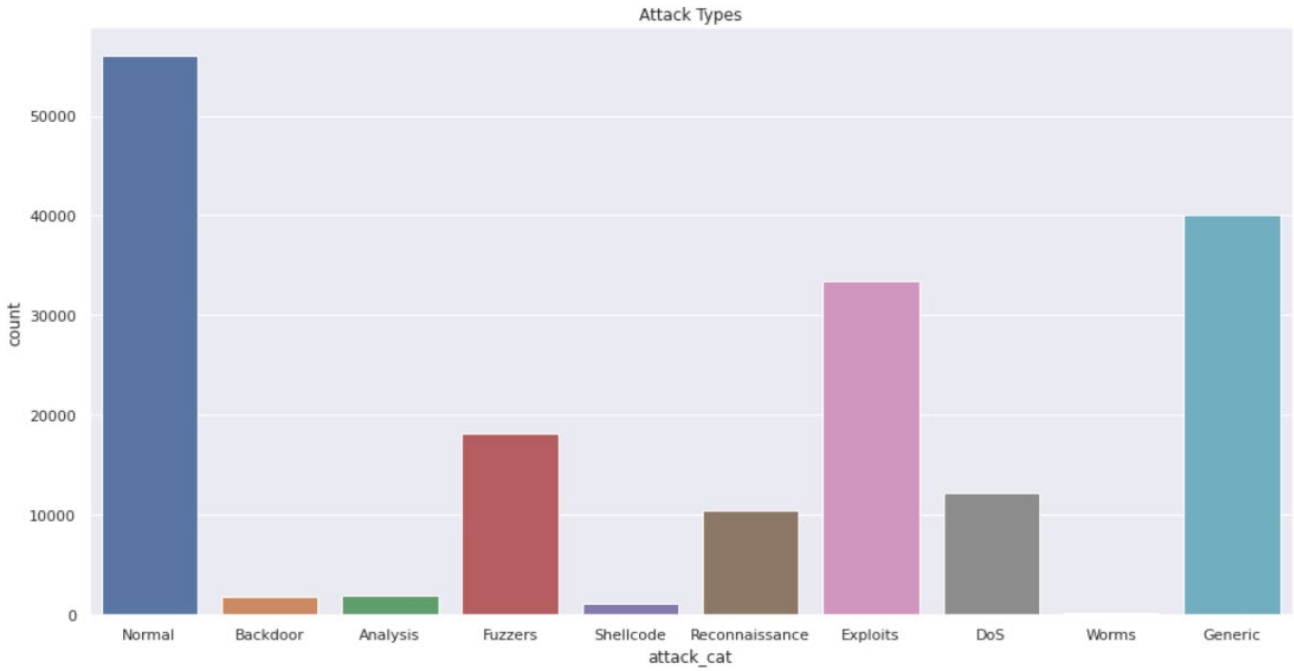


Figure 3: UNSW-NB15 Dataset Attack Categories

C. Pre-Processing –

Data sets are frequently pre-processed by standardizing the numeric properties and a hot encoding of categorical UNSW-NB15 data set features. UNSW-NB15, on the other hand, has an abnormally small number of records in categories such as Worms, Fuzzers, and so on, as demonstrated above. To address this issue, the over-sampling approach was implemented in the training set to ensure that each category of attack received a comparable quantity of data [32] [33].

One hot encoding: The UNSW-NB15 provides categorical features, which must be transformed into numerical values for our deep study model to deliver effective predicted results. This has resulted in the columns in the pre-processing section being converted to numerical values using the dummy's function from the panda python library's dummy function. Because only one hot encoding is selected for the label encoder, and because the label encoder generates several numbers in the same column, the model may misinterpret the values, which will have an impact on the data categorization.

Normalization: Normalization scales data to a certain range to avoid duplication and improve the model training time. Min-Max Normalization in the paper is used and the range of data is resized to [0,1].

$$X[i] = \frac{X[i] - Xmin}{Xmax - Xmin} \quad \text{.....eq1}$$

6 Evaluation

A. Evaluation Metrics –

Evaluation metrics are used for the evaluation of model performance, performance measurements comprising accuracy (ACC), detection rate (DR) and false-positive rate (FPR), F1 and the operational function area of the ROC-AUC receptor. Accuracy and DR are two measures evaluating a model's ability to foresee all sorts of attacks. This is especially relevant in conjunction with DR and ACC, as it calculates the fraction of typical assault records. The model may not be advantageous when the FPR is high even with strong DR and ACC. The F1 scoring provides an efficient performance evaluation, because precision and memory alone cannot provide a clear overview of the entire performance level. Equations (1), (2)(3) and (4). provide a formulation of the above metrics.

