A Novel Wireless Network Intrusion Detection Method Based on Adaptive Synthetic Sampling and an Improved Convolutional Neural Network

Now a day’s network connectivity is everywhere and malicious attackers are using this advantage to attack all types of network connectivity servers and user’s system to steal data or to disturb network activities. There are many existing algorithms are available but their accurate detection rate is less and false detection rate is high because of diversity or change of techniques used by attackers.

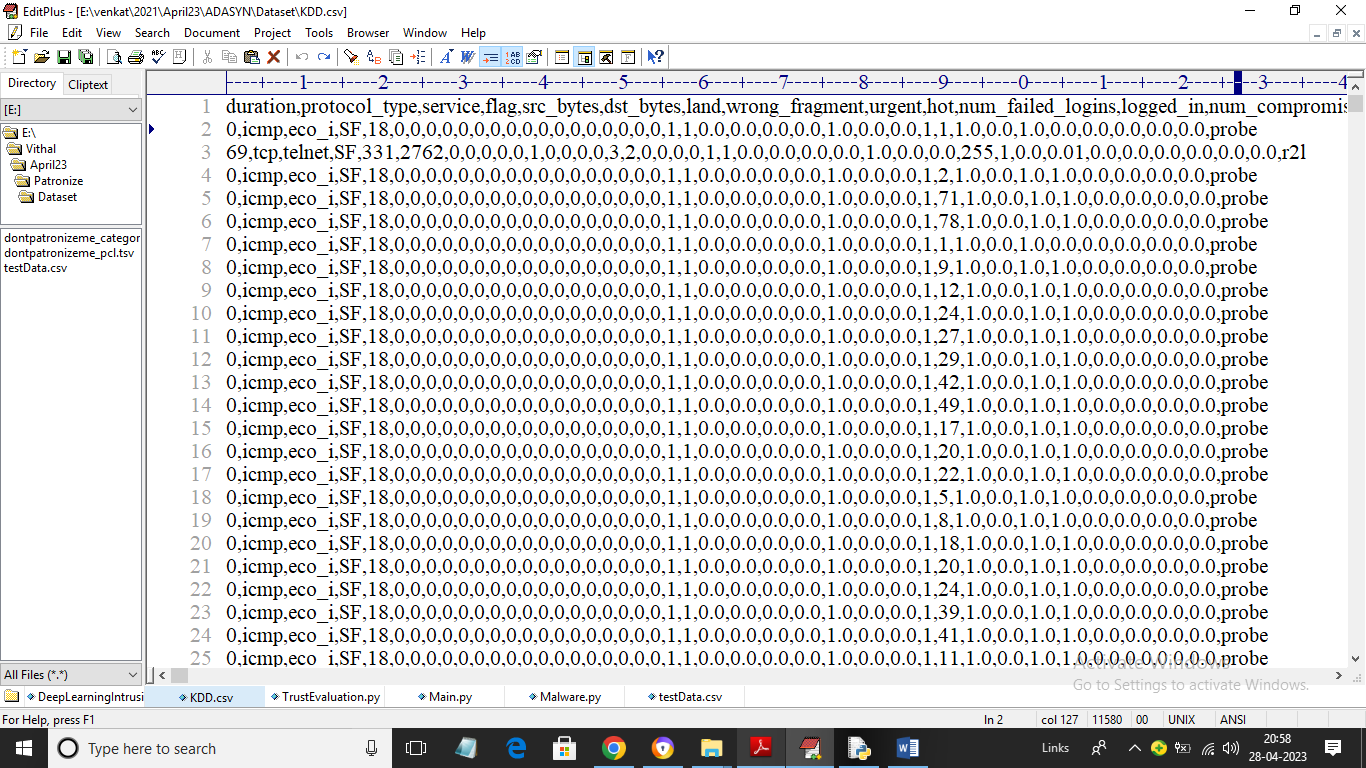
Most of the time existing algorithms detection rate is highly impact because of data imbalance where one class contains high number of instances and other class contains fewer instances and this data imbalance (small or large samples) always confuse algorithms for miss-classifications.

