IJNRD.ORG ISSN : 2456-4184
INTERNATIONAL JOURNAL OF NOVEL RESEARCH AND DEVELOPMENT (IJNRD) | IJNRD.ORG
An International Open Access, Peer-reviewed, Refereed Journal

A DEEP LEARNING FRAMEWORK FOR CYBER SECURITY ATTACKS DETECTION USING CNN- RNN-BI-LSTM ALGORITHMS USING JUPYTER NOTEBOOK

[1]Ms.Rubina.G.Kureshi, [2]Dr.R.K.Dhuware
[1]Research Scholar, [2]Head,Department Of Computer Science
[1]Department of Electronics and Computer Science,RTMNU,Nagpur,
[2]Dhote Bandhu Science College,Gondia

Abstract : Cybersecurity threats have increased in complexity, necessitating advanced threat detection mechanisms. Traditional security solutions often fail to detect sophisticated cyber attacks, making deep learning techniques an important means in modern cybersecurity frameworks. In this paper we presents an experimental approach using hybrid deep learning model CNN- Long Short-Term Memory (LSTM)-Recurrent Neural Network (RNN) to classify cybersecurity attacks types. The proposed approach preprocesses network traffic data, encodes categorical attributes, and applies normalization methods to ensure ideal model performance. The model's performance is evaluated using key classification metrics, including accuracy, precision, recall, F1-score, and confusion matrix analysis. The findings determine the possibility of bi- LSTM networks in detecting cyber threats efficiently, flagging the way for their integration into real-time security monitoring systems.

IndexTerms **- Cyber Security,Deep Learning, CNN-LSTM-RNN,Cybersecurity attacks.**

1.INTRODUCTION

As digital infrastructure has grown, cyberthreats have evolved to be more complicated and difficult to identify with traditional security methods. Deep learning-based intrusion detection systems (IDS) have demonstrated encouraging outcomes in spotting network traffic irregularities. Using its sequential learning ability to identify patterns in network data, this finding inspects a deep learning method based on LSTM for cybersecurity threat detection. The development of effective and intelligent attack detection schemes is necessary due to the surge in cybersecurity risks brought on by the quick expansion of digital infrastructure. Because they are static, traditional cybersecurity tools like firewalls and rule-based intrusion detection systems (IDS) frequently miss complex cyberthreats. Techniques for machine learning (ML) have become a viable way to detect cybersecurity attacks. This study uses a hybrid deep learning methods that unites Long Short-Term Memory (LSTM), Convolutional Neural Networks (CNN), and Recurrent Neural Networks (RNN) to construct a deep learning-based cybersecurity attack detection system. To create an effective classification model for attack prediction, the study makes use of a structured dataset that includes network traffic data, packet attributes, and anomaly scores.

Even before the global pandemic, the world was already shifting towards digitalization. The pandemic only accelerated it. Ever heard of cloud computing and budget-friendly access to the internet? Ports, hospitals, financial systems, and reservation systems all fall under 'Critical National Infrastructure' or CNI. And that means that these establishments are already digitized. In order to safeguard the data and other digital assets, advanced Intrusion Detection Systems, or IDS, must be used (Anderson and Khan,

1980 and 2021). Speaking of digital safety, Jim Anderson is quite a famous name, all thanks to his work on intrusion detection systems. Current designs have evolved to address modern security challenges (Debar et al, 1999).

As noted by Mohan et al. (2022), an intrusion breaches confidentiality, integrity, and availability (CIA). An intruder detection system (IDS) is such a device that observes the traffic of a computer network for any novelties that can be harmful or dangerous(Vinayakumar et al., 2019). This identifies intrusions as difference from the normal legitimate activity. This part of the defensive system is one of the latter ones, but it makes the system's resistance against the attacks stronger by cooperating with other measures of security like, identification and encryption processes. Researching about intrusion detection has been a theme of discussion within the cyber world for a while now. Most significantly, existing Cyber Security measures are becoming less and less effective due to the increased rate of data processing since they are primarily dependent on signature and rule-based systems (Dong and Wang, 2016).

Given that these methodologies are limited to a database of known attack patterns, they are unlikely to be useful in coping with dynamically changing and adaptable attack environments. The application of artificial intelligence (AI) has grown in scope in a number of systems and it is not different in cyber security. Specifically, several Deep Learning (DL) models have been developed to enhance IDS and they have been quite effective at spotting malevolent conduct (Ferrag et al., 2020). Even so, the increased vulnerability to cyber attacks demands smarter solutions. In conjunction with IDS in big data contexts, conventional methods of machine learning which are set at a distance by their low levels of sophistication suffer from limitations.

They're struggling to keep up with the latest attacks and effectively implement timely defences due to the overwhelming noise present in massive datasets, which makes it difficult for the algorithms to handle. The machine learning (ML) algorithms used in intrusion detection systems (IDSs) primarily focus on feature engineering to extract valuable data from the network.

Unlike traditional methods deep learning-based intrusion detection systems don't need feature engineering. Their complex structure lets them pick up complex features and pull out high-quality representations from raw input. Deep learning models learn useful features from network data, which helps them tag an activity as normal or not based on patterns they've picked up. These models are known to be data-hungry needing huge amounts of information to spot patterns and sort traffic . However, company network data has stayed closed off and private for security reasons. Now, with more training data on hand deep learning models can show better accuracy and fewer false alarms.

IDS can be categorized into a number of groups based on the techniques they employ to identify any negative event. Signature Based IDS (S-IDS), Anomaly Based IDS (A-IDS), and Hybrid Inspired IDS (H-IDS) are the three main varieties of intrusion detection systems. A database of predetermined attack signatures is compared to network activity in order for S-IDS to function; the signature database is updated continuously by human input. A-IDS, on the other hand, uses heuristic algorithms to infer previously unknown harmful behaviors from network data in order to detect and learn patterns suggestive of possible threats or unauthorized intrusions. In most cases, the false positive rate for A-IDS is high. (Mishra & colleagues, 2019). For commercial use, the majority of organizations have used a hybrid method to spotlight this issue.

Convolutional Neural Network-LSTM (Convolutional Neural Network-Long Short-Term Memory), ANN-Artificial Neural Network, LSTM-Long Short-Term Memory, BiLSTM-Bidirectional Long Short-Term Memory, and BiGRU-Bidirectional Gated Recurrent Unit models were trained on the preprocessed data using univariate or recursive feature elimination techniques. CNN-LSTM, the best-performing suggested model, achieved accuracy close to zero using the most recent IDS on the NSL-KDD dataset with fewer features. Historically, network intrusion detection systems have faced a number of noteworthy constraints that impact their functionality and usefulness. Feature redundancy and irrelevance are among the main problems; conventional approaches frequently use datasets with a large number of redundant or irrelevant features, which can result in overfitting and ineffective model training.

Moreover, class imbalance, in which there are much fewer attack samples than benign ones, is a problem for many IDS techniques. This disparity distorts performance indicators and hinders the model's capacity to identify infrequent but critical attacks. Using out-of-date datasets, like the KDD Cup 1999 dataset, presents another difficulty because they might not adequately represent modern network settings and new attack vectors, leading to models that are unable to adjust to evolving

threats. Moreover, high-dimensional feature spaces and complex models can exacerbate overfitting, a phenomena where models perform well on training data but badly on unknown data, which reduces the models' applicability in real-world situations.

Traditional techniques can be passive and demanding in terms of processing power leading to performance bottlenecks, making them a challenge for real-time implementation of Intrusion Detection Systems. An efficient and scalable solution to this issue is provided by our study which uses a common dataset,i.e.,the NSL-KDD dataset, which provides a more relevant and balanced feature set than previous datasets.

To reduce the dimensionality of data and avoid overfitting, we proposed a new method based on Recursive Feature Elimination (RFE) in this work. By focusing on the maxims beneficial and educative highlight, RFE upgrades model generally execution and decreases superfluous computational heap.

We proposed a sophisticated hybrid deep learning model incorporating Long Short-Term Memory, Recurrent Neural Networks, and Convolutional Neural Networks.

The proposed method uses advanced deep learning models, specifically the CNN-RNN-LSTM model. This hybrid model therefore combines CNNs, which are efficient in extracting features, with LSTMs for finding patterns in a time sequence, solving the problems of overfitting and computational complexity, and ensuring better scalability.

