CS577: Deep Learning
Project

Detection of Epileptic Seizure from Deep Learning Techniques

Sushravya Udapikar (CWID: A20447295)

Yogasree Segar (CWID: A20448385)

INDEX

I. Abstract

Epileptic seizure is a condition when sudden triggers are encountered in the nervous system that causes hallucinations, body impairment and unconsciousness. Since this disturbance is a due to a sudden trigger, it is highly difficult to determine the issue prior to give proper treatment. But the activity of a brain is monitored through the electroencephalogram (EEG). It is possible to determine the occurrence of a seizure by continuously monitoring the EEG report of a person vulnerable to the seizure. The aim of this project is to detect epileptic seizure through deep learning techniques that are applied on the EEG report. The methods of Multi-Layer Perceptron, Convolution Neural Networks and finally Bi-LSTM are applied to the reports to create a model that can determine epileptic seizure.

II. Problem Statement

Epileptic Seizure

Human nervous system that consists of millions of nerves communicates through electrical signals. Sometimes a sudden electrical disturbance is caused in the electrical activity of the brain. This causes the epileptic seizure. As there is miscommunication among the nerves, the coordination between the body parts is lost. Since epileptic seizures are sudden triggers, it can cause physical as well social problems. About 50 million people worldwide are determined to have epileptic seizures and the most affected people are from the older and younger groups.

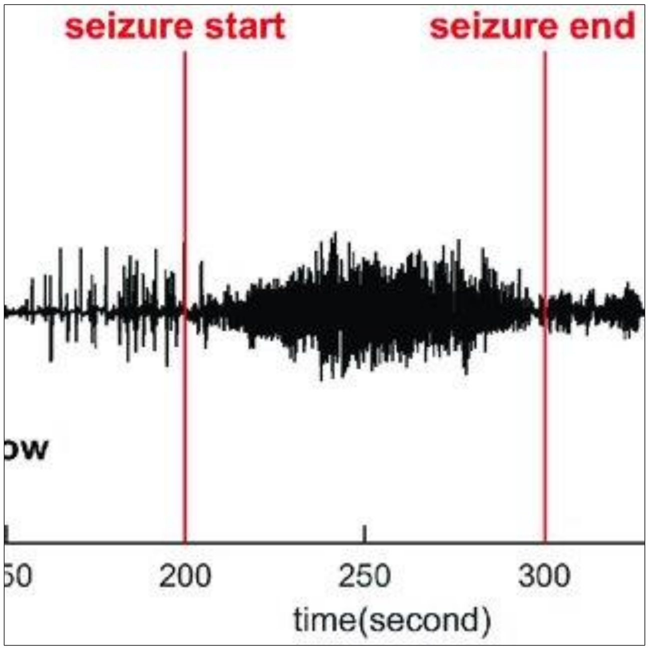
Diagnosing Epileptic Seizure

Epileptic seizure is usually diagnosed by doctors from the Electroencephalogram (EEG) reports. These reports are the timestamp reports of the activity of the brain. An EEG report at a normal time is considered to have a waveform at a normal pace. During a seizure, the EEG reports consists of waveforms with various distortions and with high frequency. Thus it forms a pathway to diagnose seizure through the EEG reports.

Phases of Seizure

The EEG waveform can be classified into three types based on the seizure. They are:

- Interictal – The timestamp of the EEG with normal waveform and no distortions.
- Preictal – The timestamp of the waveform before the seizure. This includes the portion of the timestamp when various symptoms of seizure can be noticed in the patients.
- Ictal – The timestamp of EEG when the seizure occurs. This portion is supposed to have high distortions in the waveform.
- Postictal – The timestamp after the seizure with normal waveform indicating the normal working nature of the brain.

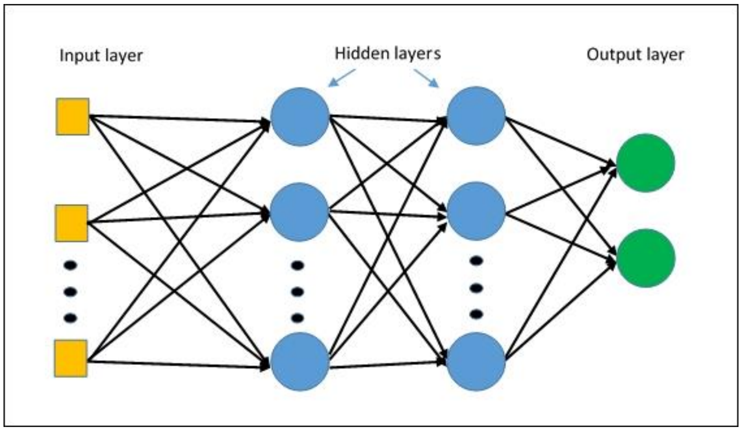


III. Deep Learning Methods

Various deep learning methods are applied on the epileptic seizure datasets and models are created and trained on the phases of the seizure. These models are then hyper tuned to detect seizure in prior so as to make the seizures less effective.