To overcome from above problem author of this paper employing Adaptive Synthetic Sampling (ADASYN) technique to augment data which means ADASYN will generate new synthetic instances for all those class labels which are having few instances and by applying this concept all class labels will have equal instances and algorithm will have less chances of miss-classifications which result into higher accuracy and less false alarm rate.

In propose paper author modifying CNN algorithm to remove all redundant and irrelevant features while training so it will get trained only on optimized and relevant features and will get better accuracy. Propose CNN algorithm removing redundant features is called as SPC-CNN.

Author has used NSL-KDD dataset to evaluate performance of propose SPC-CNN algorithm and it is comparing with existing algorithms called Naïve Bayes and SVM.

Below screen showing NSL-KDD dataset details



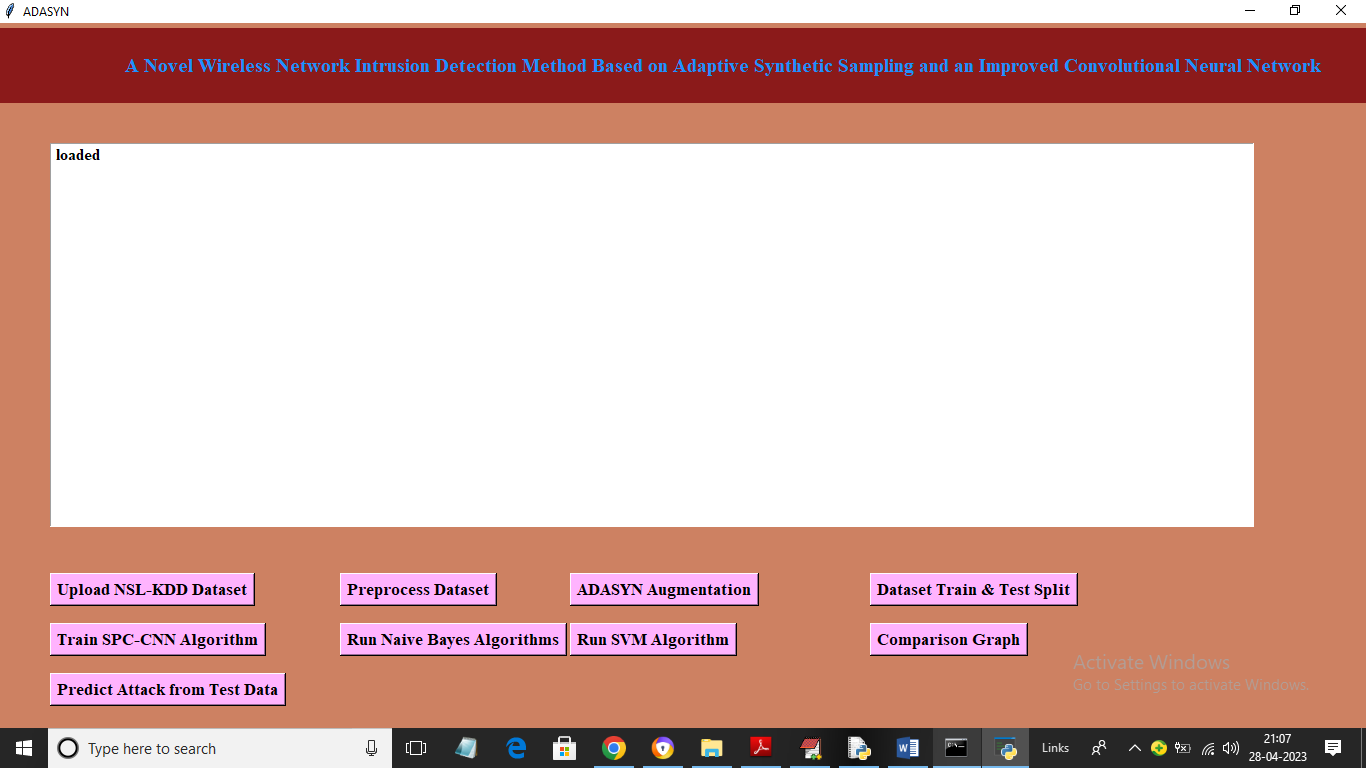
In above screen first row represents dataset column names and remaining rows represents dataset values and in last column we have attack class labels called DOS, R2L, U2R and Probe. So by using above dataset we will train and test performance of all algorithms

