International Journal of Environmental Sciences
ISSN: 2229-7359
Vol. 11 No. 17S, 2025
https://www.theaspd.com/ijes.php

An Investigation Of The ML/DL Hybrid Model For Social Media Attack Prediction And Detection

Dr. Gaurav Aggarwal[1] and Rashmi Tiwari[2]
[1]Professor, Faculty of Engineering & Technology, Jagannath University, Delhi NCR, Bahadurgarh, India
[2]Research Scholar, Faculty of Engineering & Technology, Jagannath University, Delhi NCR, Bahadurgarh , India
[1]gaurav.aggarwal@jagannathuniversityncr.ac.in and [2]rt45720@gmail.com

Abstract: *Deep learning-based intrusion detection technology has been extensively researched in both academia & business. In order to classify challenges involving hybrid model to predict and detect attacks in social media, deep learning uses hybrid models to extract probable features from high-dimensional data. The 1D-CNN is a novel form of the CNN that can distinguish between normal and attack input in order to identify attacks Separate DL models based on 1D-CNN were created and applied to both the combined datasets and the other sub-datasets. The suggested IDS uses a deep learning model that blends CNN & LSTM algorithms, providing a hybrid approach. CNN can identify unknown threats since it is built to recognize objects.*

Keywords: *CNN, Deep learning, Machine Learning, IDS, social media, Attacks.*

INTRODUCTION

Deep learning uses ANN algorithms for machine learning, making it hierarchical. Assaults are identified using 1D-CNN, a unique form of convolutional neural networks, which classifies traffic into normal and attack data (CNN). When real-world data increases over time, generating a high-dimensional space, the performance of ML algorithms declines because these techniques rely too much on the qualities chosen by human experts. By automatically learning features from a massive amount of data, DL was able to circumvent this limitation, thanks to its complex architecture. In this study, we propose a 1D-CNN & LSTM hybrid neural network for social media assault prediction.

OBJECTIVES

1. To study the attacks on social media, implement machine learning models.
2. To design the abnormalities in data quartile-based approach based on statistical analysis
3. To evaluate data design exploratory data analysis and visualization system to find the correlation among various features present in data.
4. To accuracy and performance of machine learning and deep learning models implement performance matrices.

METHODOLOGY

The study of research methodology is a branch of the scientific community. It is a method for methodically resolving the research issue. An organized plan outlining the procedures to be followed in conducting the research has been prepared as part of the research design for the study. This research is a descriptive empirical survey.

Data Collection

The researcher has used different sets of data collection techniques to have comprehensive and desired information. The nature of the study demanded that the researcher should collect data from various reliable sources. The researcher has used and relied on both the primary and secondary sources of data collection.

Primary Data

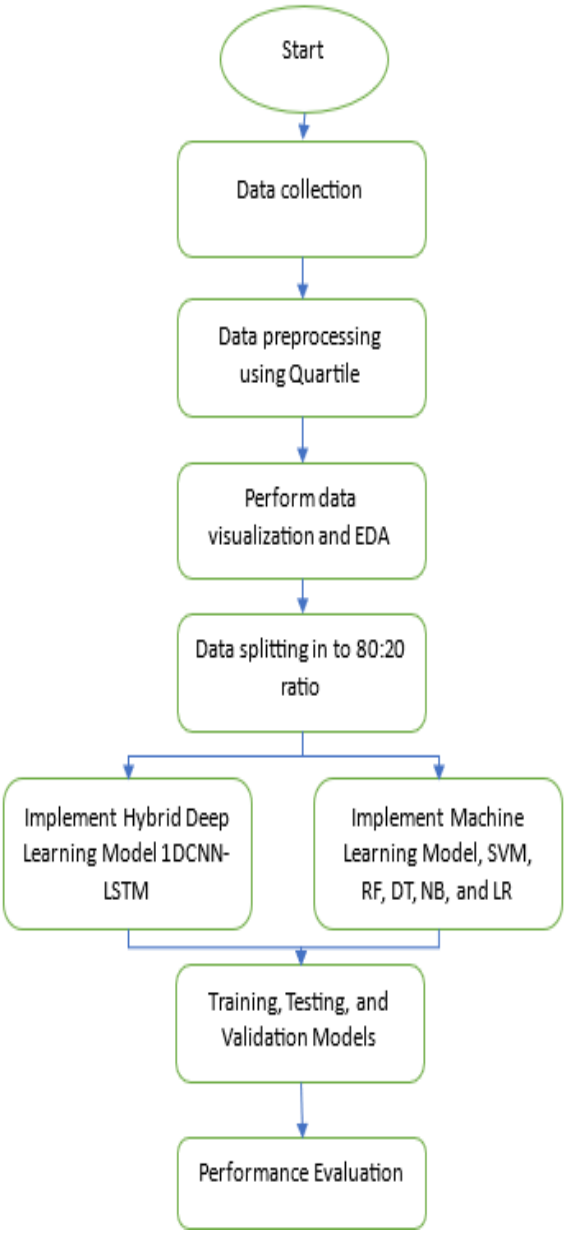
Primary data consists of information gathered directly by the researcher. You can trust it more since it is

