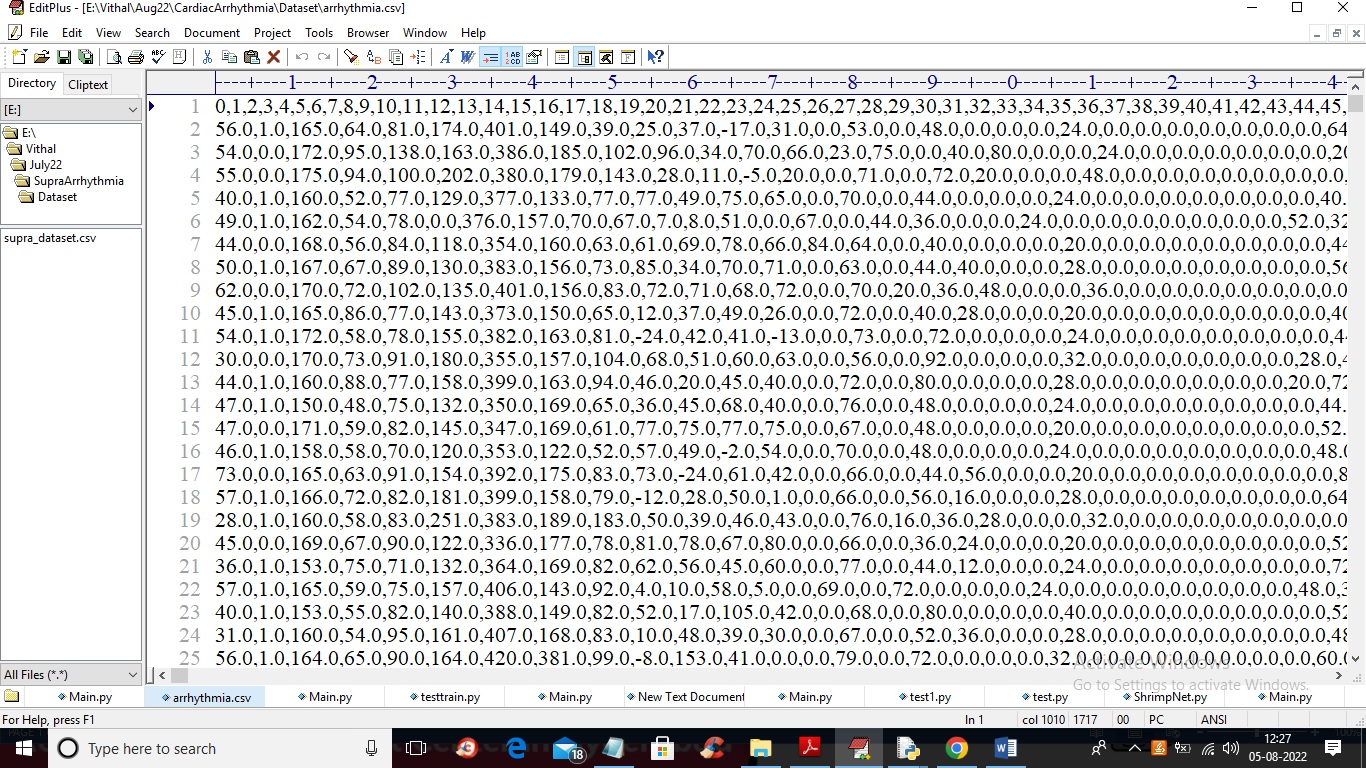
Automated Detection of Cardiac Arrhythmia using Recurrent Neural Network