To implement this project we have designed following modules

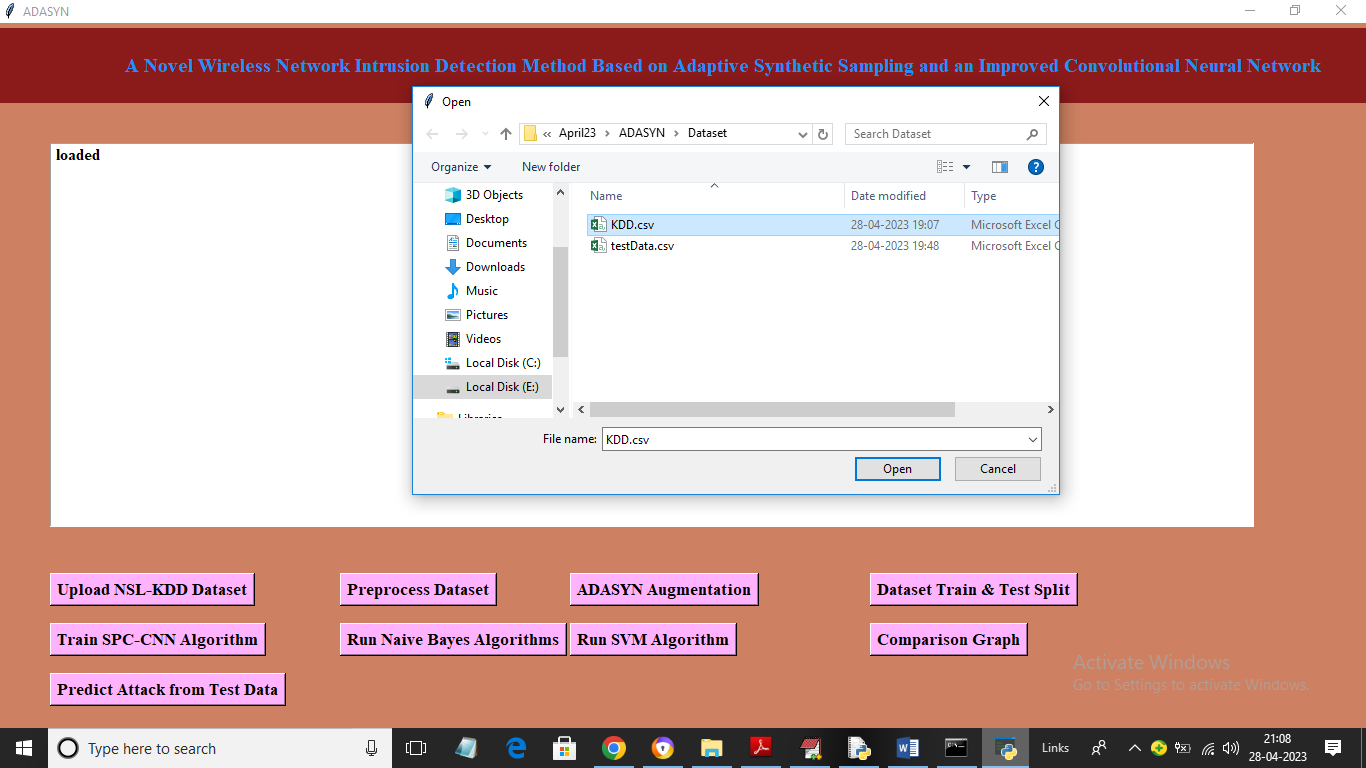
1. Upload NSL-KDD Dataset: using this module we will upload NSL-KDD dataset and then extract and plot graph of various attacks found in dataset
2. Preprocess Dataset: dataset contains non-numeric and numeric values but algorithms accept only numeric data so by using Preprocessing technique we will encode all non-numeric data into numeric data and then normalize, handle missing values and then shuffle dataset
3. ADASYN Augmentation: using this module we will apply augmentation on dataset to generate synthetic instances for low samples
4. Dataset Train & Test Split: using this module we will split dataset into train and test where application using 80% dataset for training algorithms and 20% dataset to test algorithm prediction capability
5. Train SPC-CNN Algorithm: using this module we will trained SPC-CNN algorithm on augmented data and then generate prediction model and this model will be applied on test data to calculate prediction accuracy:
6. Run Naïve Bayes Algorithm: using this module we will train and test performance of Naïve Bayes algorithm
7. Run SVM Algorithm: using this module we will train and test performance of SVM algorithm
8. Comparison Graph: using this module we will plot comparison graph between all algorithms
9. Predict Attack from Test Data: using this module we will upload test data and then propose SPC-CNN will analyse test data and then predict test data contains which attack