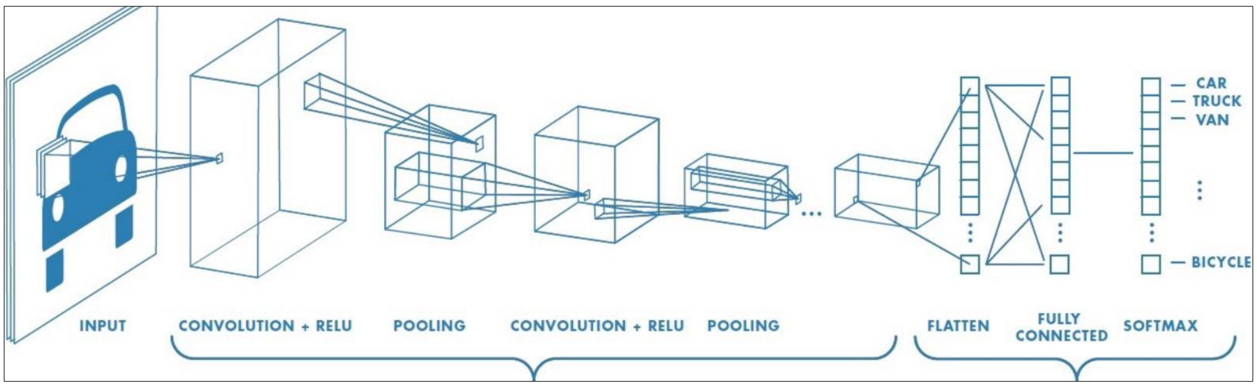
Multilayer Perceptron (MLP)

A multilayer perceptron or a fully connected layer describes a set of layers where each layer has its own hidden units. These units are a package of mathematical functions that aims in extracting the features in a given input so as to conclude to an output. The input is usually a tensor that is passed to all the hidden units of the first layer. It processes the input and produces the output that is fed to the next layer as an input. A sequence of such layers finally produce the necessary output. Each model of MLP has some parameters depending on the number of hidden units and a bias. A MLP usually achieves high accuracy with many hidden layers thus leading to many parameters.



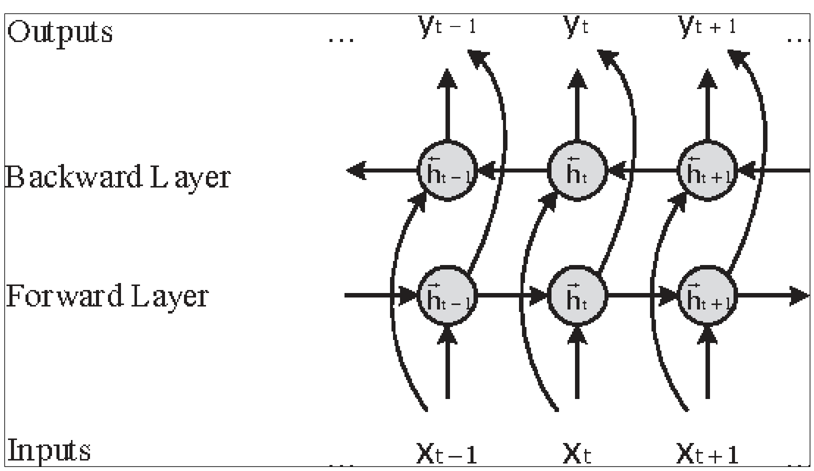
Convolution Neural Network (CNN)

Convolution neural networks includes the process of convolution among the images that are passed as the input and filters. It includes filters of fixed dimensions that are created and convoluted to the images. The aim of this convolution is to extract features from various images to produce results. Another important function performed in Convolution Neural Networks is pooling. The aim of this function is to reduce the dimensionality of the images at the same time maintain its features. Thus by doing so the number of parameters trained in the network is reduced (usually halved). An advantage of CNN over MLP is the number of parameters is reduced by both convolution and pooling. Also a high accuracy can be attained with less number of middle layers.



Bi-LSTM

Bidirectional LSTM works by duplicating the layer of recurrent layers. To the first layer, the input is passed and to the second layer the reversed input is passed. By this way the model learns more than any other methods. It is more beneficial in speech recognition but it can also be beneficial in this case. The layers of Bi-LSTM consists of many LSTM cells that perform specific mathematical functions to extract specific features. Bi-LSTM when connected with CNN proves to be more efficient.



IV. Dataset

1. UCI Dataset: Epileptic Seizure Recognition Data Set

This dataset consists of 11500 samples with 5 classes. One of the classes give recording of the seizure activity. Each file among the files in database is a recording of the brain activity for 23.6 seconds. The corresponding time is sampled into 4097 data points. Total there are 500 individuals data recorded with 4097 data points for 23.6 seconds time period. The data points of 4097 is randomly shuffled and divided into 23 chunks which contains 178 data points for 1 second. So considering 500 individual and 23 chunks so total samples available = (23x500 = 11500). This dataset is suitable for the MLP and BiLSTM models.

The link to the dataset is given below:
https://archive.ics.uci.edu/ml/datasets/Epileptic+Seizure+Recognition

2. PhysioNet ATM

This dataset is considered for CNN model which requires images. The images of this dataset is exported to the UCI dataset and is preprocessed for a faster work. This dataset consists of various timestamp reports that are obtained in real life and is uploaded for public use. The dataset CHB – MIT EEG samples is considered. The dataset consists of 24 classes where each class has 35 samples of EEG timestamp. The timestamp images are downloaded from this site. For the purpose of accuracy, 10 second time sample is selected. The images are then preprocessed and then classified into ictal and preictal stages to make them suitable for CNN.