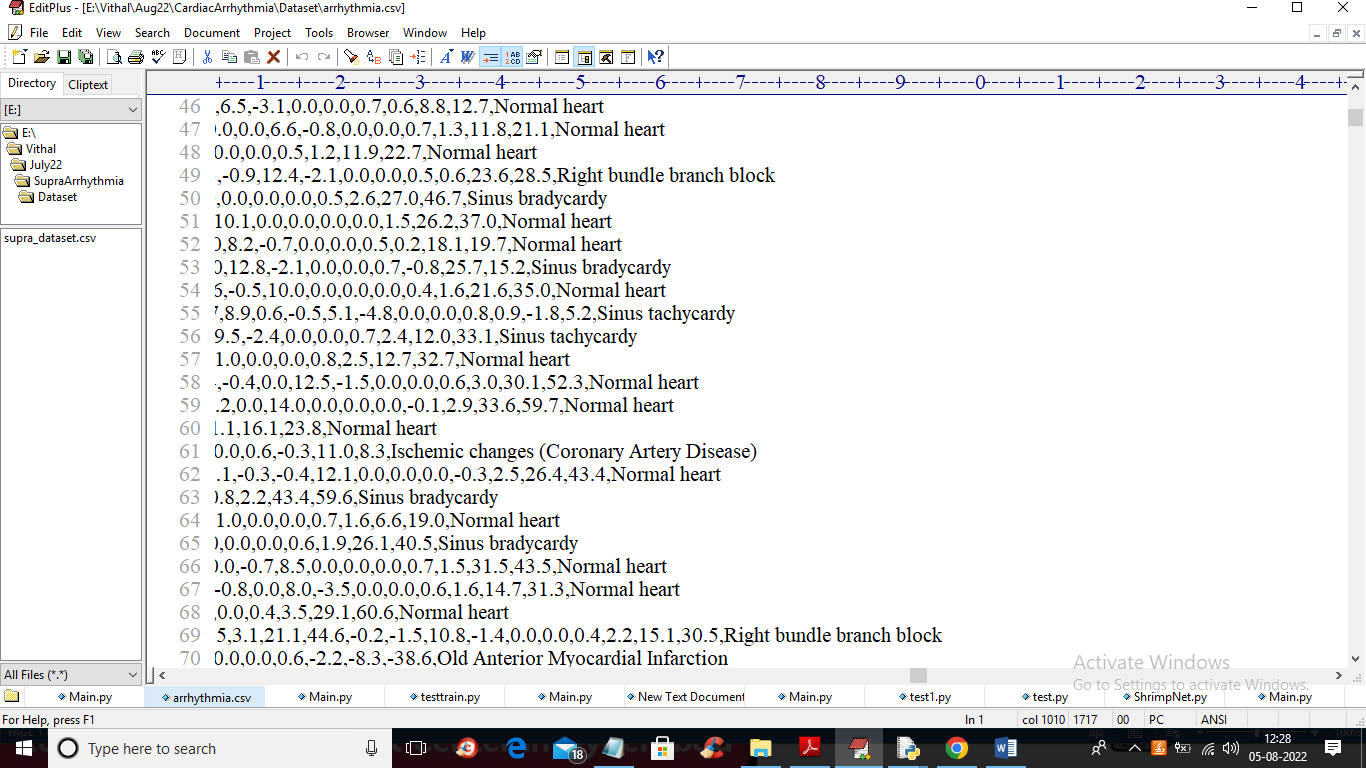
In this project you asked to design CNN and LSTM algorithm to predict Arrhythmia diseases with 7 different stages. To train both algorithm we have used MIT-BH dataset with 7 different disease stages.

You told to get 95% accuracy which is difficult to get with LSTM but with CNN we got 97% accuracy.

To train both algorithm we have used below dataset



In above dataset screen first row contains column names and remaining rows contains dataset values and in last column we can see class label as disease name



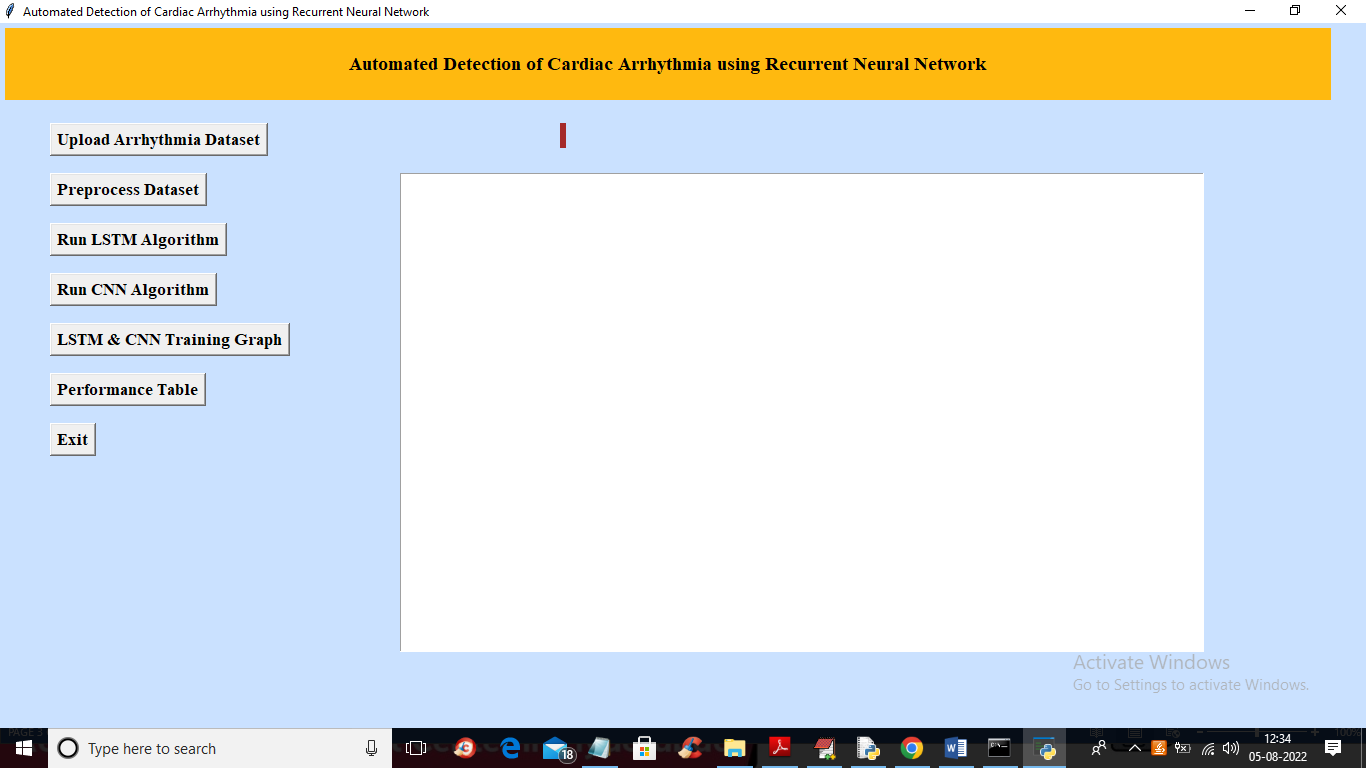
In above screen in last column we can see disease name and by using above dataset we are training both algorithms