Not just improves detection accuracy, but also as a more effective and flexible solution for contemporary network security issues, the CNN-RNN-LSTM model overcomes the shortcomings of conventional IDS techniques. The planned study has been well arranged as follows: Section 2 presents a literature review of the relevant works related to the proposed theme. A detailed description of the system design, operation, statistical indicators, flow chart, and algorithm is provided in Section 3 Materials and Methods. All the results and findings are in section 4. In the section 5 Proposed study Conclusion and future scope of it discussed.

2.Related Work:

As technology progresses in this era of growing digitization, protection systems from cyberattacks becomes more and more critical. Because cyberattacks on critical infrastructure are becoming more sophisticated, it is imperative to improve cyber intrusion detection systems (IDS). This study proposes and evaluates a deep learning-based system for hack detection using the Cybersecurity Attacks dataset. The system pre-processes data using a Decision Tree classifier and Recursive Feature Elimination (RFE) to identify the most essential attributes in order to optimize model performance. Many deep learning models, including ANN, LSTM, BiLSTM, CNNLSTM, GRU, and BiGRU, have been evaluated.

CNN-LSTM performed better than the others, with a f1-score of 0.94, a recall of 0.89, and an accuracy of 95%. These outcomes demonstrate how well the suggested IDS can differentiate between malicious and benign network traffic. To further improve IDS performance, forthcoming studies can examine collective approaches like boosting or bagging.

New techniques for creating DL intrusion detection systems and strategies for enhancing the model's performance in follow-up investigations have been offered by a sizable body of academic literature. According to a thorough review of the literature, LSTM and deep neural networks (DNN) are the models that are most frequently studied.

A CNN with weight-dropper LSTM (WDL-STM) hybrid model was presented by researchers in (Hassan et al., 2020) as an intrusion detection system that may be used in a Big Data setting. The suggested method avoided overfitting on recurrent connections and preserved long-term relationships between derived features by combining CNN and WDLSTM to extract valuable features from vast volumes of data. According to reports, the hybrid model achieved 97.1% accuracy using the UNSW-NB15 big dataset. The distinctions between CNN-LSTM and ConvLSTM were initially examined by the authors of (Liu and Patras, 2022). To increase the generalizability of the supervised algorithms, they used a novel data augmentation technique and used an ensemble Bidirectional Asymmetric LSTM (Bi-ALSTM) as the basic ID logic.

The algorithm using the augmentation technique was evaluated on CIC-IDS-2017 (X-eval) and CSE-CIC-IDS-2018; on the latter, it obtained an F1 score of 94.85%. Recent studies have shown that, despite the DL models' encouraging accuracy range, they can be manipulated by intentionally designed inputs, or adversarial examples. Using the Fast Gradient Sign Method (FGSM), the researchers in (Fu et al., 2021) created an adversarial example. We used these examples to evaluate CNN, LSTM,

and GRU models. With an accuracy of 81.92%, CNN was shown to be the most dependable model for hostile scenarios while using the conventional training process. Nevertheless, the robustness of GRU and LSTM to hostile instances considerably improved after adversarial training.

Simple RNN, LSTM, and GRU models for IDS were used in a comparative analysis by the authors of a related research (Guler and Alpay, 2021). All of the models were assessed using the UNSW-NB15 dataset, and they received F1 scores of 0.94, 0.96, and 0.96, respectively. The authors of (Bu and Cho, 2020) proposed a technique that simulates the function of inquiry by combining a classic earning classifier system (LCS) with CNN for IDS based on the RBAC mechanism. When evaluated agai nst a simulated query dataset, it was demonstrated that the combination outperforms alternative machine learning classifiers with an accuracy of almost 92%. Fewer research have systematically combined numerous classifiers using ensemble approaches, which yielded better results.

In (Ahmad et al., 2022), for example, the authors have employed an autoencoder, CNN, and LSTM ensemble to more reliably and efficiently identify large-scale assaults (DDoS, DOS, etc.). Using the ensemble technique, the DoS attack class of the NSL-KDD dataset was identified with a 62.96% accuracy S.S. Bamber et al. Computers & Security 148 (2025) 104146 2 with the BoT-IoT and two other datasets. A related research (Govindarajan and Chandrasekaran, 2011) examines the effectiveness of employing MLP and RBF individual models as well as their ensembles. The dataset utilized in this study was produced utilizing a small number of apps in a specific setting.

Additionally, at the learning percentage and on the latter dataset, the NN model alone achieved 83.6 %, and the ensemble model stood at 92 % accuracy. Using a feature selection technique based on XGBoost., the Authors in (Kasongo, 2023) created an IDS that condenses the feature space of each dataset.Three RNN techniques combined by Researchers as LSTM, simple RNN and GRU. XGBoost-LSTM obtained the best test accuracy of 88.13 %, applying the NSL-KDD dataset for the binary classification issues; for the multiclass classification job, the same model on the dataset had a test accuracy of 86.93 %. In a similar paper using RNNs (Yin et al., 2017), the author's method resulted in 83.28 % accuracy in the binary classification scheme and 81.29 % accuracy in the multiclass classification scheme. This work, make public in 2017, is one of the most cited papers in intrusion detection, with over 970 citations. Numerous findings have used Deep Neural Network (DNN); authors in (Yang et al., 2019) make use of DNN counter to adversarial examples. Implemented model achieved 89 % accuracy utilizing the NSL-KDD dataset.).The dataset UNSW-NB15 ,in a equivalent paper (Khamis et al., 2020 was utilized by the authors to train the DNN counter to adversarial samples using the min-max (also known as saddle point) technique. Additionally, researchers employed Principal Component Analysis (PCA)-based feature reduction., and their trials showed that this helped lower evasion rates. In (Ashiku and Dagli, 2021), on the UNSW-NB15 dataset, the paper provides a DNN that uses CNN with a regularized multi-layer perceptron method, yielding an accuracy of 94.4 %. Furthermore, authors in (Kavitha and Amutha, 2022) employed DNN to attain 91.8 % accuracy on the NSL-KDD dataset. Authors in (Vinayakumar et al., 2019) presented an IDS approach using DNNs on six publicly available datasets; among them, the model achieved an accuracy of 80.1 % with 1 layer of DNN. Furthermore, researchers reported an accuracy of 78.4 % and 96.3 % using the CICIDS2017 and UNSW-NB15 datasets, respectively. Similarly, an improved conditional variational auto encoder (ICVAE) using DNN proposed by Author Yang et al. (2019) , the latter is employed to discover and investigate possible sparse representations between classes and network data characteristics. Using the NSL-KDD and UNSW-NB15 datasets, the ICVAE-DNN model accomplishes 85.97 % and 89.08 % accuracy, respectively. In a similar paper an adversarial IDS based on random neural networks(RNN-ADV) (Qureshi et al., 2020), on random neural networks was proposed by the authors.

The approach known as Jacobian Saliency Outline Assault (JSMA) was utilized to deliver unfriendly occasions, which discover characteristics that can change kind tests as much as conceivable whereas including the slightest sum of unsettling influence. Utilizing the NSL-KDD dataset, the recommended strategy has successfully recognized between typical designs and variations from the norm with an precision of 82.14 %. Creators of (Elmasry et al., 2020) recommended a twofold Molecule Swarm Optimization (PSO)-based strategy for pre-training consequently choosing the model's hyper-parameters and ideal features. This strategy was tried on profound learning models, counting DNN, LSTM-RNN (Long Short-Term Memory Repetitive Neural Arrange), and DBN (Profound Conviction Organize). This ponder is one of the few that utilized the LSTM-RNN approach.

Two datasets, NSL-KDD and CICIDS2017, were utilized to test each demonstrate for double and multiclass classification. In multiclass classification, the models achieved an generally precision of 90.63 %, 93.6 %, and 96.91 %, separately, on the previous dataset and 88.04 %, 92.41 %, and 95.81 %, on the afterward dataset. Analysts recognized that models with pre-training had beated models without pre-training. (Masum and Shahriar, 2020) presented Interruption discovery frameworks utilizing exchange learning (IDS), a valuable strategy for trading data that licenses the utilize of a few state-of-the-art (SOTA) pre-trained models for ID issues by security analysts. The to begin with arrange of TL-NID is a two-phase method that employments VGG-16, pre-trained utilizing the ImageNet dataset, to extricate main part for classification. The recovered features were subjected to a DNN application for classification in the moment arrange.