genuine. Questions, interviews, observations, and the provision of schedules are all viable methods for its acquisition. The researcher in this study gathered primary data through the use of questionnaires.

Secondary Data

The primary researcher may make use of secondary data, which consists of information gathered from other sources such as agencies and studies. Academic journals, vernacular news studies, annual reports, government reports, and trustworthy websites are the sources from which it is compiled. Secondary data gathered from a variety of sources is utilized in this research.

Flow Chart



RESULTS AND DISCUSSION

There is an imbalance in the UNSW-NB15 dataset, with one class being lower than the other. Here we can see the x-axis showing the class (0 normal & 1 attack) & y-axis showing the count of the class. A classification issue called an imbalanced dataset led to the model's bias toward the majority, as shown in Fig. 1a, where the attack class is higher and the normal class is lower. There are two ways to deal with datasets that aren't balanced: using a cost function or sampling. The method based on sampling was

selected for this article. For the purpose of minimizing the difference between two classes, two sampling-based approaches can be employed: under-sampling and over-sampling. The under-sampling method is used to decrease the dataset size by removing instances from the majority class that are selected at random. Less accuracy could occur as a result of the accidental deletion of crucial data. Examples in the minority class are randomly duplicated using the over-sampling technique. This method avoids losing data, but it could be overfitted if there is a lot of duplicate data. SMOTE is employed in this paper to circumvent the issue of overfitting.

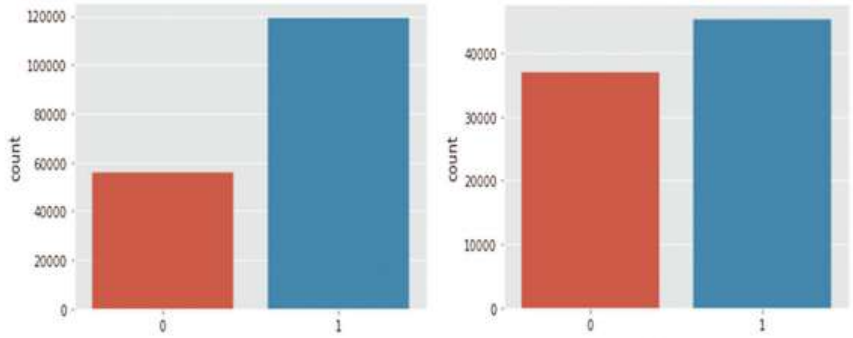


Figure 1: Data distribution (a) before SMOTE (b) after SMOTE in UNSW-NB

Algorithm 1: SMOTE Technique

Input: original training set **(X)**, Percentage of oversampling **(N)**, and KNN numbers **(K)**.

Output: oversampled training set.

$\mathbf{n}$ = #observation, $\mathbf{m}$ = #attributes, $\mathbf{n_{min}}$= minobservation

if $\mathbf{N} < 100$, **then**

Stop: warning (N must be greater than 100)

end if

$N = \text{int}(\frac{N}{100})$

$S_{(n*N)\times m}$ is an empty array for synesthetic samples

for i $\leftarrow$ 1 to n_{min} **do**

Compute *KNN* for *i* and save the indices in the nnarray

newindex $\leftarrow$ 1

while N $\neq$ 0 **do**

K_c= random number in range 1 and K

for j $\leftarrow$ 1 to *m* **do**

diff $\leftarrow$ = X[nnarray[K_c]][j]$-$X[i][j]

gap $\leftarrow$ uniform (0, 1)

synthetic[newindex][j] $\leftarrow$ X[i][j]+gap $\times$ diff

end for

newindex = newindex + 1

N = N $-$ 1

end while

end for

return X

CNN, LSTM, and dense layers comprise architecture.