To implement this project we have designed following modules

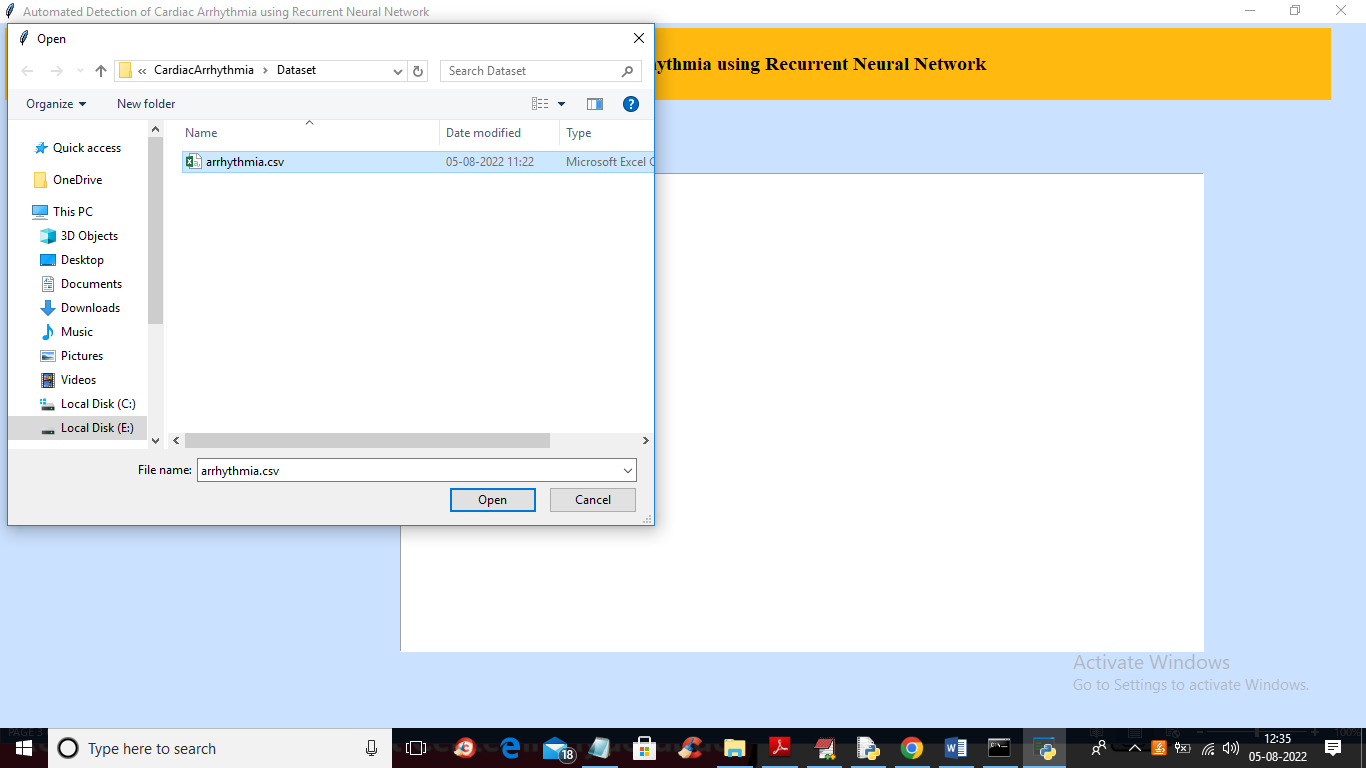
1. Upload Arrhythmia Dataset: using this module we will upload dataset to application
2. Preprocess Dataset: using this module we will read all dataset values and then replace missing values with MEAN and then normalize training values and then selected important features from dataset by applying PCA algorithm. Dataset contains more than 270 columns and all this columns are not required so by using PCA we selected relevant features from dataset. After features selection we have splitted dataset into train and test where application using 80% dataset for training and 20% for testing
3. Run LSTM Algorithm: we will input 80% training data to LSTM to trained a model and then model will be applied on 20% test data to perform prediction and then calculate accuracy
4. Run CNN Algorithm: we will input 80% training data to CNN to trained a model and then model will be applied on 20% test data to perform prediction and then calculate accuracy
5. LSTM & CNN Training Graph: using this module we will plot CNN and LSTM training graph
6. Performance Table: using this module we will display both algorithms performance in tabular format.