SCREEN SHOTS

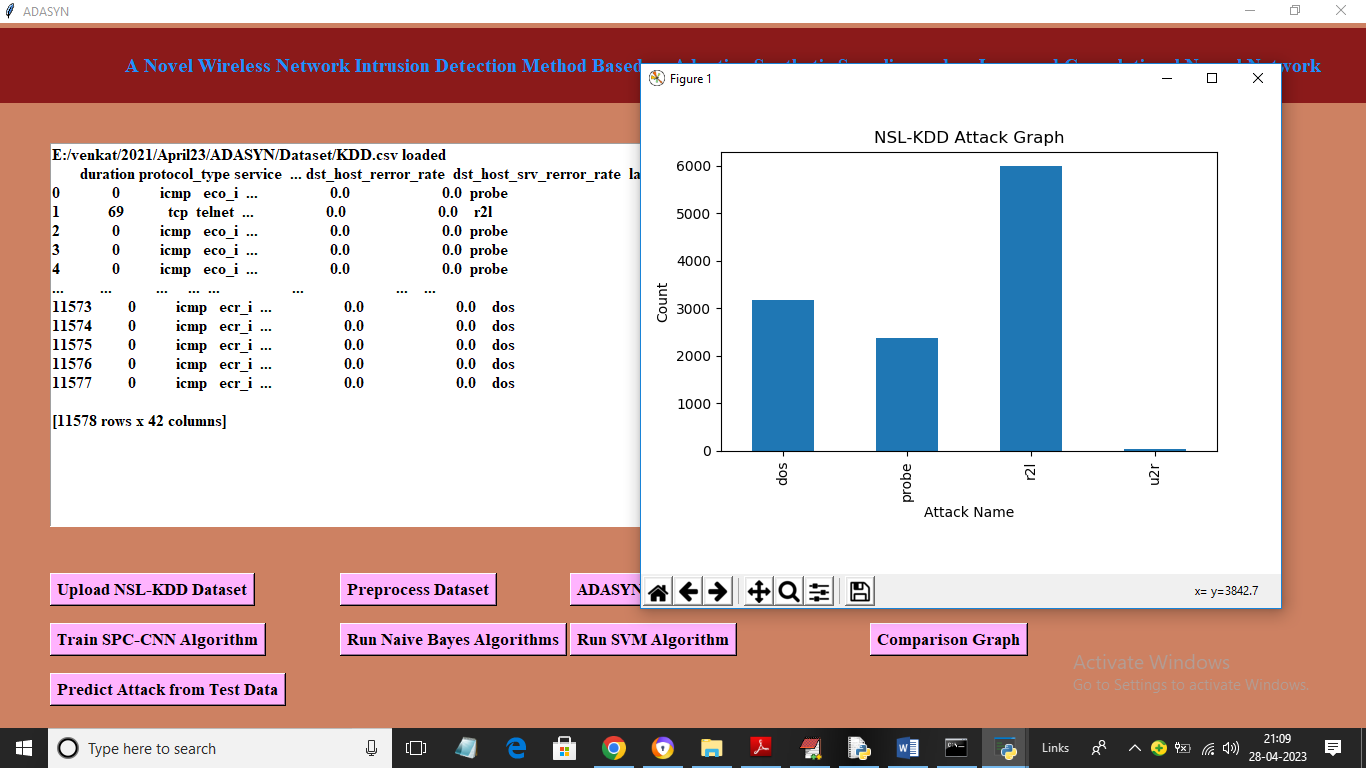
To run project double click on ‘run.bat’ file to get below screen