The link to the dataset is given below:

https://archive.physionet.org/cgi-bin/atm/ATM

V. Proposed Solution

The deep learning methods to detect the epileptic seizure from the EEG samples was obtained from the following reference paper “Efficient Epileptic Seizure Prediction based on Deep Learning” published in *IEEE Transactions on Biomedical Circuits and Systems, October 2019.*

Approach of Reference Paper

Five different models were implemented and evaluated against the dataset with 8 instances of EEG and produced a maximum accuracy of 83% in MLP and 96 % in the other methods. They are

1. Multi-layer Perceptron (MLP)
2. Deep Convolution Neural Networks and Multi-layer Perceptron
3. Deep Convolution Neural and BiLSTM
4. DCAE + BiLSTM
5. DCAE + BiLSTM + CS

The aim of this project is to implement these proposed models of MLP, CNN and Bi-LSTM for a larger dataset and evaluate them as well as produce a better model for the larger datasets.

VI. Methodology

The proposed models were implemented using keras through which the models were created and trained. Finally the accuracy of the models were tested. In cases of poor accuracy of the proposed models, a better model was created.

Preprocessing

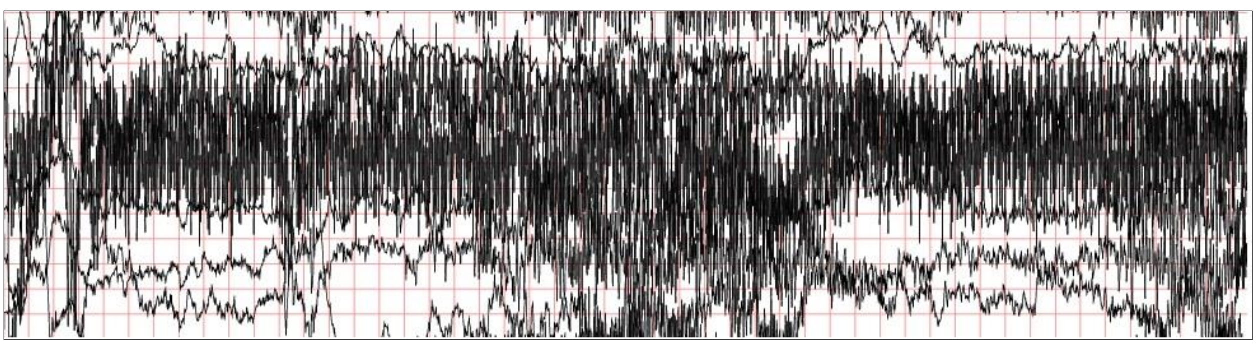
General preprocessing is done before creating the model. This includes preprocessing the dataset to make it suitable for the model.

UCI dataset – The UCI dataset is preprocessed to normalize the input and to encode the labels. The unnecessary columns are removed and the labels are popped out. Finally the data is split into train, test and validation set randomly.

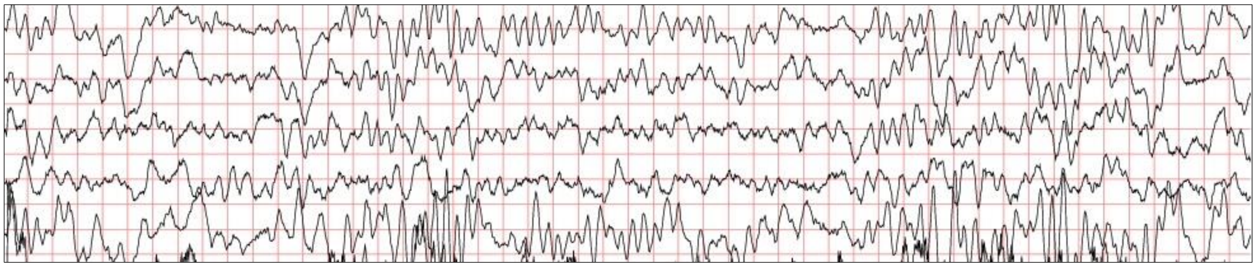
	Unnamed: 0	X1	X2	X3	X4	X5	X6	X7	X8	X9	...	X170	X171	X172	X173	X174	X175	X176	X177	X178	y
0	X21.V1.791	135	190	229	223	192	125	55	-9	-33	...	-17	-15	-31	-77	-103	-127	-116	-83	-51	4
1	X15.V1.924	386	382	356	331	320	315	307	272	244	...	164	150	146	152	157	156	154	143	129	1
2	X8.V1.1	-32	-39	-47	-37	-32	-36	-57	-73	-85	...	57	64	48	19	-12	-30	-35	-35	-36	5
3	X16.V1.60	-105	-101	-96	-92	-89	-95	-102	-100	-87	...	-82	-81	-80	-77	-85	-77	-72	-69	-65	5
4	X20.V1.54	-9	-65	-98	-102	-78	-48	-16	0	-21	...	4	2	-12	-32	-41	-65	-83	-89	-73	5

PhysioNet dataset – PhysioNet datasets are available in the form of EEG waveforms that are to be sorted and classified into preictal and ictal datasets. The waveforms are selected from a 10 second timestamp to develop a better model. These images were then split into test, train and validation datasets and were trained upon. Due to time constrain only 60 samples were considered. The data images obtained were also augmented to develop a better model.