The proposed strategy gotten a essential 89.3 % precision on the NSL-KDD dataset. In Kunang et al. (2021), the analysts displayed a profound auto-encoder and DNN show pre-training strategy for optimization utilizing hyper-parameters that couples arbitrary and network look. The proposed procedure appeared 86.02 % and 95.38 % multiclass classification exactness based, separately, on the CSE-CIC-IDS2018 and NSL-KDD datasets. In arrange to decrease the dimensionality of the input information and capture both straight and non-linear relationships between highlights, the creators of paper (Thakkar et al., 2024) displayed a fusion-based procedure that coordinating AE, Central Component Examination (PCA), and LSTM approaches. NSL-KDD, UNSW-NB15, CIC-IDS-2017, and MSCAD assessed this proposed strategy, and the comes about were 82.22 %, 76.28 %, 93.78 %, and 99.43 %, separately. Furthermore, the creators of paper (Al-Omar and Trabelsi, 2023) portray an IDS that makes utilize of LSTM and Convolutional Neural Organize (CNN) models that are attention-based. They affirm the vigor and adequacy of our show with a location rate of >95 % utilizing the benchmark dataset UNSW-NB15.

Also The CNN-LSTM model outperformed the others, with 95 % accuracy, 0.89 recall, and 0.94 f1-score. These results prove the effectiveness of the proposed IDS in accurately distinguishing between malicious and benign network traffic.(Sukhvinder Singh Bamber and Aditya Vardhan Reddy Katkuri ,2025)

3.Materials and Methods:

3.1 System Description

The system description is explained in the following section

3.1.1 Dataset Description:

We downloaded dataset from Kaggle named cybersecurity_attacks .This dataset consist of Consists of 40,000 records and 25 varied metrics like Timestamp ,Source IP Address,Destination IP Address,Source Port,Destination Port,Protocol,Packet length,Packet Type,Traffic Type,Payload Data,Malware Indicators,Anomaly Scores,Alerts/Warnings,Attack Type,AttackSignature,ActionTaken,SeverityLevel,UserInformation,DeviceInformation,Netw-ork Segment,Geo-location Data,Proxy Information,Firewall Logs,IDS/IPS Alerts and Log Source.

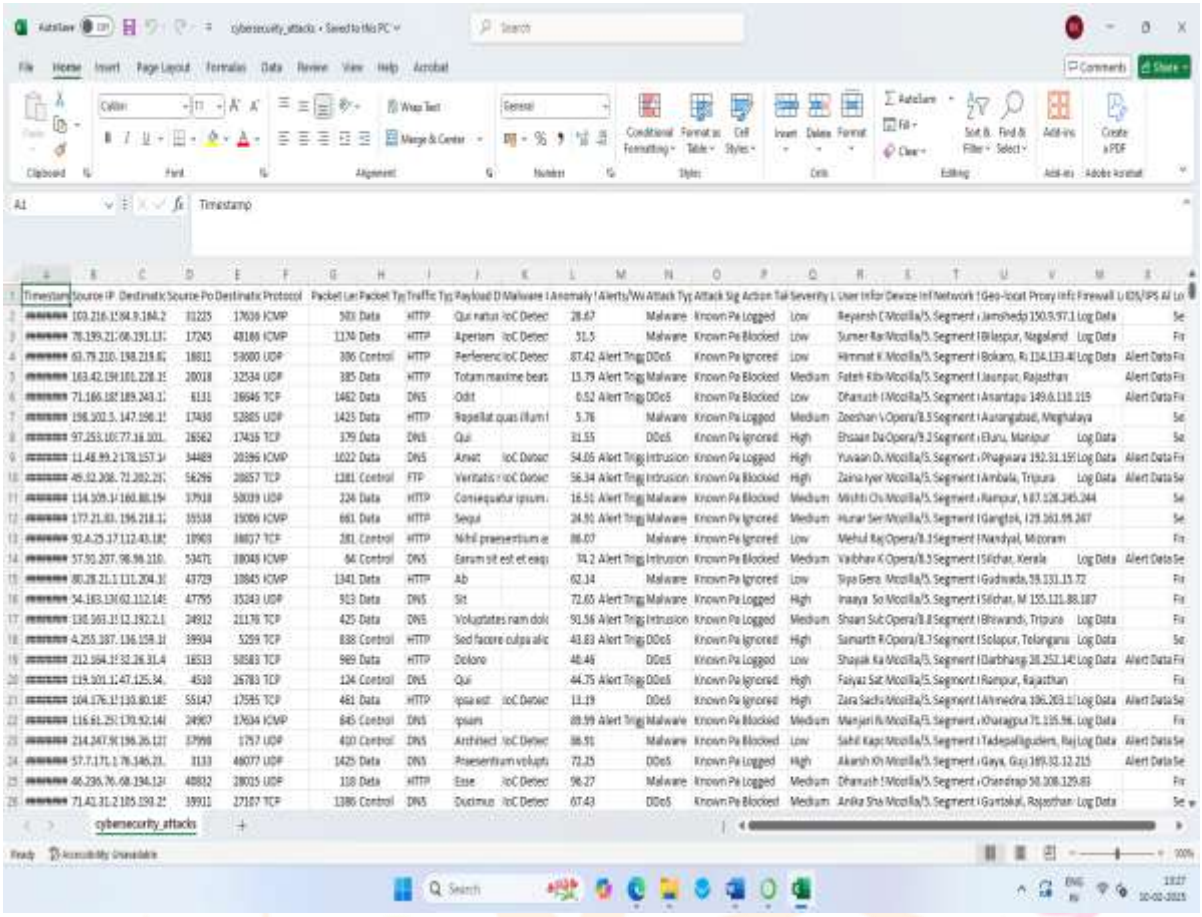

Fig 1.1 Dataset For CyberSecurity Attacks Dataset

We first visualize a cybersecurity Attacks dataset to visualize attack distribution to understand the frequency of each attack type and display missing values.So that we fill missing values and balance the dataset at the preprocessing phase.

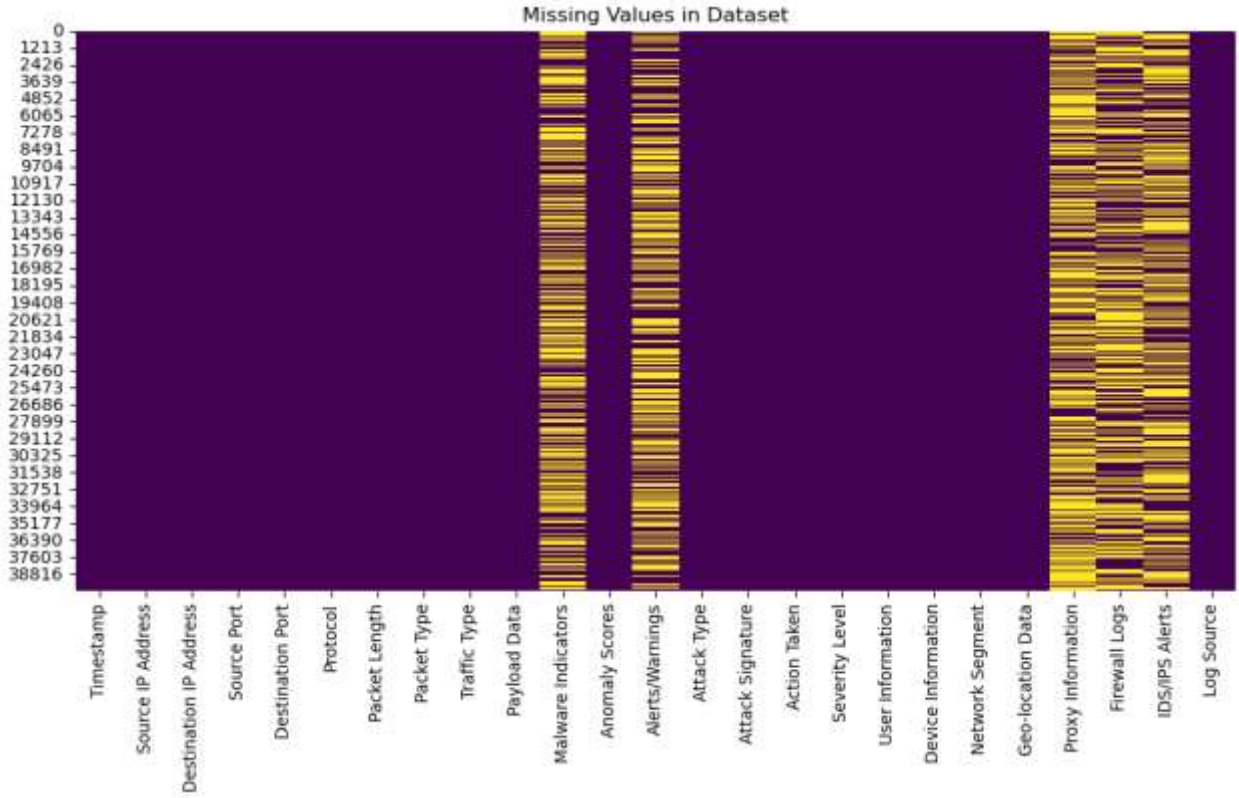

Fig 1.2 Data Visualization before preprocessing

3.1.2Flow Chart:

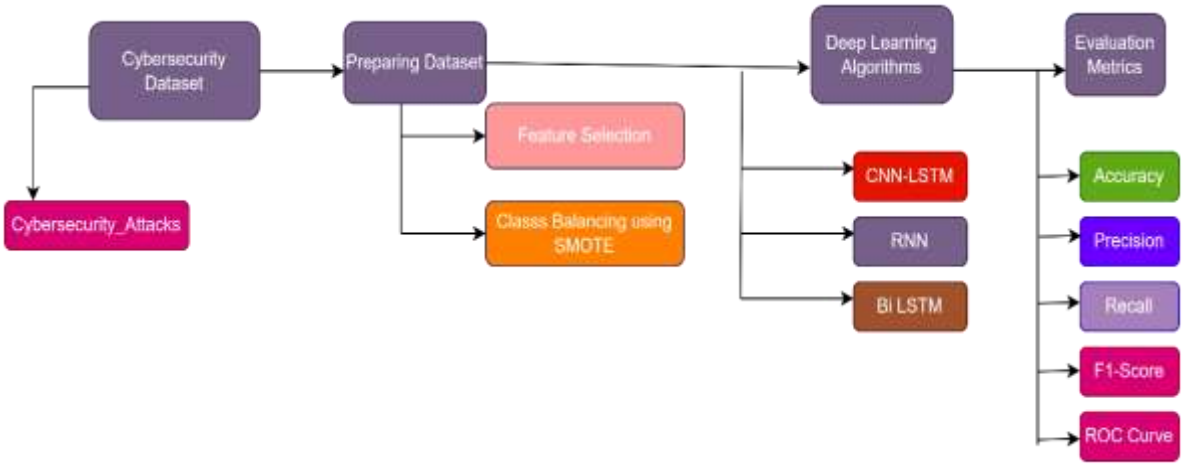

Fig 1.3 Flowchart of proposed Methodology

3.1.3 Proposed Algorithm: Optimized CNN-BiLSTM-RNN for Cybersecurity Attack Prediction

Step 1: Data Preprocessing

1. **Load & Clean Data**: Read the dataset, drop unnecessary columns, handle missing values, and convert timestamps into features.
2. **Encode & Normalize**: Convert categorical features using LabelEncoder, scale numerical data with MinMaxScaler.
3. **Balance Data**: Use **SMOTE** if class imbalance is significant, compute class weights.
4. **Split Data**: Divide into **80% training, 20% test** and reshape for CNN-LSTM input.

Step 2: Build Optimized Deep Learning Model

5. **Define Architecture**:
   - **CNN Layer**: Extract spatial features (Conv1D).
   - **BiLSTM Layer**: Capture bidirectional time dependencies.
   - **RNN Layer**: Improve sequential learning.
   - **Dense Layer**: Fully connected network with ReLU.
   - **Softmax Output** for multi-class classification.
6. **Compile Model**: Use **Adam optimizer**, SparseCategoricalCrossentropy(), and apply class weights.

Step 3: Train & Evaluate Model

7. **Train Model**: Fit with **50 epochs, batch size = 64**, and validate on test data.
8. **Evaluate Performance**: Generate classification report, confusion matrix, and attack prediction distribution.
9. **Compute ROC-AUC**: Plot **ROC curves** for each class.

Step 4: Save & Deploy

10. **Save Model**: Store trained model as cybersecurity_model_optimized.h5.
11. **(Optional) Deploy**: Implement **Flask API** for real-time attack prediction

3.1.4 DL Framework for Cyber Security Attack Detection using Hybrid Model

Python is supported by open source web application .The Jupyter Notebook which moreover underpins other programming language. Python is a free programming language with a basic syntax.Linux ,Windows,Mac OS are stages where python is available. We here use Keras with TensorFlow through the Keras API. TensorFlow's APIs use Keras to allow users to make their own machine-learning models. TensorFlow can help load the data to train the model. TensorFlow Serving can help deploy the model. Keras models come with extra functionality that makes them easy to train, evaluate, load, save, and even train on multiple machines.

As we have proposed a Deep Learning Framework for Cyber Security Attacks Detection following is the evaluation environment and description of our experimental setup listed in table
Evaluation Environment &Description of performed experimental setup:

Table 1:Evaluation Environment &Description

	Parameters or Variables	**Values**
	H/W Specification	
a.	CPU Model name	**Intel(R) Core(TM) i5-10505 CPU @ 3.20GHz 3.19 GHz**
b.	RAM	**8.00 GB (7.72 GB usable)**
c.	System Type	**64-bit operating system, x64-based processor**
d.	Windows Specification -Edition	**Windows 11 Home Single Language**
	Model Evaluation Parameter	
a.	Epoch	**50**
b.	(i)Train -Validation ratio:RNN (ii)Train -Validation ratio:Hybrid CNN-RNN-biLSTM	**8:2**
c.	Optimizers	Adam
d.	Loss Evaluation metrics	SparseCategoricalCrossentropy()

In our cyber security attacks dataset various attacks are there so We group attack types into three categories:

- **0:** High-severity attacks (e.g., DoS, DDoS, SYN Flood).
- **1:** Medium-severity attacks (e.g., Malware, Brute Force, Ransomware).
- **2:** Low-severity attacks (all others).