$$\text{Accuracy(ACC)} = \frac{TP+TN}{TP+TN+FP+FN} \ldots\ldots \text{Eqn 1}$$

$$\text{DetectionRate} = \frac{TP}{TP+FN} \ldots\ldots\ldots\ldots \text{Eqn 2}$$

Whereas the number of attacks is accurately classified as TP, the volume of normal traffic is accurately classified as TN, and the number of miss staffed attacks classified as normal traffic is accurately classified as FN but erroneously classified as FP.

$$\text{False PositiveRate} = \frac{FP}{FP+TN} \ldots\ldots\ldots \text{Eqn 3}$$

$$F1 - \text{score} = \frac{2 * \text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}} \ldots\ldots\ldots \text{Eqn 4}$$

After that, the ROC-AUC curve tests whether or not the model is capable of distinguishing between distinct classes in the data set, assuming that the threshold is set differently for each class. The AUC is the area beneath the receiver operating characteristic curve, which can range from 0 to 1 depending on the situation. The greater the AUC, the more accurately the model can classify different classes.

6.1. Experiment 1

The findings of the UNSW-NB15 s data set, which has classes ranging from 0.1 to 0.5, are depicted in the illustration 5. Approximately 0.96 percent of the time is accurate, 0.85 percent of the time is DR, and 56001 of the time is a false positive.

6.2. Experiment 2

The findings of the UNSW-NB15 s data set, which has classes ranging from 0.1 to 0.5, are depicted in the illustration 5. Approximately 0.90 percent of the time is accurate, 0.94 percent of the time is DR, and 116031 of the time is a false positive.

6.3 Experiment 3

The findings of the UNSW-NB15 s data set, which has classes ranging from 0.1 to 0.5, are depicted in the illustration 5. Approximately 0.93 percent of the time is accurate, 0.90percent of the time is DR, and 172032 of the time is a false positive.

	Precision	Recall	FI Score	Accuracy
Experiment 1	0.96	0.77	0.85	0.553
Experiment 2	0.90	O.99	0.94	0.661
Experiment 3	0.93	0.88	0.90	0.799

Figure.4. comparison of scores from different results

Figure 4. above shows that the final results of 0.91% are compared with the test phase of the Bi directional LSTM model. This comparison shows apparent that the model is efficient, given the utmost precision of the Bi- directional LSTM.