Ictal Phase



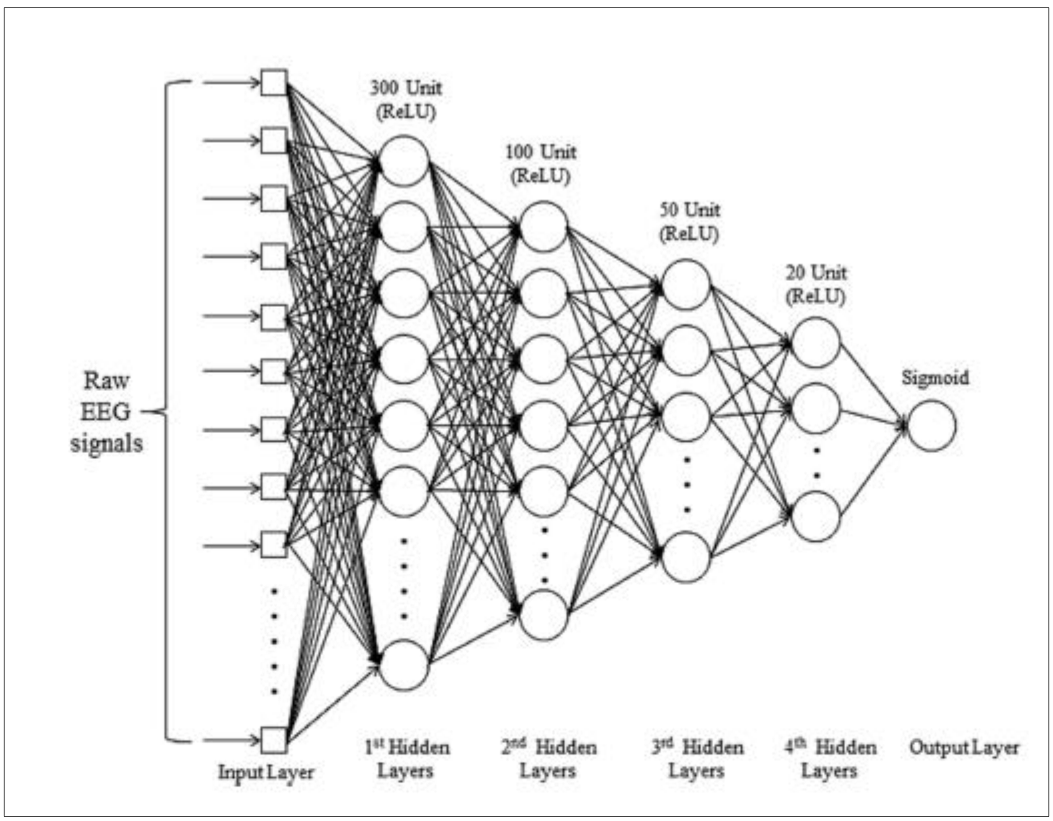
Pre ictal Phase



Proposed Models in the Paper

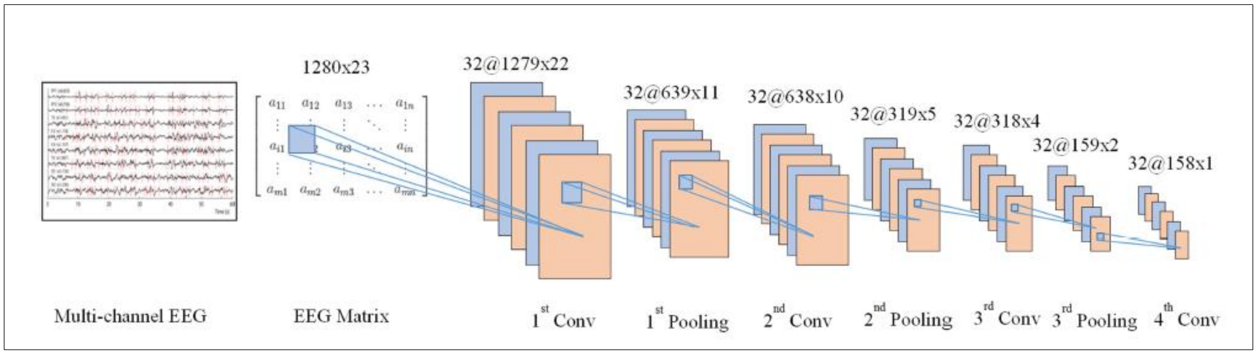
MLP

A multi-layer perceptron model was created as described in the paper. It consisted of 4 fully connected hidden layers with 300, 100, 50, 20 hidden units respectively. The final layer is an output layer with sigmoid activation. Adam optimizer and binary cross entropy values are used in the model.