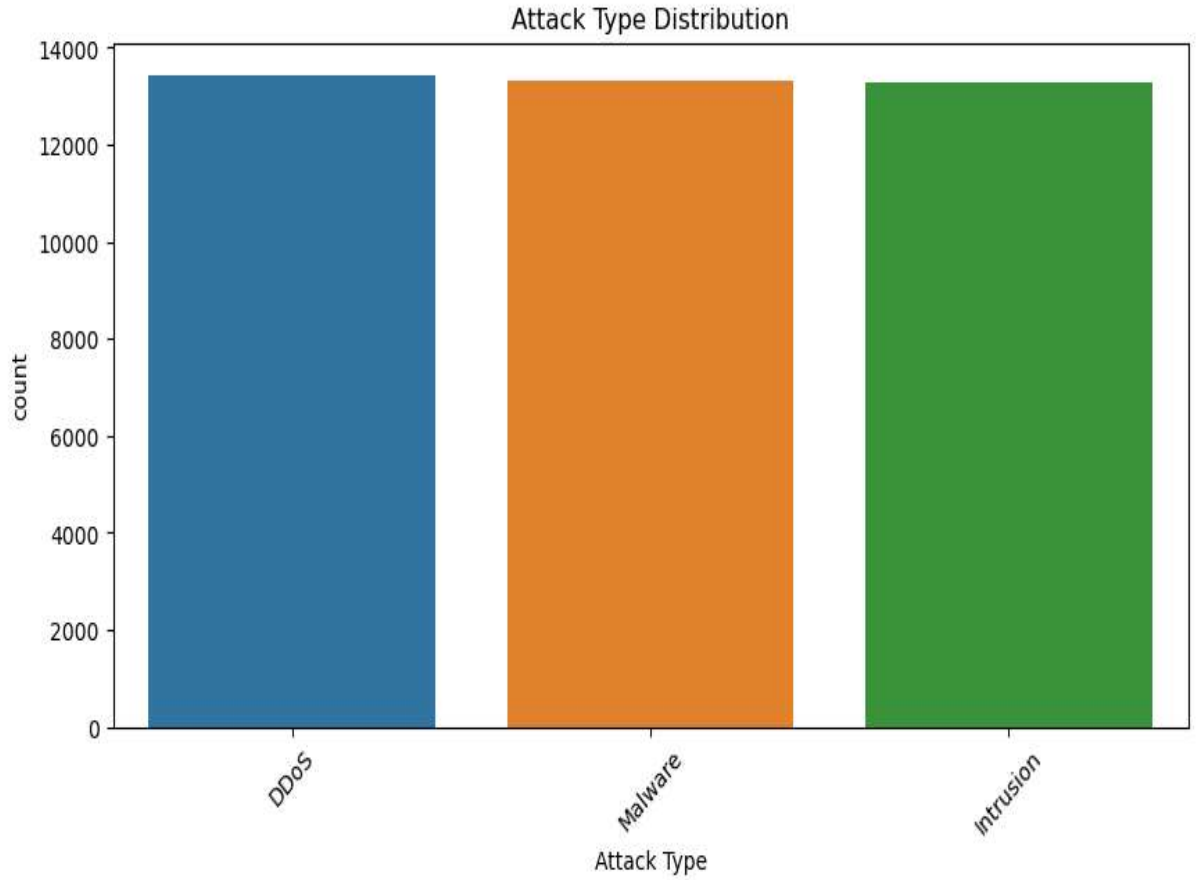

Fig 1.4 Distribution of Cyber Attacks before preprocessing

3.1.5.:Dataset Preprocessing and Denoising and Class Balancing

we first import the dataset from a CSV file into a Pandas Data Frame in our proposed model here. The dataset consist of missing values, raw data also contains categorical and numerical characters and class inequality. Afterwards we perform data preprocessing which consist of multiple steps like drop unnecessary columns that contain personally identifiable information (PII) or are not useful for classification and dropped columns that contain only missing values. Also we converted timestamps into numeric features (hour, day of the week, time elapsed) and Convert categorical features into numerical values. We have normalized the numerical features using Min-Max scaling and handled missing values by imputing the mean.Applied data balancing in this proposed novel hybrid Cybersecurity Attack Detection Model using SMOTE which is more significant in balancing the imbalance classes.For this we have computed class weights to ensure balanced learning.After that we split the dataset and reshape X for compatibility with CNN and LSTM layers.

In this proposed work we have performed two separate experiments based on Deep Learning Algorithms on Cyber Security_Attacks dataset namely RNN Model For Cyber Security Attacks Prediction and Hybrid CNN-RNN-LSTM model for detection of Cyber Security Attacks.Also we have evaluated precision ,recall f1 score ,confusion matrix and roc curve.

3.1.6. Model Architecture

Here is the proposed model architecture consist of following four layes -

a**) Conv1D layer** extracts spatial features.

b) **Bidirectional LSTM** captures sequential dependencies.

c) **Simple RNN** layer adds recurrent learning.

d)**Flatten + Dense** layers prepare data for classification.

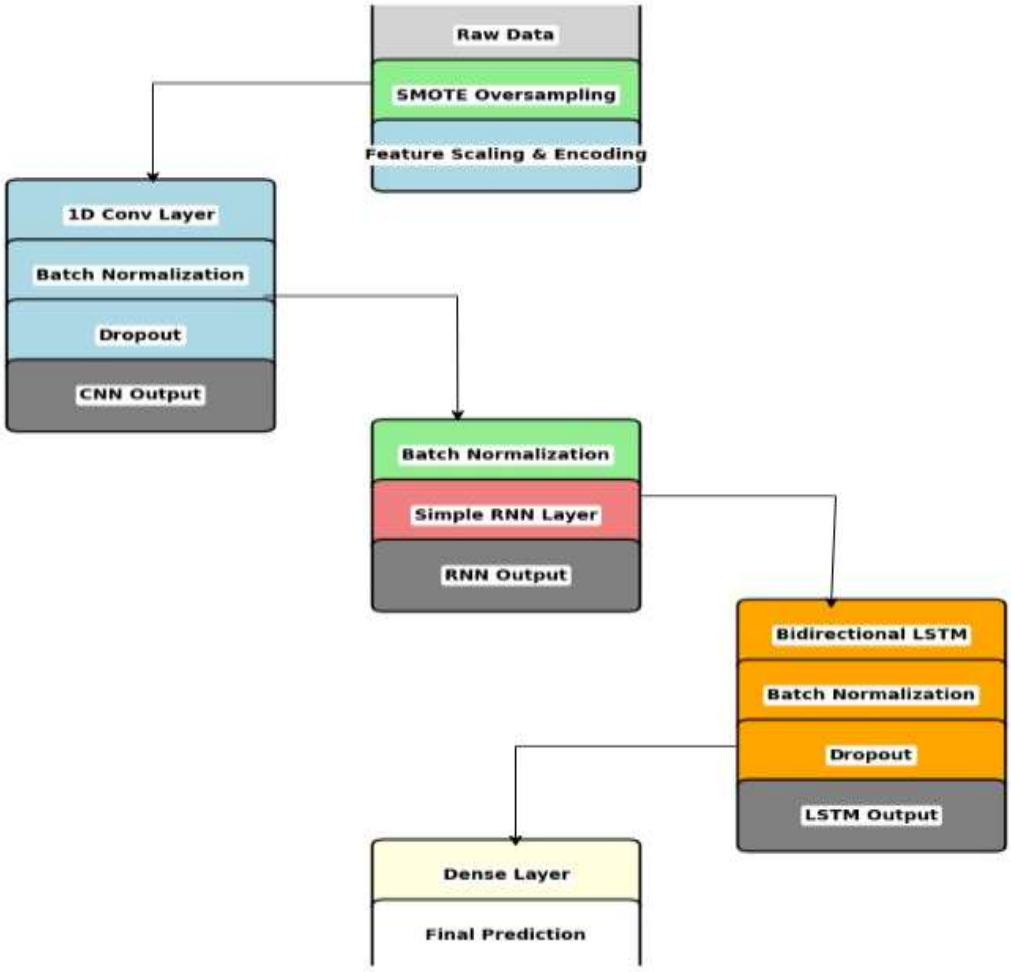

Fig 1.5: Proposed model architecture

In the following code snippet contains the execution of the RNN-LSTM model and hybrid model in Jupyter Notebook