SCREEN SHOTS

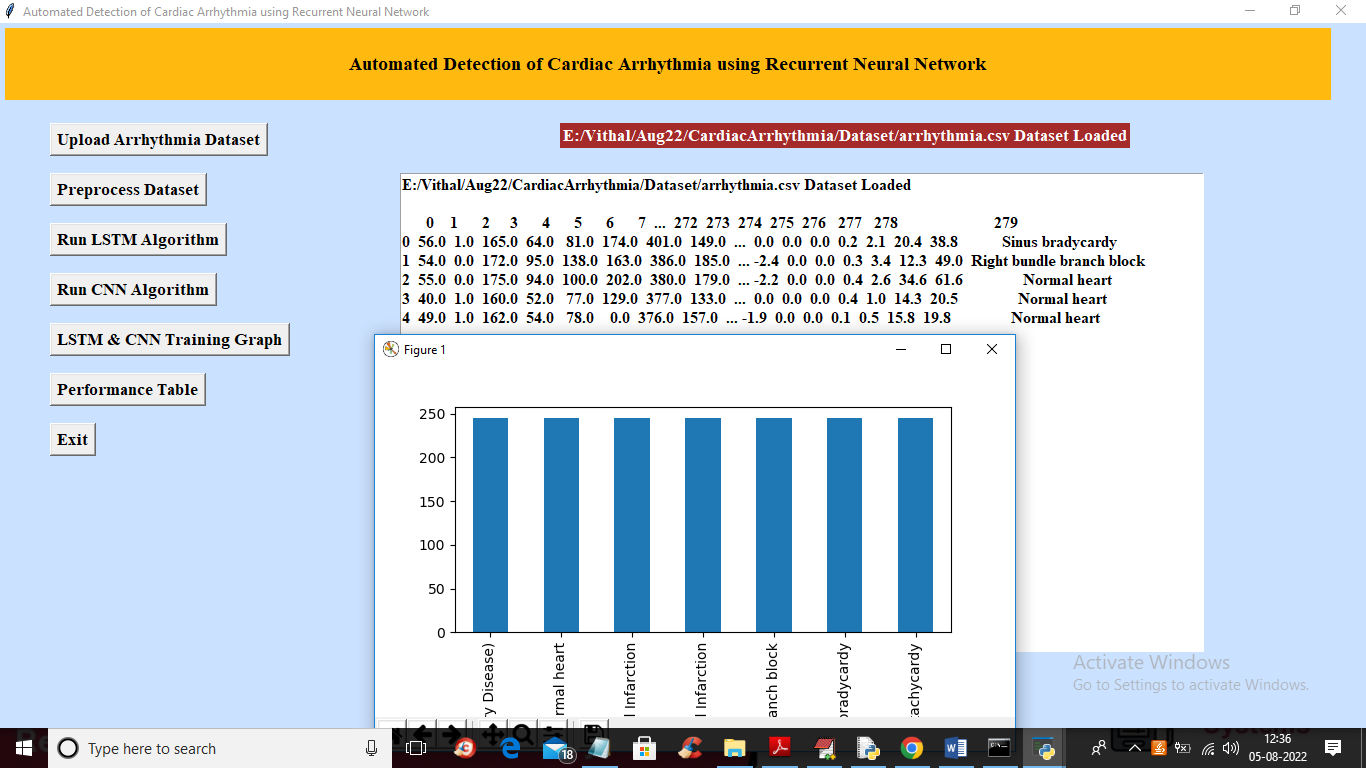
To run project double click on ‘run.bat’ file to get below screen