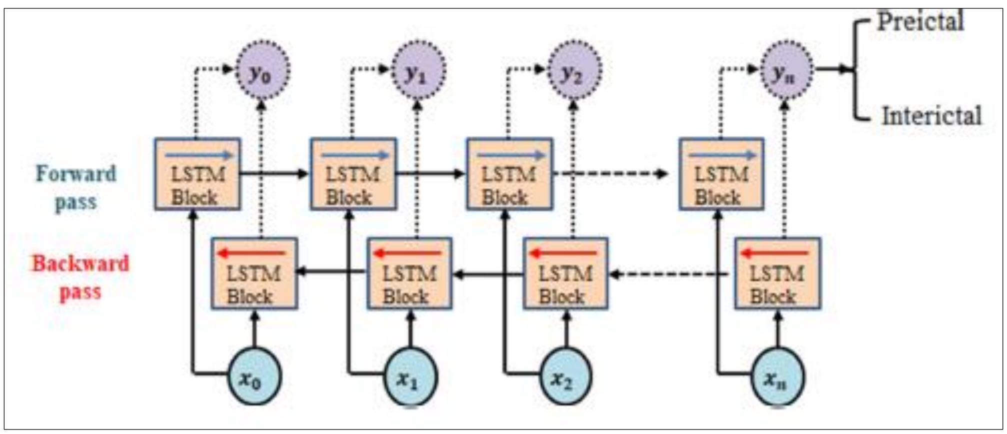
CNN + MLP

In the next stage a convolution neural network is created with a multi-layer perceptron layer at the end. The same adam optimizer and binary cross entropy loss is used. The dimensions of the convolution and the pooling layers that are selected is given as follows. The MLP layers at the end are not the MLP layers implemented previously. The model specifications are given below.



CNN + BiLSTM

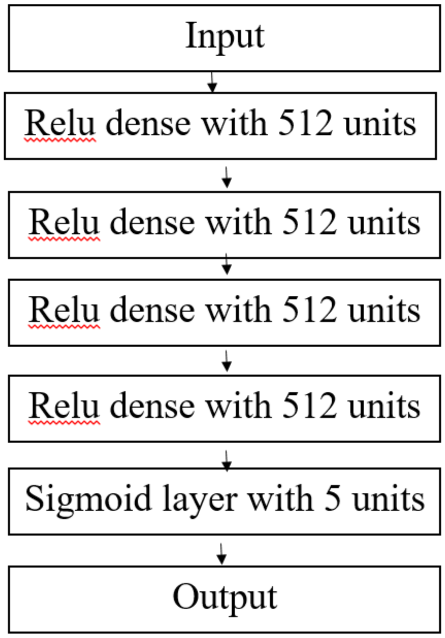
In the next stage the BiLSTM model is implemented. In this recurrent model, the inputs are fed into the model recurrently and the final obtained answer makes the prediction. As mentioned in the paper, the LSTM model is implemented after the CNN layers. A dropout of 10% tot the input layer and 50% to the recurrent layers is added. The optimizer used is RMSProp and loss is binary cross entropy. The model specifications are as follows.



Other hyper tuned models developed

MLP

Hyper tuning the proposed models led to the development of new model of MLP. The same optimizer and loss were used but the proposed model was hyper tuned to make it suitable for larger datasets and for better accuracy.



CNN + MLP

Hyper tuning the proposed models led to the development of new model of CNN + MLP. The same optimizer and loss were used but the proposed model was hyper tuned to make it suitable for larger datasets and for better accuracy. The summary of the new CNN model is as follows.

```
Model: "sequential_1"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
conv2d_1 (Conv2D)            (None, 1278, 21, 32)      896
_________________________________________________________________
max_pooling2d_1 (MaxPooling2 (None, 639, 10, 32)       0
_________________________________________________________________
dropout_1 (Dropout)          (None, 639, 10, 32)       0
_________________________________________________________________
conv2d_2 (Conv2D)            (None, 637, 8, 32)        9248
_________________________________________________________________
max_pooling2d_2 (MaxPooling2 (None, 318, 4, 32)        0
_________________________________________________________________
dropout_2 (Dropout)          (None, 318, 4, 32)        0
_________________________________________________________________
conv2d_3 (Conv2D)            (None, 316, 2, 32)        9248
_________________________________________________________________
max_pooling2d_3 (MaxPooling2 (None, 158, 1, 32)        0
_________________________________________________________________
dropout_3 (Dropout)          (None, 158, 1, 32)        0
_________________________________________________________________
flatten_1 (Flatten)          (None, 5056)              0
_________________________________________________________________
dense_1 (Dense)              (None, 512)               2589184
_________________________________________________________________
dense_2 (Dense)              (None, 1)                 513
=================================================================
Total params: 2,609,089
Trainable params: 2,609,089
Non-trainable params: 0
_________________________________________________________________
```

CNN + LSTM

LSTM models developed with the given specifications attained the accuracy proposed in the paper. Thus LSTM models were considered. Hyper tuning the proposed models led to the development of new model of CNN + LSTM. The same optimizer and loss were used but the proposed model was hyper tuned to make it suitable for larger datasets and for better accuracy. The summary of the new model is as follows.