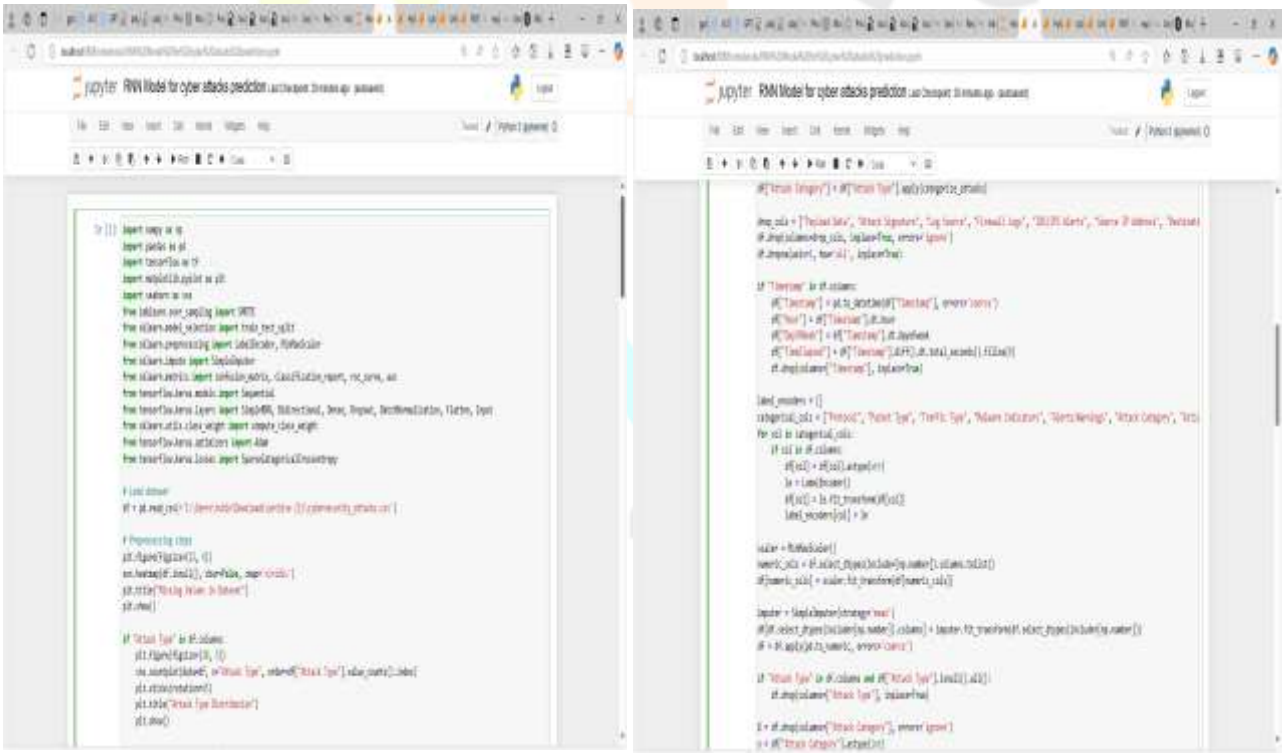

Fig 1.6 Code Snippet of Model : RNN Model for Cyber Attacks Prediction

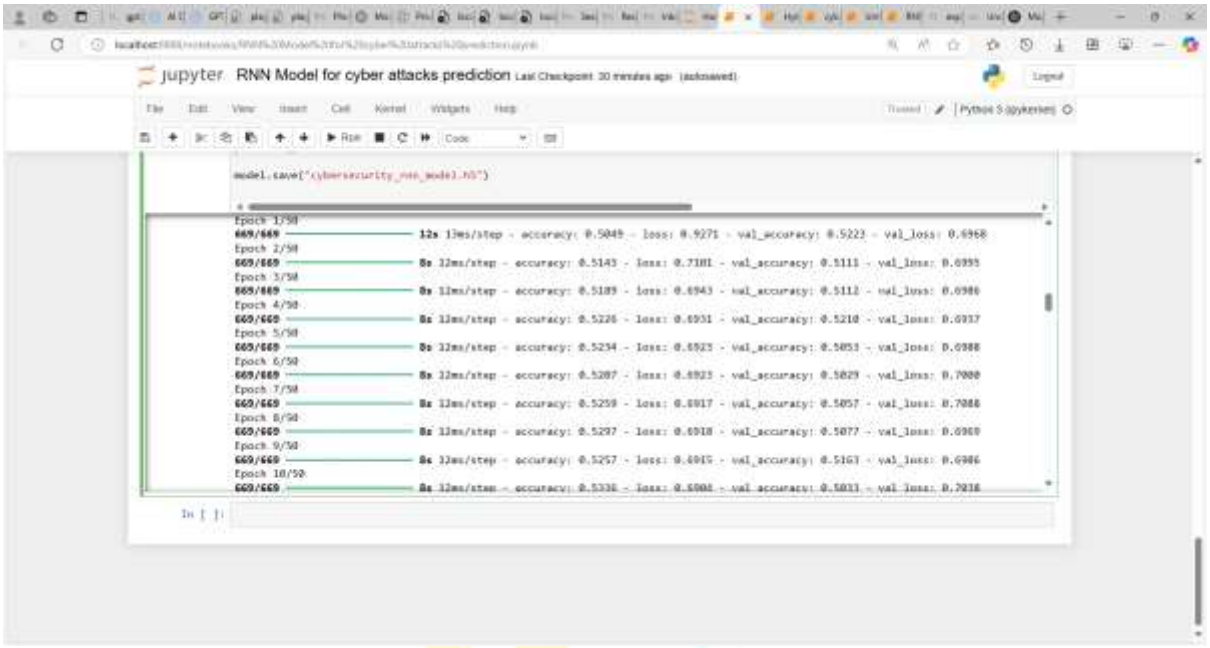

Fig 1.7:Code Snippet showing Training Epoch of Hybrid Model

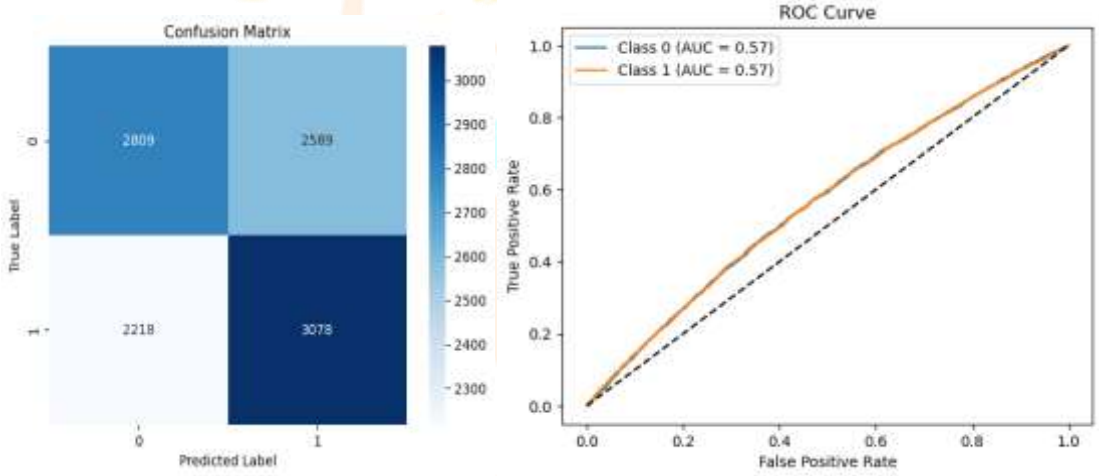

Fig:1.8 Output of RNN Model for Cyber_Security Attacks showing confusion matrix and ROC Curve

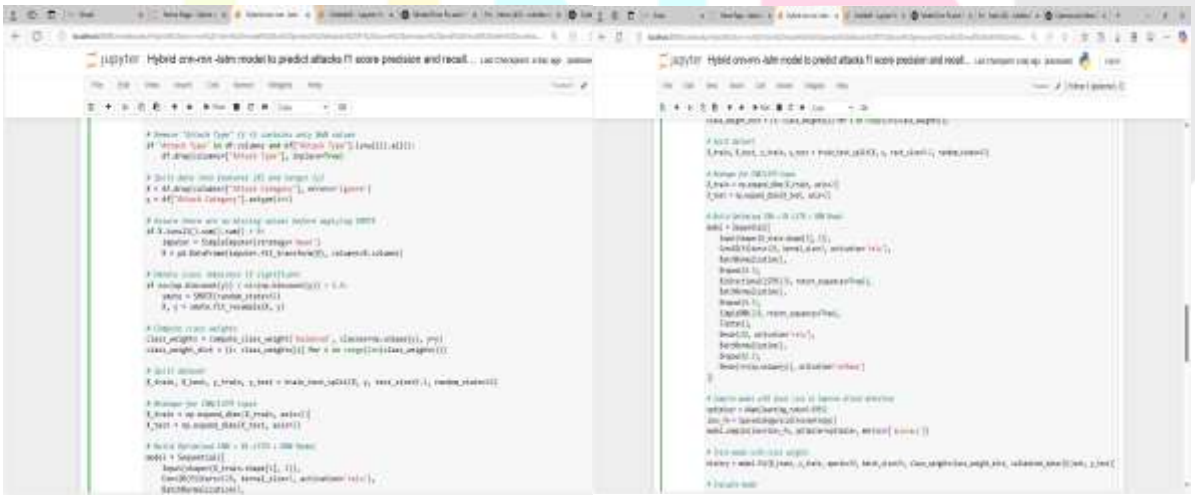

Fig 1.9 Code Snippet of Hybrid Model :CNN-RNN-LSTM Model for Cyber Attacks Prediction