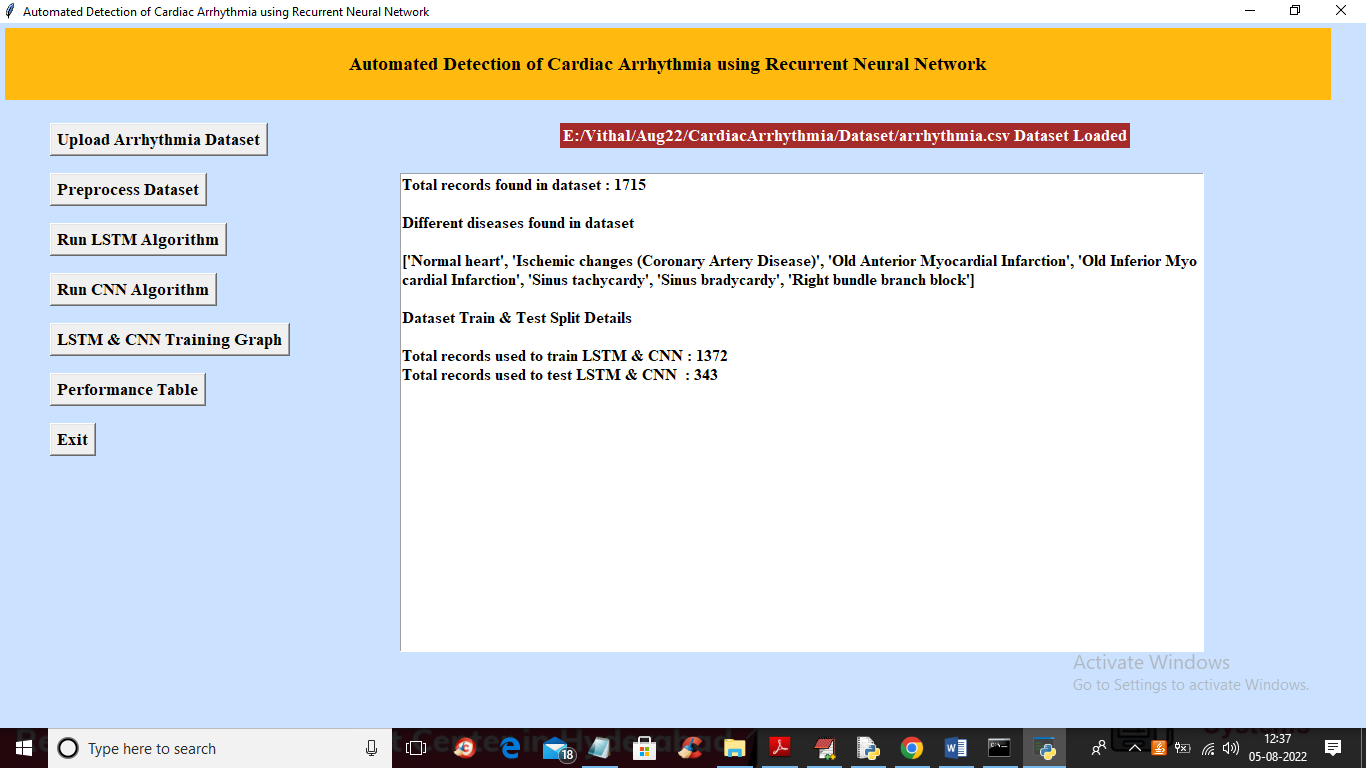
In above screen click on ‘Upload Arrhythmia Dataset’ button to upload dataset and get below output