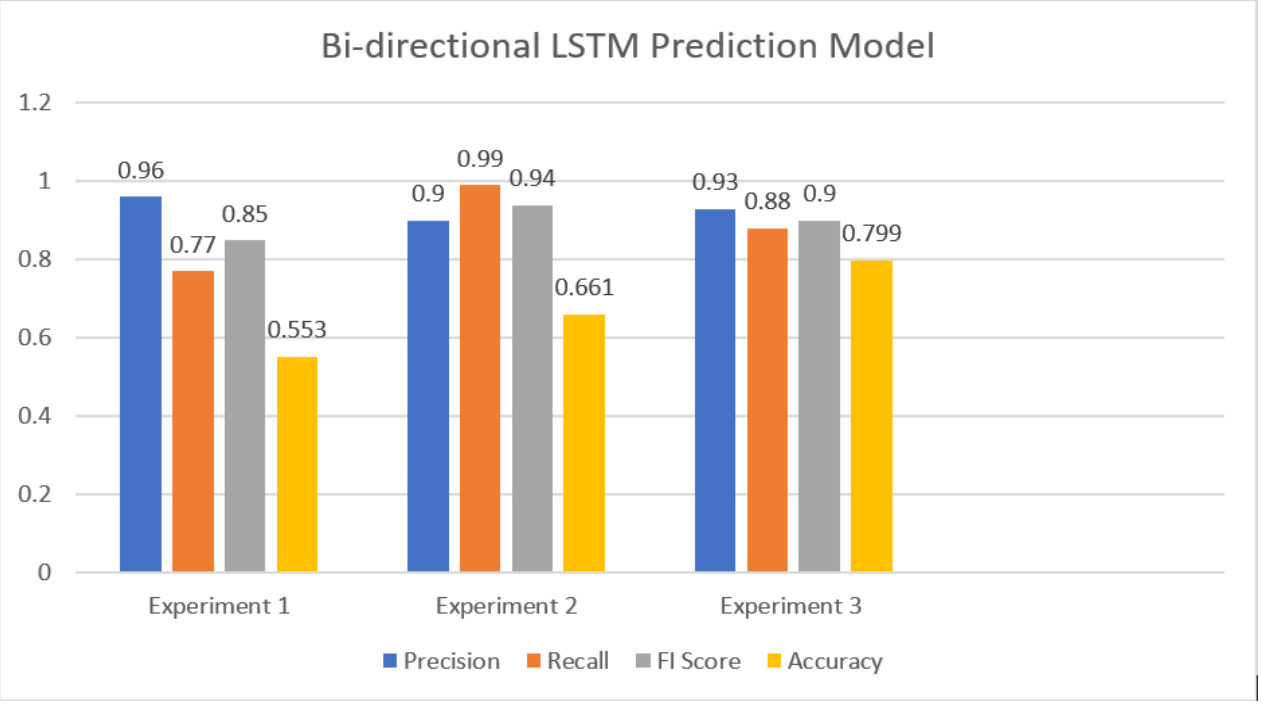


Figure 5: Comparison of Precisions, Recalls, FI Scores, and Accuracies.

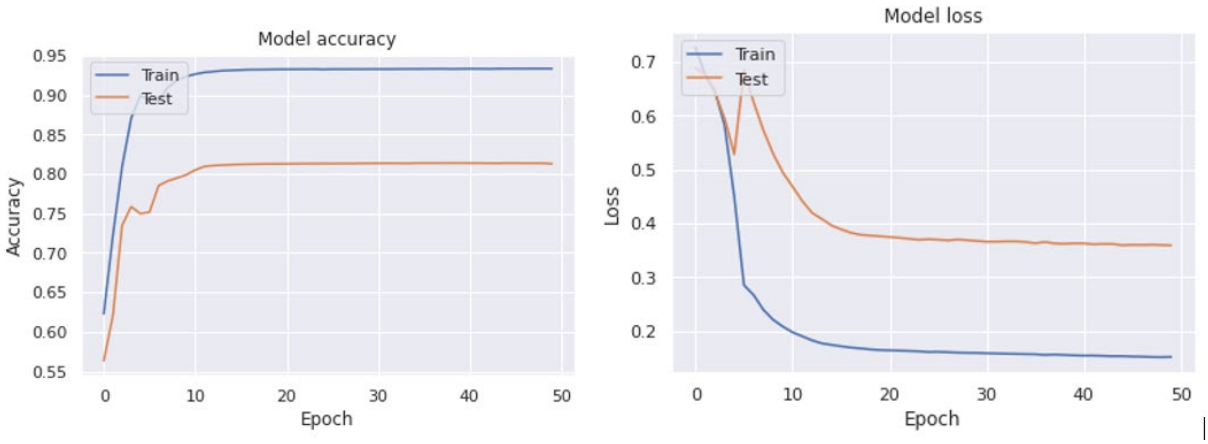
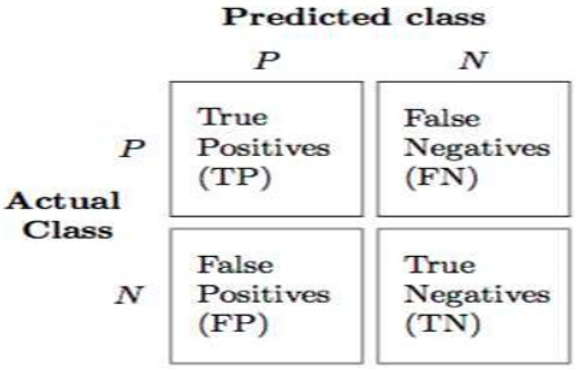


Figure 6: a)Model Accuracy Prediction. b) Model loss Prediction.

Figure 6. shows the accuracy and loss during the training phase of the Bi directional LSTM model. While the precision improved from 90% to 91% and the loss declined from 0.4% to 0.7% in 20 epochs. The increased precision of the loss decrease demonstrates the efficacy of the Bi directional model.



Predicted Classes	Actual Classes	
	Positive	Negative
Normal	42962	13039
Anomaly	1593	114438

Figure 7: Confusion Matrix

Figure 7. Above confusion matrix shows that it is the normal and zero stands for attack and it is our predicted variables. The confusion matrix is a technique for summarizing the performance of a classification algorithm. Classification accuracy alone can be misleading if you have an unequal number of observations in each class or if you have more than two classes in your dataset. Calculating a confusion matrix can give you a better idea of what your classification model is getting right and what types of errors it is making.