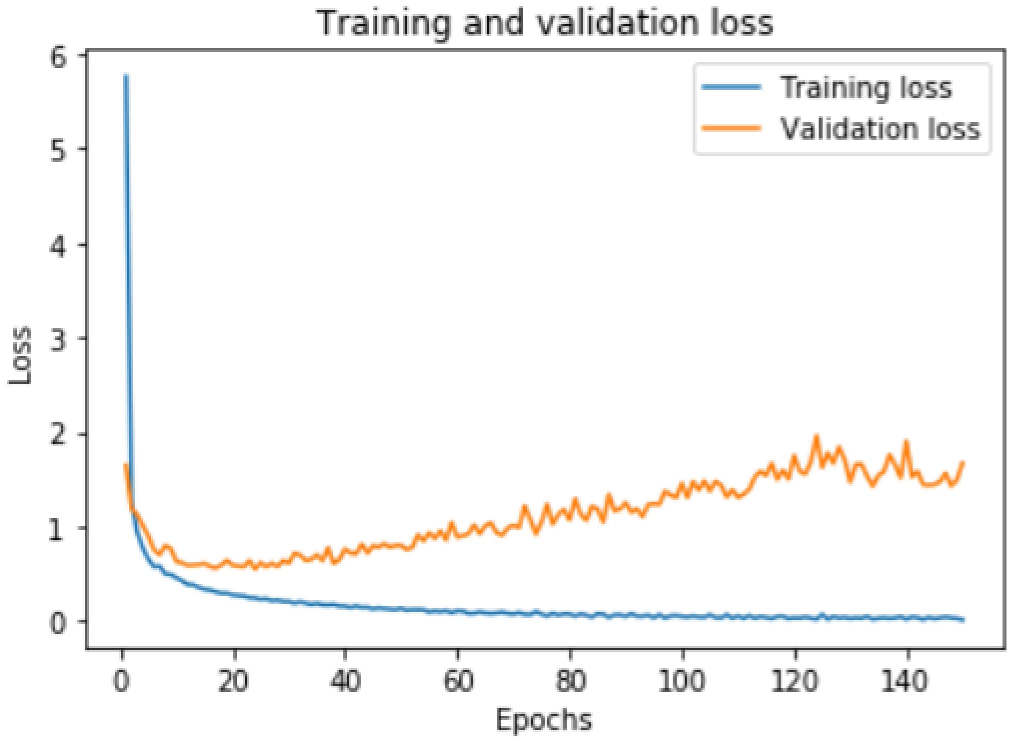
```
Model: "model_1"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
input_1 (InputLayer)         (None, 178, 1)            0
_________________________________________________________________
conv1d_1 (Conv1D)            (None, 176, 32)           128
_________________________________________________________________
conv1d_2 (Conv1D)            (None, 174, 32)           3104
_________________________________________________________________
max_pooling1d_1 (MaxPooling1 (None, 87, 32)            0
_________________________________________________________________
conv1d_3 (Conv1D)            (None, 85, 32)            3104
_________________________________________________________________
max_pooling1d_2 (MaxPooling1 (None, 42, 32)            0
_________________________________________________________________
conv1d_4 (Conv1D)            (None, 40, 32)            3104
_________________________________________________________________
max_pooling1d_3 (MaxPooling1 (None, 20, 32)            0
_________________________________________________________________
conv1d_5 (Conv1D)            (None, 18, 32)            3104
_________________________________________________________________
gaussian_dropout_1 (Gaussian (None, 18, 32)            0
_________________________________________________________________
lstm_1 (LSTM)                (None, 9)                 1512
_________________________________________________________________
dropout_1 (Dropout)          (None, 9)                 0
_________________________________________________________________
dense_1 (Dense)              (None, 25)                250
_________________________________________________________________
dense_2 (Dense)              (None, 2)                 52
=================================================================
Total params: 14,358
Trainable params: 14,358
Non-trainable params: 0
```

VII. Evaluation Metrics

Evaluation metrics are used to determine the performance of the developed models and the hyper tuned models. Various evaluation metrics are used to know the accuracy of the models. Thus a detailed comparison is available to verify the models.

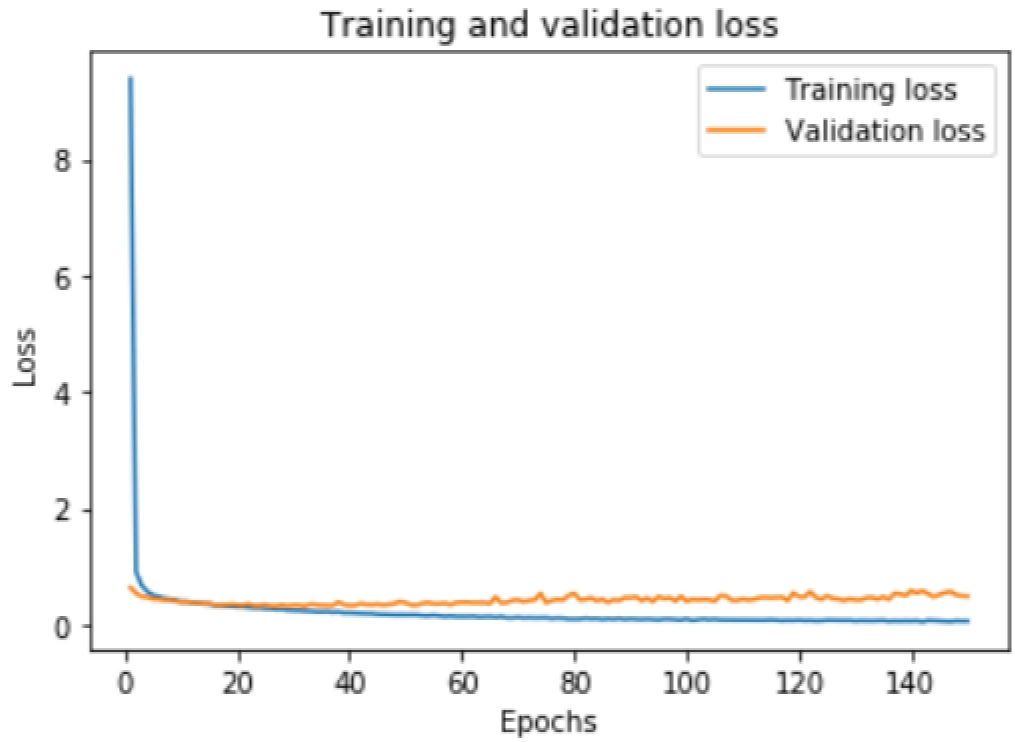
MLP models

Model Proposed in the Paper



Accuracy against the test dataset - `0.836927592754364`
Specificity – `0.9655757546424866`
Sensitivity - `0.9691815972328186`