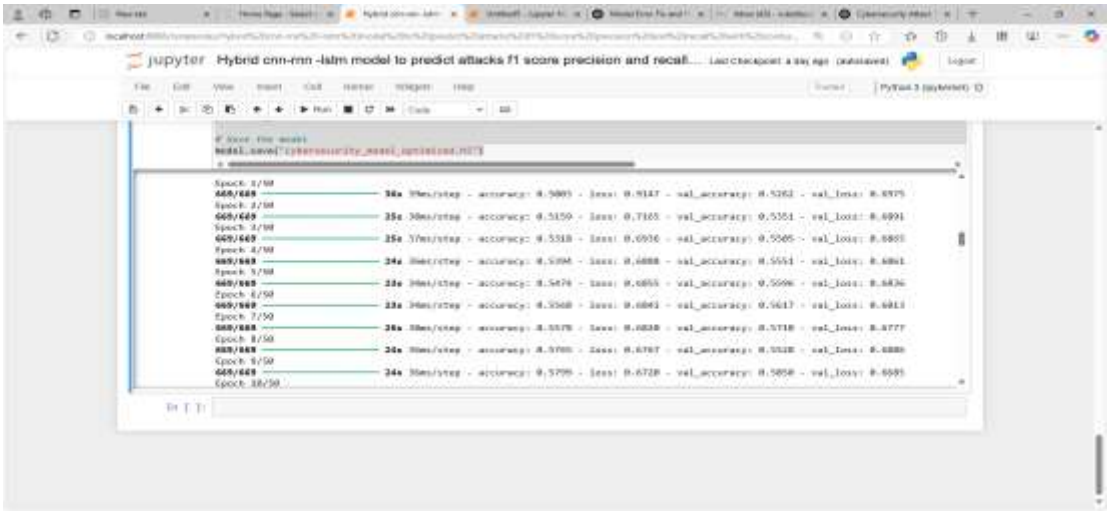

Fig 1.10:Code Snippet showing Training Epoch of Hybrid Model

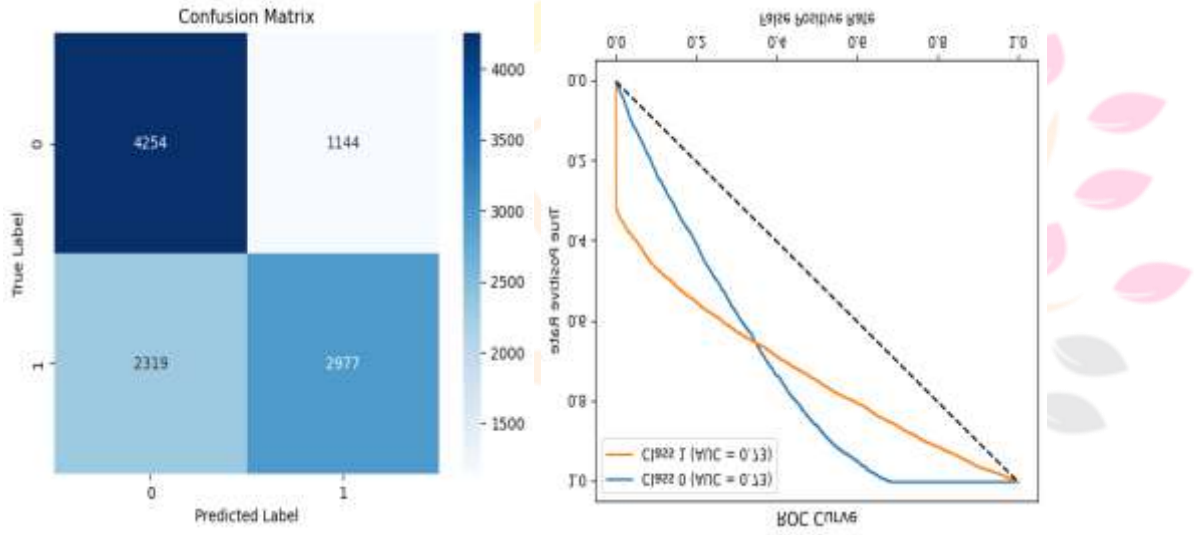

Fig:1.11Output of Hybrid CNN-RNN-LSTM Model for Cyber_Security Attacks showing confusion matrix and ROC Curve

3.1.7 Statistical Metrics:

In this paper, the statistical methods most frequently used in Intrusion Detection Systems research, such as precision, accuracy, recall, f1-score, and ROC curve, are used to evaluate the performance of the well-defined models. The above-mentioned terms relate to the metrics in use. In deep learning designs that judge overall class performance, accuracy is a measure reflected of prime importance. It is defined as the ratio of correctly recognized samples to all samples in the dataset. However, for the reason that it usually predicts the majority class, it can generate artificially high values and thus have limited use in class-imbalance conditions. Statistically, in addition it is defined as:

$$Accuracy = \frac{TP + TN}{TP + TN + FN + FP} \quad (1)$$

Amid the experiments of classification algorithms, it is observed that the fraction of true positive cases classified as positive is measured by precision. In applications , wherever the rate of false positive faults maybe extremely high, similarly to those spam detection or fraud detection, this metric is very significant. It is represented mathematically as follows:

$$Precision = \frac{TP}{TP + FP} \quad (2)$$

Recall is very useful in the case of all applicable cases for detecting threats. This is also referred to as the true positive rate or sensitivity. In other cases, such as with fault detection or medical diagnosis, where it is often desirable to classify a case as positive even if it is not, as a false negative can have catastrophic consequences, a high recall is always preferred over perfection. It is mathematically written as below:

$$Recall = \frac{TP}{TP + FN} \quad (3)$$

The F1 Score is a convenient metric that well balances recall and accuracy. It is specifically supportive when the classes aren't evenly distributed or when the stakes for false positives and false negatives are different. Mathematically, you can express it like this:

$$F1\ Score = \frac{2}{\frac{1}{Precision} + \frac{1}{Recall}} \quad (4)$$

Where true positive is represented by TP, true negative by TN, false positive by FP, and false negative by FN. Receiver Operator wind(ROC) is a visual representation that shows how True Positive Rate and False Positive Rate are traded off at different bracket situations.

$$FPR = \frac{FP}{TN + FP} \quad (5)$$

$$TPR = \frac{TP}{TP + FN} \quad (6)$$

Where FPR is False Positive Rate and TPR is True Positive Rate. The classifiers performance is summed up by a single scalar static called the area under the ROC curve (AUC-ROC). It ranges from 0 to 1 with the interpretations as: 0.9 - 1.0: Excellent performance, 0.8 - 0.9: Good performance, 0.7 - 0.8: Fair performance, 0.6 - 0.7: Poor performance, 0.5 - 0.6: Fail (Fig. 2)

4.Results:

Among the 10694 records of cybersecurity attacks dataset we have observed the following results:

The basic RNN model performed moderately it shows Accuracy is only 55%, meaning the model correctly predicted slightly more than half of the instances, Precision and Recall for both classes (Normal & Attack) are low, indicating that the RNN model struggled to distinguish between normal traffic and attacks, F1-scores are also low, showing a balance between poor precision and recall.55% accuracy is relatively low for cybersecurity models while using the CNN-RNN-Bi-LSTM,the hybrid model combining CNN, RNN, and Bi-LSTM shows a significant enhancement, accuracy increased to 68%, meaning this model correctly predicted nearly 7 out of 10 records,precision for Attack Class (1) improved to 72%, meaning it reduced false positives,recall for Normal Class (0) improved to 79%, indicating better detection of non-attack traffic , F1-Scores are higher in both classes, which shows better overall balance in detecting attacks and normal traffic.

4.1Comparison Summary:

Table 2:Showing Comparison of observations between RNN Model and CNN-RNN-Bi-LSTM model

Observation		RNN	CNN-RNN-BiLSTM (Hybrid)
Detection of Attacks		Weak	Stronger
False Positives		Higher	Reduced
False Negatives		Higher	Reduced
Overall Performance		Poor to Average	Good & improved

Picking the right threshold for recall and F1-score really changes how well a model handles true positives versus false negatives. If you lower the threshold, the model's more likely to catch everything—good for recall—but that might mean it flags stuff it shouldn't. On the other hand, a higher threshold can make it more precise, but then it might miss real threats. It's a tricky balance, especially in something like intrusion detection, where both false alarms and missed attacks can cause problems. In that kind of setup, the RNN-CNN-Bi-LSTM model just seems to make more sense. It not only lines up with what the theory says should work, but also performs well in testing. CNNs, in particular, are solid at pulling out complex patterns in the data. You can actually see the impact in Table 3, which shows how different deep learning models did on the Cyber Security Attacks dataset—no feature elimination used, just raw comparisons.

Table 3:Performance on Deep Learning Models on RFE on Cyber Security_Attacks Dataset

Metric	RNN	CNN-RNN-BiLSTM (Hybrid Model)
Accuracy	55%	68%
Precision (Class 0 - Normal)	56%	65%
Precision (Class 1 - Attack)	54%	72%
Recall (Class 0 - Normal)	52%	79%
Recall (Class 1 - Attack)	58%	56%
F1-Score (Class 0)	54%	71%
F1-Score (Class 1)	58%	63%
Support	5398	5398 (Normal), 5296 (Attack)

Fig. 4. CNN-LSTM Model with RFE confusion matrix. Fig. 5. CNN-LSTM model without RFE ROC plot. S.S. Bamber et al. Computers & Security 148 (2025) 104146 7 their convolutional layers, which is crucial for identifying distinct patterns and anomalies within network traffic. These networks capture local features and hierarchical patterns that are essential for understanding the complex structures in the data. LSTM networks, on the other hand, are specifically designed to handle sequential dependencies, making them ideal for modeling temporal relationships in data. LSTMs address the limitations of traditional RNNs by maintaining long-term memory through their cell states and gating mechanisms, which enables them to remember and learn from long sequences of events—a critical factor in detecting intrusions that span multiple time steps.

After applying the z-test and t-test between the RNN model and the Hybrid CNN-RNN-BiLSTM model, the results confirm a statistically significant improvement in accuracy. The p-value obtained is less than 0.05, which means that the improvement in performance is not due to random chance. Therefore, it can be concluded that the hybrid model provides a reliable and effective solution for real-time cybersecurity attack prediction compared to the basic RNN model.