To conduct the evaluation, a single LSTM model was used to analyse the data set, and the results show that Bi-directional LSTM is a better and more effective model. The findings of the Bi directional LSTM model for the dataset were 94 percent accurate. This clearly demonstrates that the Bidirectional LSTM model outperforms other intrusion detection algorithms in terms of effectiveness. The confusion matrix from the bi-directional fusion testing phase is depicted in Figure 7, with True Positive (TP) values of 42962, False Negative

(FN) values of 114438, False Positive (FP) values of 1593, and True Negative (TN) values of 13039.

B. Outcomes Performance –

The finding of the average results of UNSW-NB 15 is 0.914, 0.89 percent of the time is DR, and 1593 of the time is a false positive as shown in figure 7. This demonstrates that the model is quite successful and precise when discriminating between different dataset classifications. UNSW-NB15 has produced models such as Adaboost, LSTM, and SVM, among others. As shown in the following table, the proposed model demonstrates increased performance in comparison with the old model across all parameters, including detection rate, false positive, and accuracy. Although the detection rate of the proposed model is higher than that of the closest comparable model.

6.4 Discussion

The model predicts based on the data whether a sample is part of an attack or if it is in the classification of the ordinary class. Under the proposed model, the classification results in the different iteration are depicted. In a specific period, the rate of detection is calculated. The UNSW-NB15 plot has a confidence interval of 0.96 percent, the accuracy (ACC) has a confidence interval of 0.89 percent, and the false-positive rate (FPR percent) has a confidence interval of 114440 percent. The model has a high detection rate (DR%) and a low FPR rate, meaning it has a low FPR rate. It shows that it has a low FPR rate (FPR percent). FIGURE 5 exhibits multiple F1 k scores between 0.1 and o.5 and also the finest F1 values between 0.91548 and dropout layer value is 0.5 in fineness. When k is 0.5 and accurate, respectively, 93 percent precision and 90 percent detection rate are reached. That is because the training model will have more samples of each attack/standard class with growing folds, which will enable the model to better categorize itself. The model has 0.914 percent on average, which is higher than normal. We note that the degree of freedom of the model (DR) increases from 0.1 to 0.5, but it increases more than 0.5, when it runs the code, the data will be lost since it transmits more data to be between 0.1 and 0.5.

7 Conclusion and Future Work

This investigation reveals data that UNSW-NB 15 uses a two-way network. Optimal parameters are identified using the training data set and two two-way LSTM layers are formed in the neural network. High precision and recovery rates were attained when the qualifying network was incorporated in the data set when the normal/attack identification was identified. To compare actual outcomes, the traditional LSTM network was utilized to train and validate data gathering. The study findings suggest that the adoption of a two-way LSTM network significantly increases the effect of detection and contributes to the identification in future of false alarm rates for equipment protection. By comparing previous investigations,

various flaws were identified, including a considerable quantity of data from an unequal classification of data sets, creating challenges with generalization. These challenges need to be overcome and results enhanced in our proposed paradigm.

Moreover, only a small amount of data applies to this model. The data does not use in this model. We can utilize this model as a reference for any problems that arise from the present investigation in the future.