The suggested IDS uses a deep learning model that blends CNN & LSTM algorithms, providing a hybrid approach. Their combined feature extraction, memory retention, and classification capabilities outperform those of the individual models in this aggregation. Figure 2 shows the three layers that make up the proposed model architecture: dense layers, CNN, & LSTM.

CNN can identify unknown threats since it is built to recognize objects. In order to reduce parameter & weight sharing, CNN automatically extracts features and executes a challenging operation on buried layers. CNNs include a number of layers, including convolutional, pooling, and fully connected ones. Four convolutional (Conv) layers make up the architecture of our CNN layer. After every pair of convolutional layers, there is a single pooling layer. There are 44 input features that are designed into a 1D matrix. Convolution 1 and Convolution 2 were run with a 32 × 64 feature matrix, a 3 × filter size, and 32 × convolution filters. The output shape of Conv 3 and Conv 4 was a 16 × 128 feature matrix, with 64 convolution filters and a filter size of 3. An activation map with a padding value of 1 is generated by each filter. The convolution layers employ the Rectified Linear Unit (Relu) activation function.

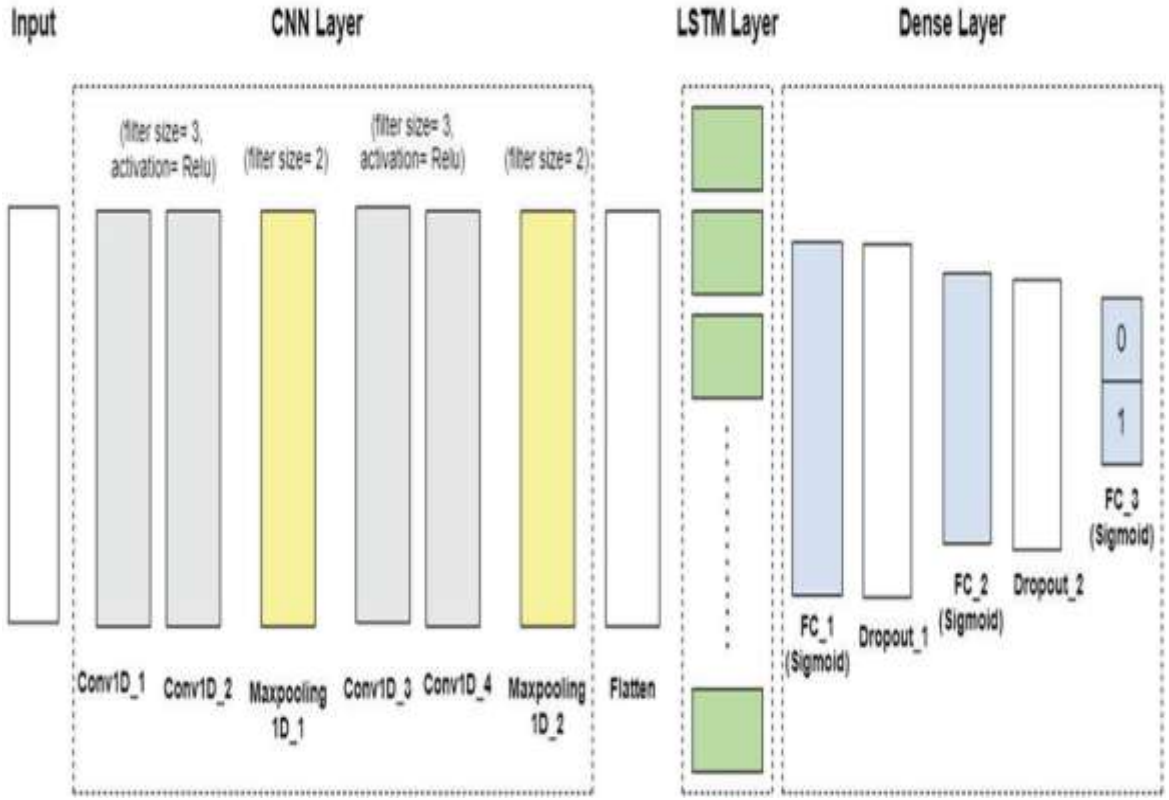


Figure 2: Proposed model architecture

Long Short-Term Memory (LSTM)

One variant of RNN that seeks to manage time-series data is LSTM, which was developed to address RNN's short-term memory issue. Convolutional layers are used to extract features in the CNN+LSTM hybrid model, while LSTM units are used to provide sequence prediction across time. For the purpose of training the weights & creating a time-varying signal model, CNN feeds its output into LSTM units. The LSTM units are made up of three gates: input, forget, and output. These gates regulate the entry, storage, and exit of data sequences within the network. Finding out if the data from the prior timestamp should be preserved or deleted is the initial stage.