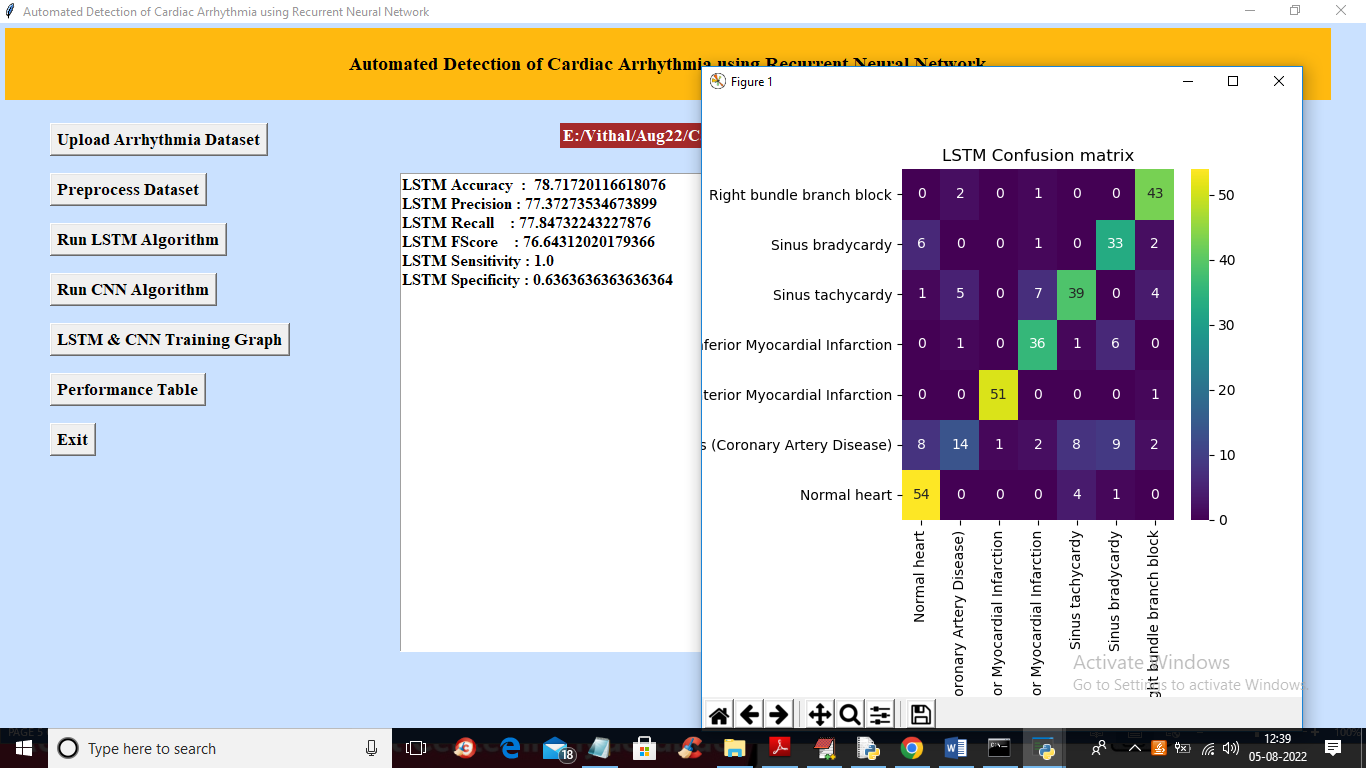
In above screen selecting and uploading ‘Arrhythmia’ dataset and then click on ‘Open’ button to load dataset and get below output



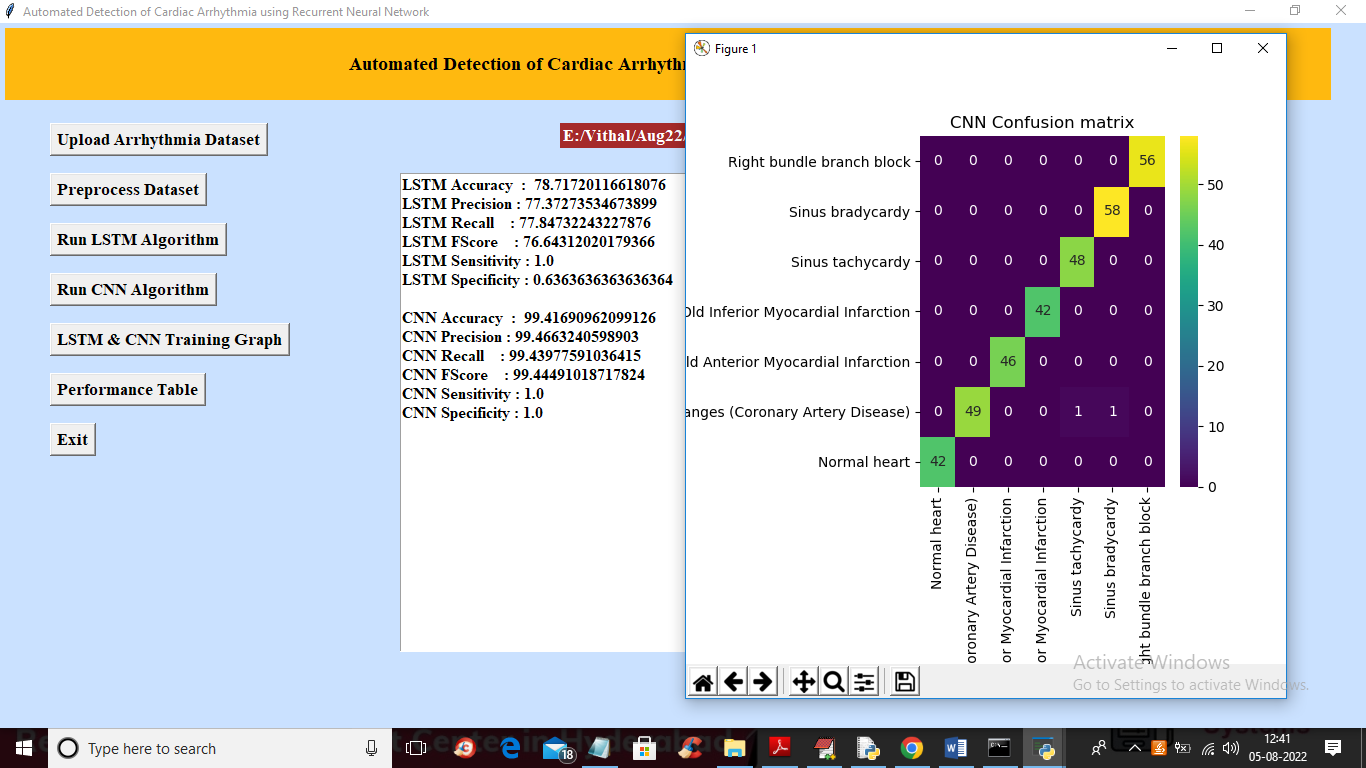
In above screen we can see dataset loaded and in graph x-axis represents 7 different disease stages and y-axis represents number of records found for that disease in dataset and in above screen we can see dataset contains some non-numeric values but algorithm accept only numeric values so close above graph and then click on ‘Preprocess Dataset’ button to process dataset and then split into train and test