References

1. Rhee, K., Adhi, B. and Rhee, K.-H. (2017). Attack Classification Analysis of IoT Network via Deep Learning Approach Attack Classification Analysis of IoT Network via Deep Learning Approach. (Doi: 10.22667/ReBiCTE.2017.11.15.015).
2. Yin, C., Zhu, Y., Liu, S., Fei, J. and Zhang, H. (2018). *An enhancing framework for botnet detection using generative adversarial networks*. [online] IEEE Xplore. Available at: https://ieeexplore.ieee.org/document/8396200 [Accessed 01 Aug. 2021].
3. Roy, B. and Cheung, H. (2018). *A Deep Learning Approach for Intrusion Detection in Internet of Things using Bi-Directional Long Short-Term Memory Recurrent Neural Network*. [online] IEEE Xplore. Available at: https://ieeexplore.ieee.org/document/8615294/ [Accessed 10 Jun. 2020].
4. Yang, S. (2019). *Research on Network Behavior Anomaly Analysis Based on Bidirectional LSTM*. [online] IEEE Xplore. Available at: https://ieeexplore.ieee.org/document/8729475?denied= [Accessed 29 May 2021].
5. Kim, T., Kang, B., Rho, M., Sezer, S. and Im, E.G. (2019). A Multimodal Deep Learning Method for Android Malware Detection Using Various Features. *IEEE Transactions on Information Forensics and Security*, [online] 14(3), pp.773–788. Available at: https://ieeexplore.ieee.org/abstract/document/8443370/ [Accessed 12 Aug. 2020].
6. Karbab, E.B., Debbabi, M., Derhab, A. and Mouheb, D. (2018). MalDozer: Automatic framework for android malware detection using deep learning. *Digital Investigation*, 24, pp.S48–S59.
7. Hou, S., Saas, A., Chen, L. and Ye, Y. (2016). *Deep4MalDroid: A Deep Learning Framework for Android Malware Detection Based on Linux Kernel System Call Graphs*. [online] IEEE Xplore. Available at: https://ieeexplore.ieee.org/abstract/document/7814490/authors#authors [Accessed 14 Jun. 2021].
8. Kasongo, S.M. and Sun, Y. (2020). A deep learning method with wrapper based feature extraction for wireless intrusion detection system. *Computers & Security*, 92, p.101752.
9. Khan, F.A., Gumaei, A., Derhab, A. and Hussain, A. (2019). TSDL: A Two-Stage Deep Learning Model for Efficient Network Intrusion Detection. *IEEE Access*, 7, pp.30373–30385.
10. Zhipeng, L. and Qin, Z. (2019). *Intrusion Detection Using Temporal Convolutional Networks*. [online] Available:https://www.researchgate.net/publication/337803242_Intrusion_Detection_Using_Temporal_Convolutional_Networks [Accessed 3 Aug. 2021].
11. Fu, R., Zheng, K., Zhang, D. and Yang, Y. (2011). *An intrusion detection scheme based on*

anomaly mining in internet of things. [online] IEEE Xplore. Available at: https://ieeexplore.ieee.org/abstract/document/6197881.

12. Hanan Hindy. (2020). Cryptography and Security (cs.CR); Artificial Intelligence (cs.AI); Networking and Internet Architecture. 2020 IEEE. 10.1109/ACCESS.2020.3000179.

13. Kim, T., Kang, B., Rho, M., Sezer, S. and Im, E.G. (2019). A Multimodal Deep Learning Method for Android Malware Detection Using Various Features. *IEEE Transactions on Information Forensics and Security*, [online] 14(3), pp.773–788. Available at: https://ieeexplore.ieee.org/abstract/document/8443370/.

14. Jun, K., Lee, D.-W., Lee, K., Lee, S. and Kim, M.S. (2020). Feature Extraction Using an RNN Autoencoder for Skeleton-Based Abnormal Gait Recognition. *IEEE Access*, [online] 8, pp.19196–19207. Available at: https://ieeexplore.ieee.org/abstract/document/8963659 [Accessed 20 Jul. 2021].

15. Staudemeyer, R.C. and Omlin, C.W. (2013). Evaluating performance of long short-term memory recurrent neural networks on intrusion detection data. *Proceedings of the South African Institute for Computer Scientists and Information Technologists Conference on - SAICSIT '13*.

16. Koroniotis, N., Moustafa, N., Sitnikova, E. and Turnbull, B. (2019). Towards the development of realistic botnet dataset in the Internet of Things for network forensic analytics: Bot-IoT dataset. *Future Generation Computer Systems*, 100, pp.779–796.

17. Aldhaheri, S., Alghazzawi, D., Cheng, L., Alzahrani, B. and Al-Barakati, A. (2020). DeepDCA: Novel Network-Based Detection of IoT Attacks Using Artificial Immune System. *Applied Sciences*, 10(6), p.1909.

18. Ferrag, M.A. and Maglaras, L. (2019). DeepCoin: A Novel Deep Learning and Blockchain-Based Energy Exchange Framework for Smart Grids. *IEEE Transactions on Engineering Management*, pp.1–13.

19. Roopak, M., Tian, G.Y. and Chambers, J. (2020). *An Intrusion Detection System Against DDoS Attacks in IoT Networks*. [online] IEEE Xplore. Available at: https://ieeexplore.ieee.org/abstract/document/9031206.

20. Kumar, M. and Guria, C. (2017). The elitist non-dominated sorting genetic algorithm with inheritance (i-NSGA-II) and its jumping gene adaptations for multi-objective optimization. *Information Sciences*, 382-383, pp.15–37.