Dense Layer

When the LSTM units learn higher-order feature representations appropriate for classifying the output as either normal or attack, the dense layer's output is directed. Finally, the design incorporates two

dropout layers & three fully connected (FC) layers. Following the LSTM units, the FC layer is enhanced with the sigmoid function for low-level feature-based categorization.

CNN-Based IDS Approaches

An extensive review of the literature on CNN-based IDS methods is presented in this section. There are a total of four groups that the approaches that were assessed fall into. The first group consists of approaches that just use a CNN. Group 2 also includes intrusion detection systems that use a CNN & RNN in tandem. Use a ANN technique like a GRU or LSTM to do this. This category makes use of RNN algorithms for the purpose of extracting temporal information. On the other hand, IDS solutions that use a combination of CNNs and deep learning techniques other than RNNs are included in group 3. Group 4 concludes with intrusion detection systems that use a combination of CNN and other algorithms such as fuzzy logic, clustering, Fourier transformation, or evolutionary algorithms.

- **Single CNN-Based Schemes**

Here we give and discuss studies that study CNNs in their purest form, without any additional machine learning or deep learning algorithms. Researchers have looked into improving the CNN algorithm as well as feature preprocessing, data reduction, feature fusion, & imbalanced data management. By integrating dimensionality reduction & feature engineering, Al-Turaiki et al. proposed a two-stage preprocessing approach. The development of useful features also made use of deep-feature synthesis. Alternatively, statistical behaviors were used by Lam et al. instead of the usual anomalous attack behaviors due to their ease of computation & extraction without compromising efficiency. If they are able to train the CNN's kernel to accurately reflect the network's characteristics, Jo et al. claim that the CNN's input data might outperform competing deep learning models in network intrusion detection systems. But this can't happen without good preprocessing, thus the researchers suggested utilizing the "direct" approach. This allows the network intrusion detection system to take advantage of the kernel's capabilities by making use of the bare minimum of protocol data, field size, and offset. "Weighted" and "compressed" are two further preprocessing methods that they suggest. To use these strategies, you also need network information. The field-to-pixel philosophy, which informed the approaches proposed by Jo et al., allows for the extraction of each pixel's convolutional features, which is useful for CNNs. Because it faithfully reproduces the CNN's convolutional features, the direct method makes field-to-pixel conversion the most natural and user-friendly approach. The study by Kim et al. focused on one particular type of denial of service. They also looked at ways to identify attacks that fall into the same general group. They accomplished this by taking into account the image type, kernel size, and number of convolutional layers as hyperparameters for eighteen different scenarios. Following the establishment of binary & multiclass classifications for every scenario, the best and highest-performing scenarios were found. The model with three convolutional layers and 2 × 2 or 3 × 3 kernel sizes was found to have the best performance. The model utilizing two convolutional layers yielded the best results, nevertheless, when the kernel size was raised to 4 x 4. When using multiclass classification, performance was often good when using many convolutional layers. However, it seems that neither the binary nor the multiclass classifications were greatly impacted by the kernel size. The investigation presented a CNN architecture with a means-convolutional layer (CNN-MCL). The aberrant content attributes are learned by this layer so that it can recognize those particular anomalies in the future. By incorporating a novel convolutional layer that permits the learning of low-level anomalous features, the CNN-MCL model paves the way for a powerful network intrusion detection system.

- **Hybrid CNN/RNN Schemes**

To extract temporal properties, RNNs depend on sequential data. However, the disappearing gradient problem isn't exclusive to RNNs; GRU & LSTM were built to solve it. In order to handle both spatial and temporal information concurrently, several IDS systems have used a combination of CNNs & RNNs. The CNN is able to extract the global and spatial components, whereas the RNN, GRU, & LSTM can keep the temporal aspects. On the other hand, more training data samples may be necessary if this combination increases the model's complexity. Wu et al. [38] created a CNN+RNN system with a

hierarchical structure. A very low percentage of false positives is associated with this system's excellent detection capabilities. However, when the samples in the training dataset are too few, their suggested model fails to categorize attacks. This is a fundamental problem. Inter alia, Yao et al. [39] created an AMI intrusion detection model that fused CNN & LSTM features across layers. The CNN part finds and recognizes local features so it may learn global ones. In contrast, the LSTM part uses a memory function to extract periodic features. This is how you get a full range of features that are good for more than one domain. This allows AMI to detect intrusion information with a high degree of accuracy.