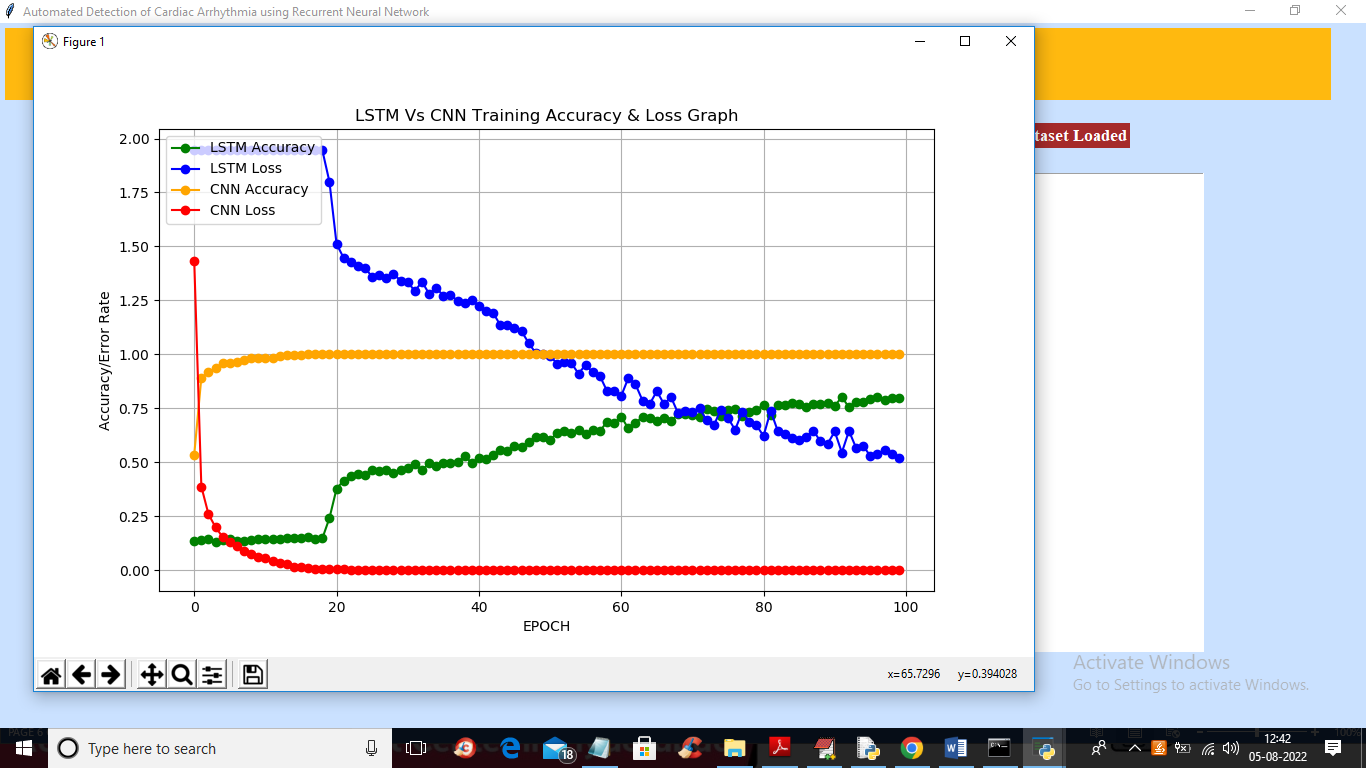
In above screen all dataset converted to numeric format and we can see total dataset size with train and test split details and displaying names of disease and now click on ‘Run LSTM Algorithm’ button to train LSTM with above process dataset and get below output