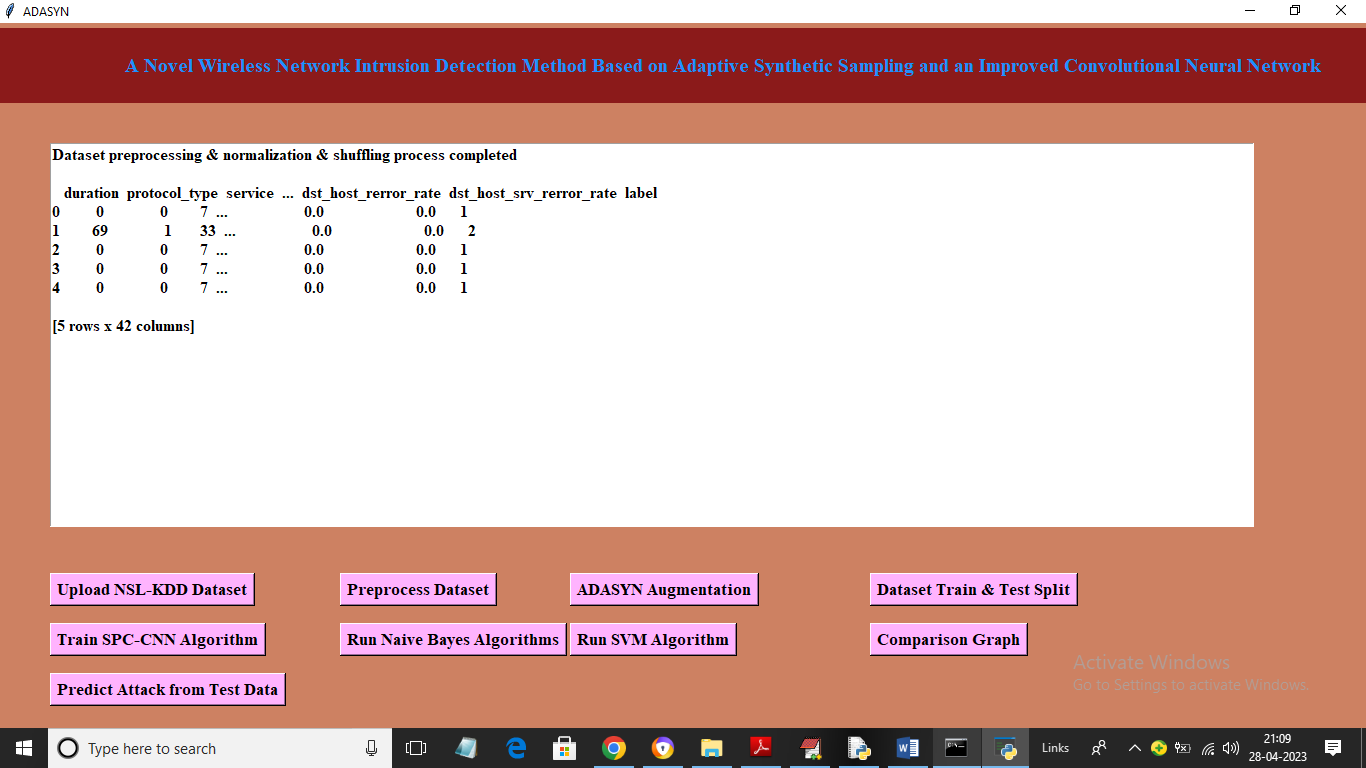
In above screen click on ‘Upload NSL-KDD Dataset’ button to upload dataset and get below screen