- **An Approach to Deep Learning and Hybrid Convolutional Neural Networks**

This particular set of papers has two different kinds of models. A number of models have been influenced by popular deep learning frameworks, such as TextCNN and Inception. Second, models denoise, clean, and extract features by using deep learning techniques such AE & denoising autoencoder (DAE). Statistical and payload aspects are utilized by Min et al. in the TR-IDS approach. This procedure extracts useful information from payloads using word embedding and Text-CNN. Furthermore, it categorizes Biflow, which differs from packet-level datasets in that it includes additional temporal data. This procedure makes use of random forest classification in addition to statistical & payload feature extraction. The study's two-stage network intrusion detection method used CNN and GoogLeNet Inception as its foundational models. To identify problems with network packet binaries, this system uses the GoogLeNet Inception Model. The next step is to extract features from the packet's raw data. In order to train the system and improve the convergence speed of the model, the given CNN algorithm also uses an inception model & batch normalization approach. By expanding the breadth and depth of the network, this strengthens the model. At the end of the day, it makes the network more scalable. The hierarchical structure of network traffic (byte-packet-flow) should also be remembered. The team built a CNN using hierarchical packet data. The PBCNN is the name given to this. Bytes in an unprocessed Pcap file packet are automatically used by this model to identify abstract features at the first level. Because of this, instead of utilizing feature-ready CSV files, the packets in a session or flow are displayed at the second level. The efficient utilization of all original data information is guaranteed by this. The representation of traffic flow can be obtained using a one-layer Text CNN, or several convolution-pooling modules are included in byte-friendly filter sizes. In order to categorize attacks, this data is further fed into three interconnected networks. To make network attack categories more reliable, PBCNN-based few-shot learning can be used.

- **Method Schemes for Machine Learning, Including Hybrid CNNs**

The goal of Moustakidi et al. was to improve consumers' confidence in and comprehension of data without sacrificing accuracy. They came up with a solution that combines the information gathered by the IDS into a single risk indication that can be used for action. The feature extraction pipeline relies on three main components: a fuzzy allocation scheme for transforming raw data into fuzzy class memberships, a new transformation mechanism called Vec2im for converting feature vectors into images, & dimensionality reduction module that employs Siamese CNN to reduce the input data's dimensionality to a 1D feature space. In the second stage of processing, the memberships that were generated are transformed into a matrix format. This results in the production of one grayscale image for each case. To be more specific, 41 fuzzy memberships, ui (xk,j), j = 1,... 41 were created from 41 characteristics, xk,j j = 1. The process culminates in creating a 7 × 7 image for every sample by inserting fuzzy memberships into a matrix. To fill the eight gaps, random cells of the matrix were additionally filled with zero values. Random ordering was applied to fuzzy memberships and zero values once it was determined that the order did not significantly impact the final performance of the newly constructed system. Concurrently, Nguyen et al. suggested a feature subset selection method that combines exhaustive search based on evolutionary algorithms with fuzzy C-means clustering. An effective extractor, the CNN model is part of the algorithm's design for identifying bagging (BG) classifiers. The BG classifier is used to validate the performance using the deep feature subset extracted by the chosen CNN model. Afterwards, the final detection system's performance was significantly improved by integrating a high-quality feature set obtained from the GA's three-layered feature building with the GA,

FCM, CNN extractors, hybrid CNN, or BG learning methods. It is necessary to choose features, select a model, and validate the model in order to create this new model.

PROPOSED WORK

The steps that we will work on as part of this proposed work are as follows: first, we will collect data from the UNSW website; then, we will preprocess the data by removing any null and superfluous information; then, we will perform EDA by separating the data into train and test sets; and finally, we will train ML and DL models.

1. Data collection

We needed a data set that was based on attacks in order to implement machine learning and deep learning models for the purpose of identifying attacks in social media and online shopping. For this purpose, we will use data such as UNSW nb-15 data for attack detection.

2. Data Preprocessing and Perform EDA

Design Quartile 25–75% based on the identification of data abnormalities; remove upper and lower bound of outer data by using outlier filtration; apply pandas' data frame functions over data frame to remove null and nan values; Use the matplotlib and seaborn libraries to perform EDA tasks such as plotting graphs and performing visualisation. And last but not least, the data will be divided into training and testing ratios, such as 80:20.