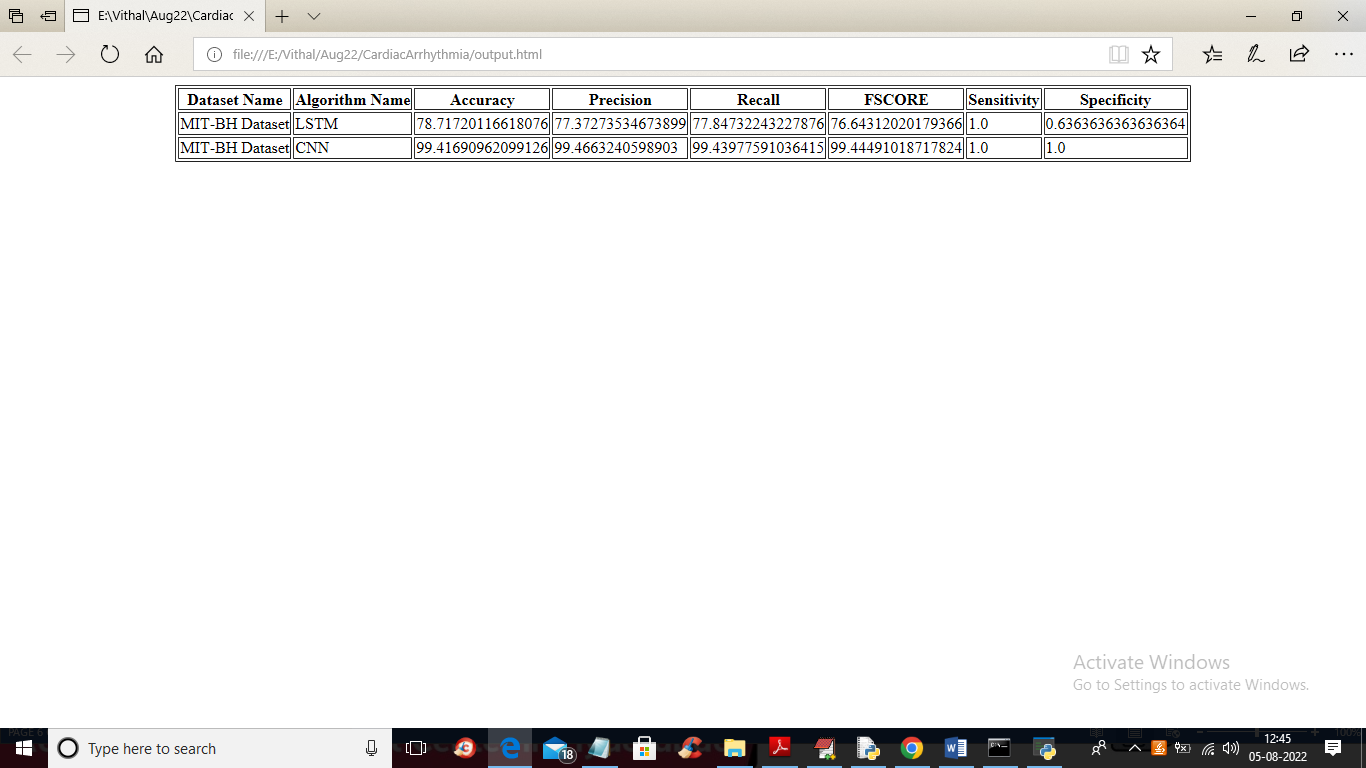
In above screen with LSTM we got 78% accuracy and in confusion matrix graph x-axis represents Predicted classes and y-axis represents TRUE classes and all blue colour boxes count are wrong prediction and different colour boxes count are correct prediction and we can see LSTM predicted so many wrong classes and now close above graph and then click on ‘Run CNN Algorithm’ button to train CNN and get below output



In above screen with CNN we got 99% accuracy and in confusion matrix graph only 2 counts in blue colour boxes are wrong prediction and rest are correct prediction. Now click on ‘LSTM & CNN Training Graph’ button to get below graph.



In above graph x-axis represents training epoch and y-axis represents training accuracy and loss values and green colour line represents LSTM accuracy and orange colour line represents CNN accuracy and red colour line represents CNN loss and blue line represents LSTM loss and in above graph we can see both algorithms accuracy got increase in every epoch and loss get decrease and now close above graph and then click on “Performance Table’ button to get below output



In above screen we can see output metrics of both algorithms in tabular format