. The proposed CNN-LSTM model has significant practical implications for real-world cybersecurity scenarios. Its high accuracy and robust performance in detecting network intrusions make it a valuable tool for enhancing cybersecurity systems. In practice, this model can be integrated into existing systems by deploying it as a part of an IDS that continuously monitors

network traffic. The CNN component excels at identifying patterns in network data, while the LSTM component captures temporal dependencies, crucial for detecting sophisticated and evolving threats. Integrating this model involves incorporating it into a real-time processing pipeline where it can analyze network packets, generate alerts for potential intrusions, and provide actionable insights. Moreover, the model's ability to effectively classify and prioritize threats can improve incident response times and reduce false positives, thereby enhancing overall network security and operational efficiency. Additionally, the model can be updated periodically with new data to adapt to emerging threats, ensuring sustained protection against evolving cybersecurity challenges. Additionally, proposed research acknowledges the constraints of our CNN-LSTM model, recognizing possible obstacles in terms of real-time usage because of the model's intricate nature.

5.Conclusion and Future Scope:

This suggested study put forward a cyberattack detection system based on deep learning that is able to distinguish between harmful and innocent network traffic. Based on the confusion matrix, precision, recall, accuracy, and ROC curve, the experimental results indicate that the proposed scheme was trained effectively on the CyberSecurity-Attacks dataset and delivered competitive and realistic results. Recursive feature elimination is a well-tested method employed to identify and select significant features from the Cyber Security-Attacks dataset. It employs lableencoder and minmaxscaler to encode and scale the data.

The hybrid deep learning model (CNN-RNN-BiLSTM) outperforms the basic RNN model in all major evaluation metrics — accuracy, precision, recall, and F1-score. The hybrid model's ability to learn spatial features (CNN), sequential patterns (RNN), and long-term dependencies (BiLSTM) helped it to better classify cybersecurity attacks from unstructured data.

This result suggests that for real-time cybersecurity attack detection systems, using a hybrid deep learning approach is highly recommended over simple models like RNN. It enhances the system's reliability in detecting complex attack patterns while minimizing false alarms.

REFERENCES:

1. TeamInCribo. (n.d.). Cyber security attacks. Kaggle. https://www.kaggle.com/datasets/teamincribo/cyber-security-attacks?select=README.md
2. Bamber, S. S., Katkuri, A. V. R., Sharma, S., & Angurala, M. (2025). A hybrid CNN-LSTM approach for intelligent cyber intrusion detection system. Computers & Security. https://www.sciencedirect.com/science/article/abs/pii/S0167404824004516
3. https://www.kaggle.com/datasets/teamincribo/cyber-security-attacks?select=README.md
4. Ahmad, R., Alsmadi, I., Alhamdani, W., & Tawalbeh, L. (2022). A deep learning ensemble approach to detecting unknown network attacks. Journal of Information Security and Applications. https://doi.org/10.1016/j.jisa.2022.103196
5. Al-Omar, B., & Trabelsi, Z. (2023). Intrusion detection using attention-based CNN–LSTM model. In Artificial Intelligence Applications and Innovations. Springer Nature.
6. Anderson, J. P. (1980). Computer security threat monitoring and surveillance. Technical Report. https://cir.nii.ac.jp/crid/1573950399661362176
7. Ashiku, L., & Dagli, C. (2021). Network intrusion detection system using deep learning. In Procedia Computer Science (pp. 239–247). Elsevier.
8. Bu, S. J., & Cho, S. B. (2020). A convolutional neural-based learning classifier system for detecting database intrusion via insider attack. Information Sciences, 123–136.
9. Chkirbene, Z., Erbad, A., Hamila, R., Mohamed, A., Guizani, M., & Hamdi, M. (2020). TIDCS: A dynamic intrusion detection and classification system based feature selection. IEEE Access, 95864–95877.
10. Debar, H., Dacier, M., & Wespi, A. (1999). Towards a taxonomy of intrusion-detection systems. Computer Networks, 805–822.
11. Dong, B., & Wang, X. (2016). Comparison deep learning method to traditional methods using for network intrusion detection. In 8th IEEE International Conference on Communication Software and Networks (ICCSN 2016) (pp. 581–585).
12. Elmasry, W., Akbulut, A., & Zaim, A. H. (2020). Evolving deep learning architectures for network intrusion detection using a double PSO metaheuristic. Computer Networks.

13. Ferrag, M. A., Maglaras, L., Moschoyiannis, S., & Janicke, H. (2020). Deep learning for cyber security intrusion detection: Approaches, datasets, and comparative study. Journal of Information Security and Applications. https://doi.org/10.1016/j.jisa.2019.10241
14. Fu, X., Zhou, N., Jiao, L., Li, H., & Zhang, J. (2021). The robust deep learning–based schemes for intrusion detection in internet of things environments. Annals of Telecommunications, 273–285. https://doi.org/10.1007/s12243-021-00854-y
15. Govindarajan, M., & Chandrasekaran, R. (2011). Intrusion detection using neural based hybrid classification methods. Computer Networks, 1662–1671.
16. Guler, H., & Alpay, O. (2021). Intrusion detection and classification based on deep learning. In 14th International Conference on Information Security and Cryptology (ISCTURKEY 2021) (pp. 40–44).
17. Hassan, M. M., Gumaei, A., Alsanad, A., Alrubaian, M., & Fortino, G. (2020). A hybrid deep learning model for efficient intrusion detection in big data environment. Information Sciences, 386–396.
18. Canadian Institute for Cybersecurity. (2017). IDS 2017 Dataset. https://www.unb.ca/cic/datasets/ids-2017.html
19. Canadian Institute for Cybersecurity. (2018). IDS 2018 Dataset. https://www.unb.ca/cic/datasets/ids-2018.html
20. Kasongo, S. M. (2023). A deep learning technique for intrusion detection system using a recurrent neural networks based framework. Computer Communications, 113–125.
21. Kavitha, R., & Amutha, S. (2022). Performance analysis of deep neural network and LSTM models for secure network intrusion detection system. In Proceedings of 4th International Conference on Cybernetics, Cognition and Machine Learning Applications (ICCCMLA 2022) (pp. 390–396).
22. Tavallaee, M., Bagheri, E., Lu, W., & Ghorbani, A. A. (2009). A detailed analysis of the KDD cup 99 data set. In IEEE Symposium on Computational Intelligence for Security and Defense Applications (CISDA 2009).
23. KDD Cup. (1999). KDD Cup 1999 Dataset. https://scholar.google.com/scholar?q=Kdd%20Cup%201999
24. Khamis, R. A., Shafiq, M. O., & Matrawy, A. (2020). Investigating resistance of deep learning-based IDS against adversaries using min-max optimization. In IEEE International Conference on Communications (ICC 2020).
25. Khan, A. Z., Shiang, A. S., Abdullah, C. W., & Ahmad, F. (2021). Network intrusion detection system: A systematic study of machine learning and deep learning approaches. Transactions on Emerging Telecommunications Technologies. https://doi.org/10.1002/ett.4150
26. Kunang, Y. N., Nurmaini, S., Stiawan, D., & Suprapto, B. Y. (2021). Attack classification of an intrusion detection system using deep learning and hyperparameter optimization. Journal of Information Security and Applications.
27. Liu, H., & Patras, P. (2022). Netsentry: A deep learning approach to detecting incipient large-scale network attacks. Computer Communications, 119–132.
28. Masum, M., & Shahriar, H. (2020). TL-NID: Deep neural network with transfer learning for network intrusion detection. In 15th International Conference for Internet Technology and Secured Transactions (ICITST 2020).
29. Mishra, P., Varadharajan, V., Tupakula, U., & Pilli, E. S. (2019). A detailed investigation and analysis of using machine learning techniques for intrusion detection. IEEE Communications Surveys & Tutorials, 21, 686–728.
30. Mohan, V. M., Singh, S., & Jadhav, P. P. (2022). Optimized deep ensemble technique for malicious behavior classification in cloud. Journal of Information Privacy and Security, 54, 859–887. https://doi.org/10.1080/01969722.2022.2122015
31. Najafabadi, M. M., Villanustre, F., Khoshgoftaar, T. M., Seliya, N., Wald, R., & Muharemagic, E. (2015). Deep learning applications and challenges in big data analytics. Journal of Big Data, 2(1), 1–21. https://doi.org/10.1186/s40537-014-0007-7