21. Roy, B. and Cheung, H. (2018). *A Deep Learning Approach for Intrusion Detection in Internet of Things using Bi-Directional Long Short-Term Memory Recurrent Neural Network*. [online] IEEE Xplore. Available at: https://ieeexplore.ieee.org/document/8615294/.

22. Ayo, F.E., Folorunso, S.O., Abayomi-Alli, A.A., Adekunle, A.O. and Awotunde, J.B. (2020). Network intrusion detection based on deep learning model optimized with rule-based hybrid feature selection. *Information Security Journal: A Global Perspective*, 29(6), pp.267–283.

23. Choudhary, S. and Kesswani, N. (2020). Analysis of KDD-Cup'99, NSL-KDD and UNSW-NB15 Datasets using Deep Learning in IoT. *Procedia Computer Science*, 167, pp.1561–1573.

24. Khoa, T.V., Saputra, Y.M., Hoang, D.T., Trung, N.L., Nguyen, D., Ha, N.V. and Dutkiewicz, E. (2020). *Collaborative Learning Model for Cyberattack Detection Systems in IoT Industry 4.0*. [online] IEEE Xplore. Available at:

https://ieeexplore.ieee.org/abstract/document/9120761.

25. Cui, Z., Ke, R., Pu, Z. and Wang, Y. (2020). Stacked bidirectional and unidirectional LSTM recurrent neural network for forecasting network-wide traffic state with missing values. *Transportation Research Part C: Emerging Technologies*, 118, p.102674.

26. Hayashi, T., Watanabe, S., Toda, T., Tori, T., Roux, J.L. and Takeda, K. (2016). Convolutional bidirectional long short-term memory hidden Markov model hybrid system for polyphonic sound event detection. *The Journal of the Acoustical Society of America*, [online] 140(4), pp.3404–3404. Available at: http://dcase.community/documents/challenge2016/technical_reports/DCASE2016_Hayashi_2006.pdf.

27. Schuster, M. and Paliwal, K.K. (1997). Bidirectional recurrent neural networks. *IEEE Transactions on Signal Processing*, 45(11), pp.2673–2681.

28. Qingqing Zhang, Hongbian Yang, Kai Li and Qian Zhang (2010). Research on the intrusion detection technology with hybrid model. *2010 The 2nd Conference on Environmental Science and Information Application Technology*.

29. Thaseen, I.S. and Kumar, Ch.Aswani. (2014). *Intrusion detection model using fusion of PCA and optimized SVM*. [online] IEEE Xplore. Available at: https://ieeexplore.ieee.org/abstract/document/7019692.

30. Graves, A. and Schmidhuber, J. (2005). Framewise phoneme classification with bidirectional LSTM and other neural network architectures. *Neural Networks*, [online] 18(5-6), pp.602–610. Available at: https://www.sciencedirect.com/science/article/pii/S0893608005001206 [Accessed 18 Apr. 2019].

31. Kumar, V., Sinha, D., Das, A.K., Pandey, S.C. and Goswami, R.T. (2019). An integrated rule based intrusion detection system: analysis on UNSW-NB15 data set and the real time online dataset. *Cluster Computing*.

32. Schuster, M. and Paliwal, K.K. (1997). Bidirectional recurrent neural networks. *IEEE Transactions on Signal Processing*, 45(11), pp.2673–2681.

33. Schuster, M. and Paliwal, K.K. (1997). Bidirectional recurrent neural networks. *IEEE Transactions on Signal Processing*, 45(11), pp.2673–2681.

34. Moustafa, N. and Slay, J. (2015). *UNSW-NB15: a comprehensive data set for network intrusion detection systems (UNSW-NB15 network data set)*. [online] IEEE Xplore. Available at: https://ieeexplore.ieee.org/document/7348942.

35. Staudemeyer, R.C. and Omlin, C.W. (2013). Evaluating performance of long short-term memory recurrent neural networks on intrusion detection data. *Proceedings of the South African Institute for Computer Scientists and Information Technologists Conference on - SAICSIT '13*.

36. Anani, W. and Samarabandu, J. (2018). *Comparison of Recurrent Neural Network Algorithms for Intrusion Detection Based on Predicting Packet Sequences*. [online] IEEE Xplore. Available at: https://ieeexplore.ieee.org/abstract/document/8447793 [Accessed 16 Jun. 2021].