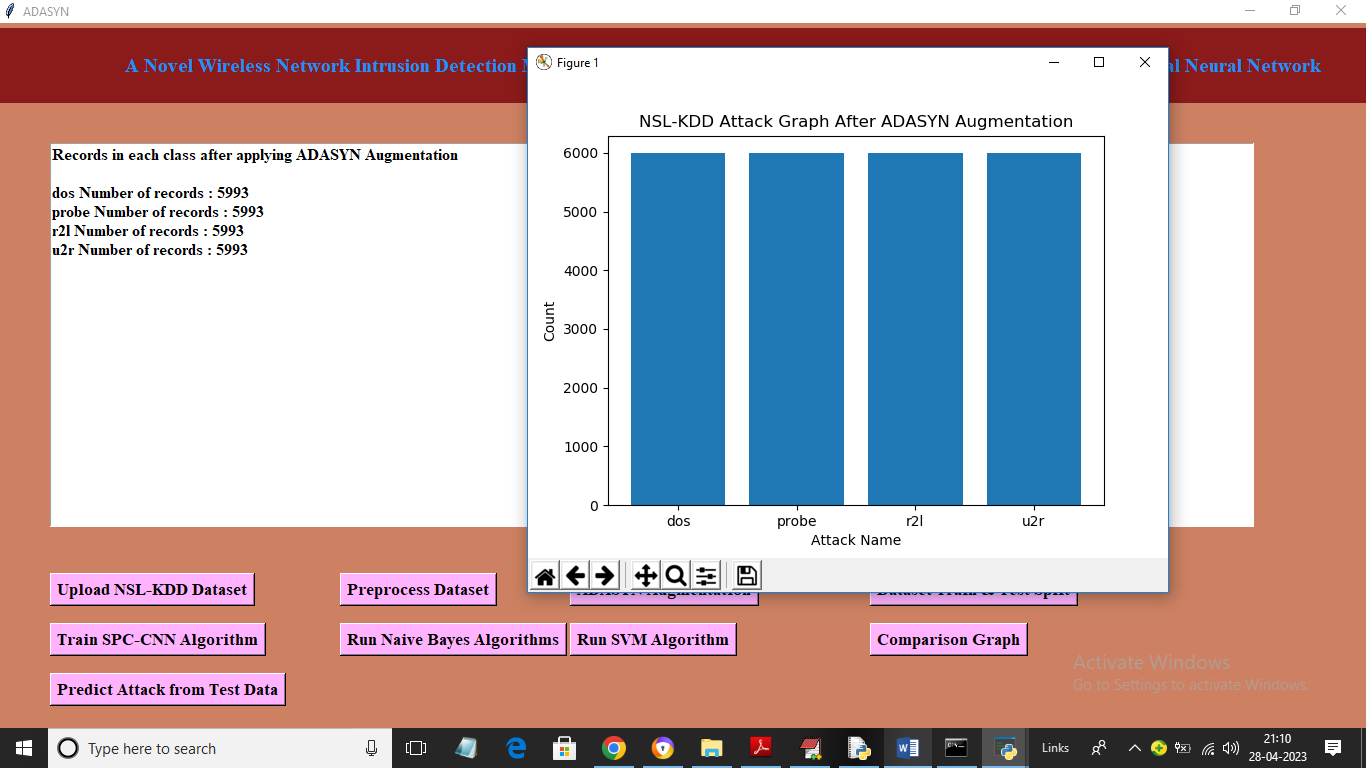
In above screen selecting and uploading ‘KDD.csv’ dataset file and then click on ‘Open’ button to load dataset and get below output