3. Machine Learning and Deep Learning Modeling

Create a hybrid neural network that is based on LSTM and 1D-CNN. In addition to this, implement state-of-the-art machine learning algorithms such as the random forest, decision tree, support vector machine, naive Bayes, and logistic regression.

4. Performance Evaluation

Following training, the performance of the models will be evaluated using matrices such as accuracy, precision, recall, f-score, and loss of models.

We overcame this obstacle by comparing the outcomes of an empirical experiment we ran on well-known CNN-based methods using industry-standard datasets. The suggested model's LSTM, Hybrid LSTM-GRU, & 1D CNN models' performance evaluations are presented in Table 1. One popular statistic for evaluating performance is accuracy, which is defined as the percentage of correct classifications relative to the total. One should not rely on accuracy as a measuring tool in the presence of an imbalanced dataset.

Table 1: Performance evaluation of LSTM model and Hybrid LSTM-GRU and 1D CNN model

Model	Accuracy	Val Acc	Loss	Val Loss
LSTM Model	0.99	0.98	0.01	0.55
Hybrid LSTM- GRU and 1D CNN model	0.99	0.97	0.01	0.07

CONCLUSION

IDSs use various AI approaches, like ML, to improve performance against new cyber-attack challenges; they are vital in defending computer systems and networks around the world. When it comes to privacy and security concerns, researchers have extensively used CNN, one of the deep learning algorithms, to enhance IDS solutions. Accordingly, a thorough overview of CNN-based IDS methods is given in this paper. Due to its intricate architecture, DL was able to get around this restriction by automatic learning features from a vast volume of data. To anticipate attacks on social media, we suggest a hybrid neural network based on 1D-CNN & LSTM in this research. We use the Accuracy, Prediction, Recall, and F1-Score to evaluate the experimental model's performance. These evaluation criteria reflect the performance of the intrusion detection system's flow recognition accuracy rate, and false alarm rate. The Performance evaluation of LSTM model accuracy is 0.99, Val Acc 0.98, Loss 0.01, Val Loss 0.55 and Hybrid LSTM-GRU and 1D CNN model accuracy is 0.99, Val Acc 0.97, Loss 0.01, Val Loss 0.07 of the proposed model are presented.

REFERENCES

- Akhtar, M. S., & Feng, T. (2022). Detection of Malware by Deep Learning as CNN-LSTM Machine Learning Techniques in Real Time. Symmetry, 14(11), 2308.
- Liu, Y., Wang, X. K., Hou, W. H., Liu, H., & Wang, J. Q. (2022). A novel hybrid model combining a fuzzy inference system and a deep learning method for short-term traffic flow prediction. Knowledge-Based Systems, 255, 109760.
- Ma, C., Du, X., & Cao, L. (2019). Analysis of multi-types of flow features based on hybrid neural network for improving network anomaly detection. IEEE Access, 7, 148363-148380.
- Mahajan, S., HariKrishnan, R., & Kotecha, K. (2022). Prediction of network traffic in wireless mesh networks using hybrid deep learning model. IEEE Access, 10, 7003-7015.
- Mrazova, I.; Kukacka, M. Can deep neural networks discover meaningful pattern features? Procedia Comput. Sci. 2012, 12, 194–199. [Google Scholar] [CrossRef]
- Petmezas, G., Haris, K., Stefanopoulos, L., Kilintzis, V., Tzavelis, A., Rogers, J. A., ... & Maglaveras, N. (2021). Automated atrial fibrillation detection using a hybrid CNN-LSTM network on imbalanced ECG datasets. Biomedical Signal Processing and Control, 63, 102194.
- Rizvi, S., Scanlon, M., McGibney, J., & Sheppard, J. (2022). Deep learning based network intrusion detection system for resource-constrained environments. In Springer (pp. 1-7).
- Sasidhar, T. T., Premjith, B., & Soman, K. P. (2020). Emotion detection in hinglish (hindi+ english) code-mixed social media text. Procedia Computer Science, 171, 1346-1352.
- Satyanegara, H. H., & Ramli, K. (2022). Implementation of CNN-MLP and CNN-LSTM for MitM Attack Detection System. Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi), 6(3), 387-396.
- Song, Z. English speech recognition based on deep learning with multiple features. Computing 2020, 102, 663–682. [Google Scholar] [CrossRef]
- Tian, C.; Fei, L.; Zheng, W.; Xu, Y.; Zuo, W.; Lin, C.-W. Deep learning on image denoising: An overview. Neural Netw. 2020, 131, 251–275. [Google Scholar] [CrossRef]