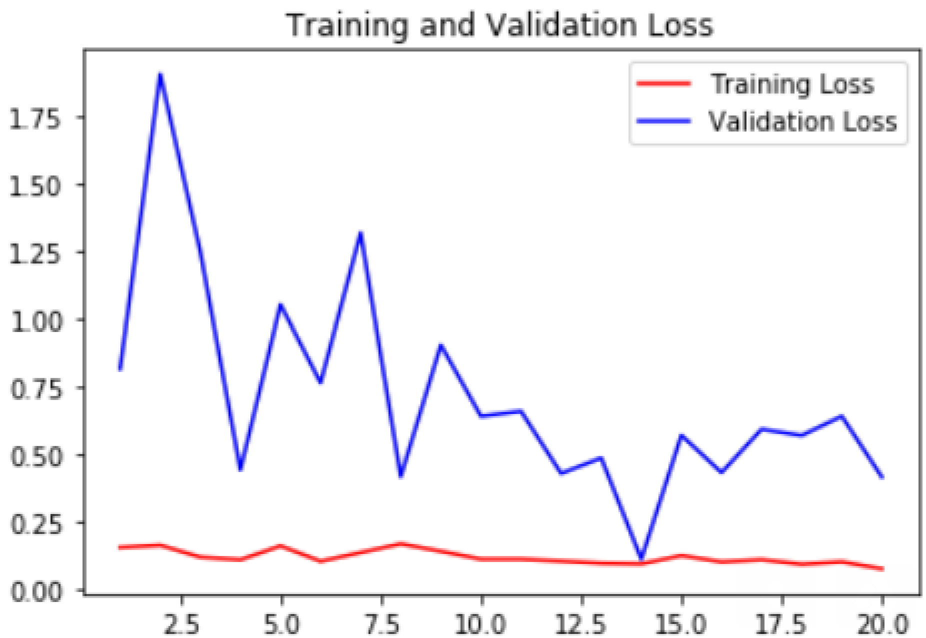
Hyper tuned model



Accuracy against the test dataset - `0.873971164226532`
Specificity – `0.9777584075927734`
Sensitivity - `0.9774889349937439`

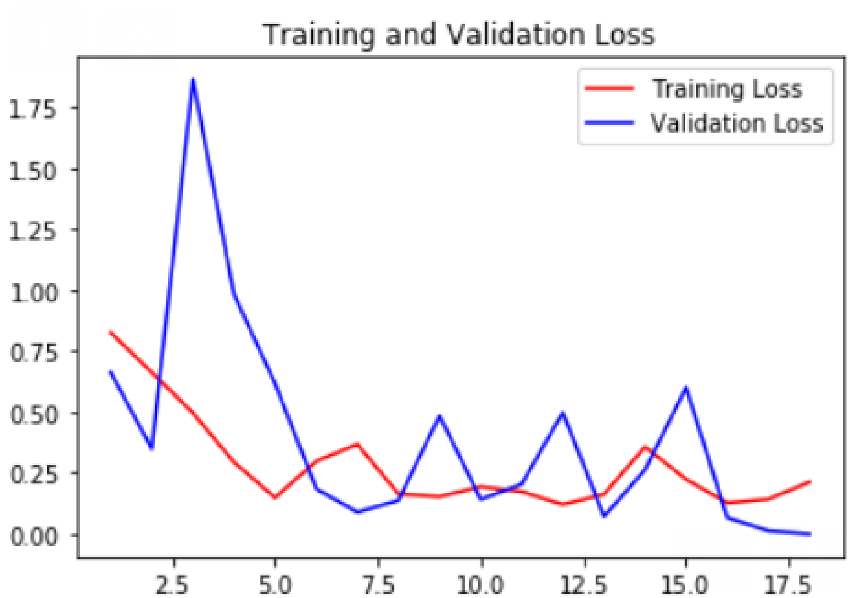
CNN models

Model Proposed in the Paper



Accuracy against test dataset - `0.8571428656578064`
Specificity – `0.7809429168701172`
Sensitivity - `0.868950605392456`