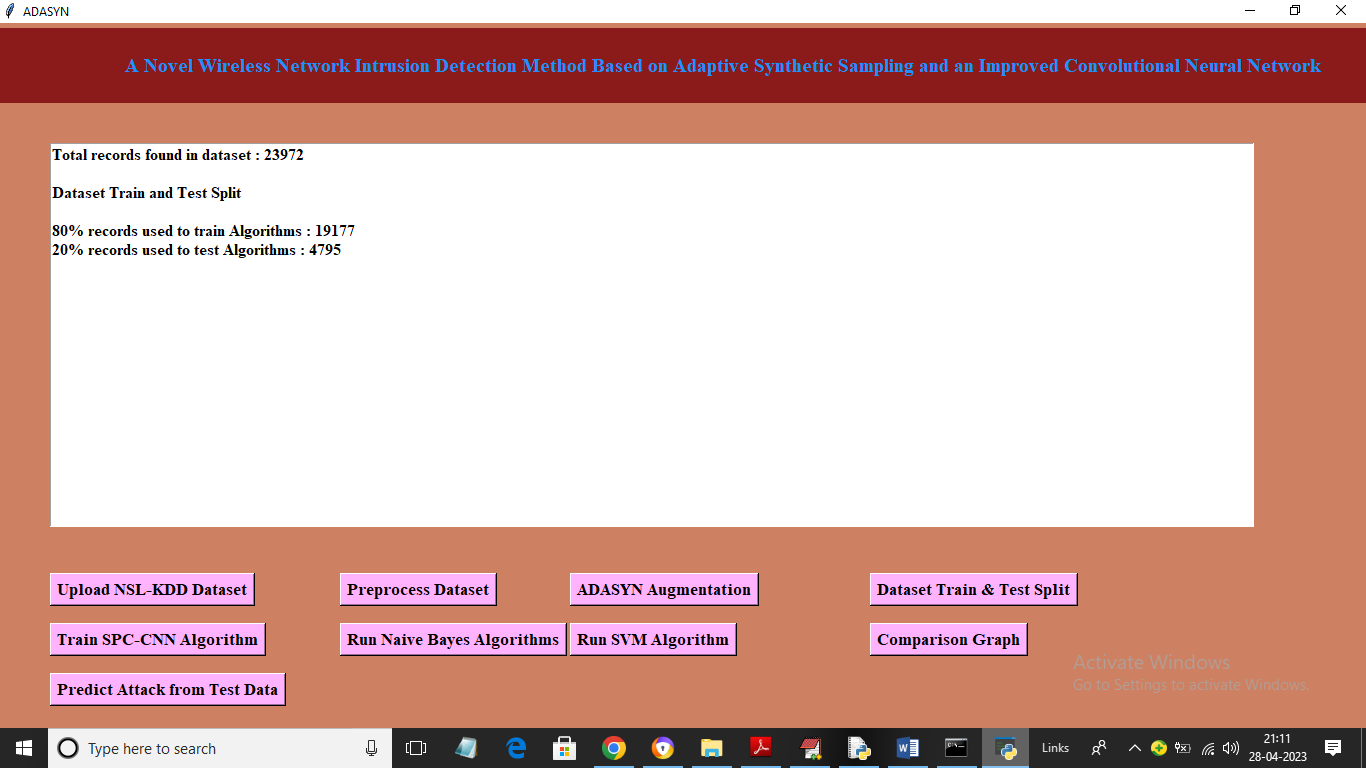
In above screen dataset loaded and in graph x-axis represents attack names and y-axis represents counts of records and we can see one attack has so many records and other attack is having very few records and in above dataset we can see some values are numeric and some are non-numeric so click on “Preprocess Dataset” button to process dataset and get below output