Hyper tuned model

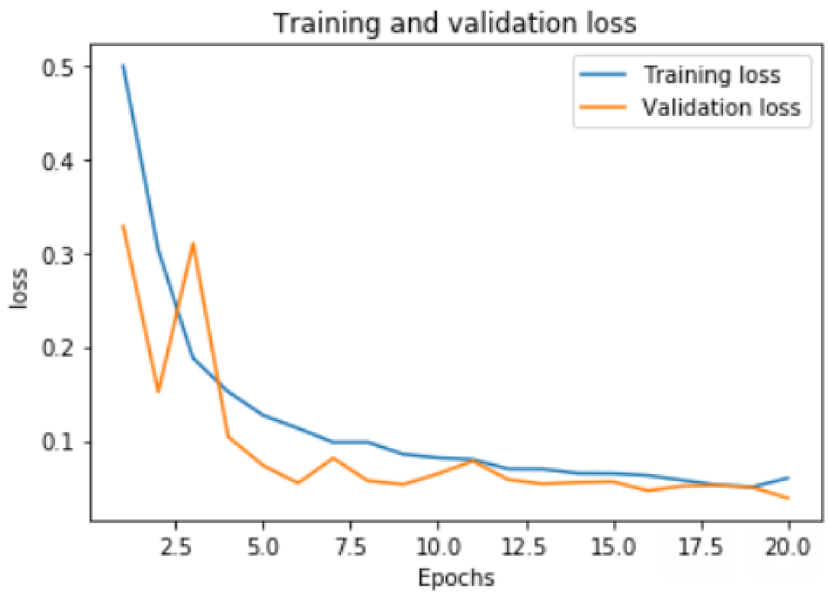


Accuracy against test dataset - `0.8421052694320679`
Specificity – `0.9600200057029724`
Sensitivity - `0.9991479516029358`

LSTM models

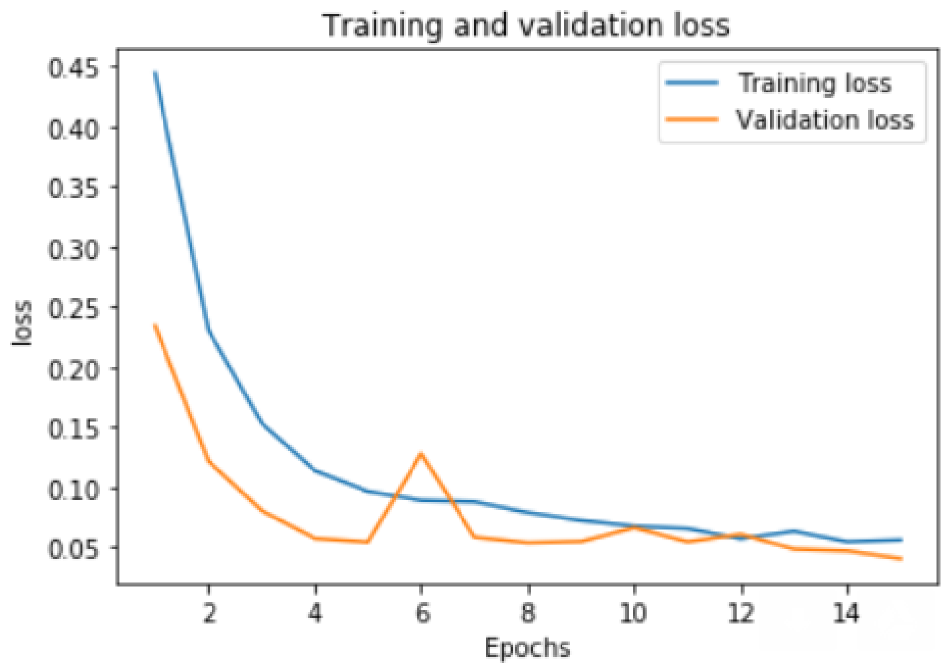
LSTM models developed with the given specifications attained the accuracy proposed in the paper. Thus LSTM models were considered.

LSTM in accordance to Model Proposed in the Paper



Accuracy against test dataset - `0.9930434823036194`
Specificity – `0.9975911378860474`
Sensitivity - `0.9976343512535095`

Hyper tuned model



Accuracy against test dataset - `0.9905797243118286`
Specificity – `0.9976755976676941`
Sensitivity - `0.9976755976676941`

VIII. Conclusion

From the above comparisons certain conclusions are made as follows.

- It is obvious that the models proposed in the paper were not suitable for a larger datasets.
- It is found that the proposed accuracy was not met in the MLP and CNN models. Whereas in the LSTM models both the proposed and the hyper tuned models produced nearly the same accuracy (nearing 100%).
- Thus it can be concluded that a LSTM model produces more accuracy than any other models and extracts more features.
- In all the cases, the LSTM models have lesser parameters than CNN and MLP.

IX. Implementation Limitations

Certain difficulties found during the implementation of the programs are as follows:
Implemented using anaconda and jupyter.
The issues included

1. Large datasets took a longer time.
2. Time consumption due to evaluation against all optimizers and loss.
3. The code was ran on a CPU that consumed more time.

4. Not every model was saved because the models were misinterpreted by the jupyter kernel. Thus only the best model code is provided.
5. Since it was difficult to save multiple models ensemble classifiers were not implemented.

X. Future Enhancement

Certain future enhancements that can be done to the projects are:

1. A better model for a much larger dataset can be developed.
2. Better deep learning models can be implemented to get a better accuracy faster and with less parameters.

XI. References

[1] Efficient Epileptic Seizure Prediction based on Deep Learning by Hisham Daoud and Magdy A. Bayoumi published in IEEE Transactions on Biomedical Circuits and Systems, October 2019.

[2] Performance Analysis of Epileptic Seizure Detection System Using Neural Network Approach by R.Vaitheeshwari (Department of Electronics Engineering, Madras Institute of Technology, Anna University-Chennai), and V.Sathiesh Kumar (Department of Electronics Engineering, Madras Institute of Technology, Anna University-Chennai) published in Second International Conference on Computational Intelligence in Data Science (ICCIDS-2019)

[3] https://keras.io/

[4] https://medium.com/dabbler-in-de-stress/detecting-epileptic-seizures-for-eeg-data-56e6103bb591

[5] Code given by the Professor.