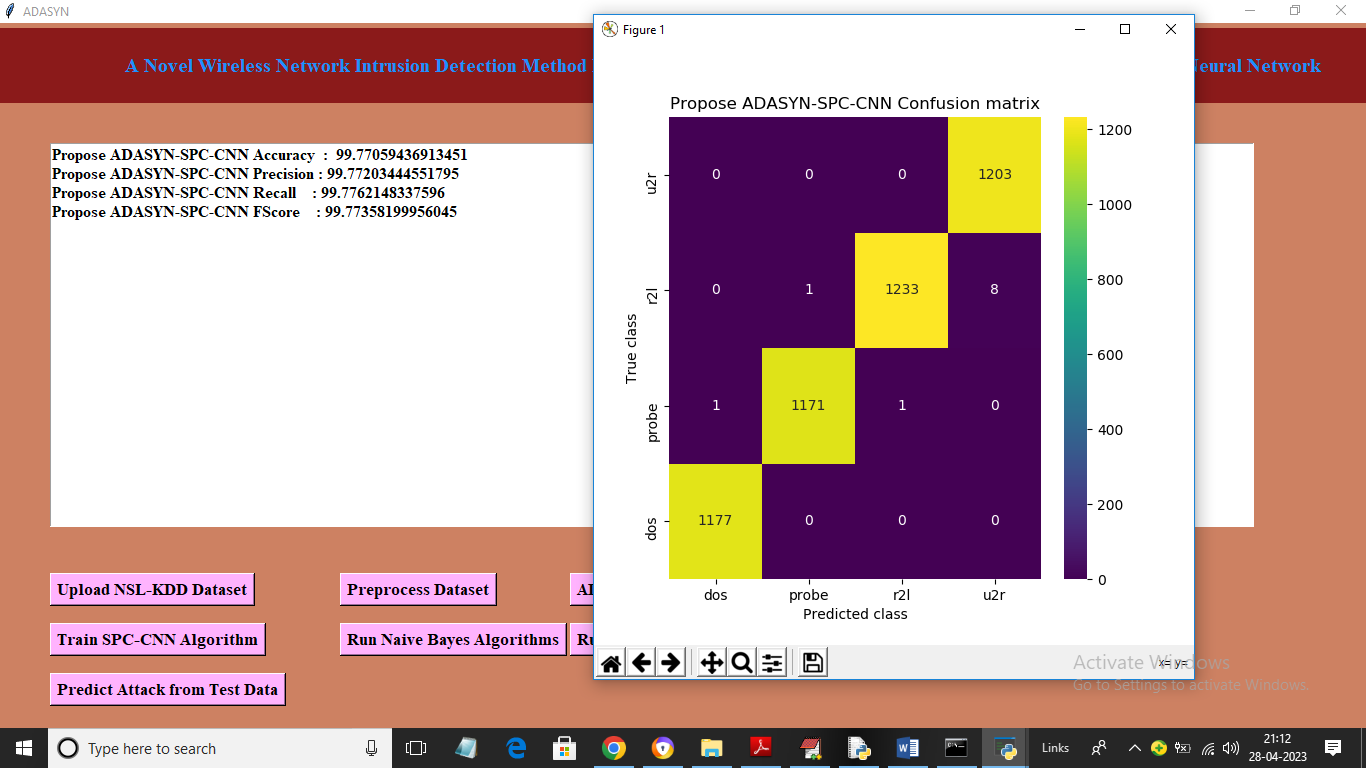
In above screen entire dataset processed and converted to numeric format and now click on ‘ADASYS Augmentation’ button to generate synthetic instances for low samples and get below output



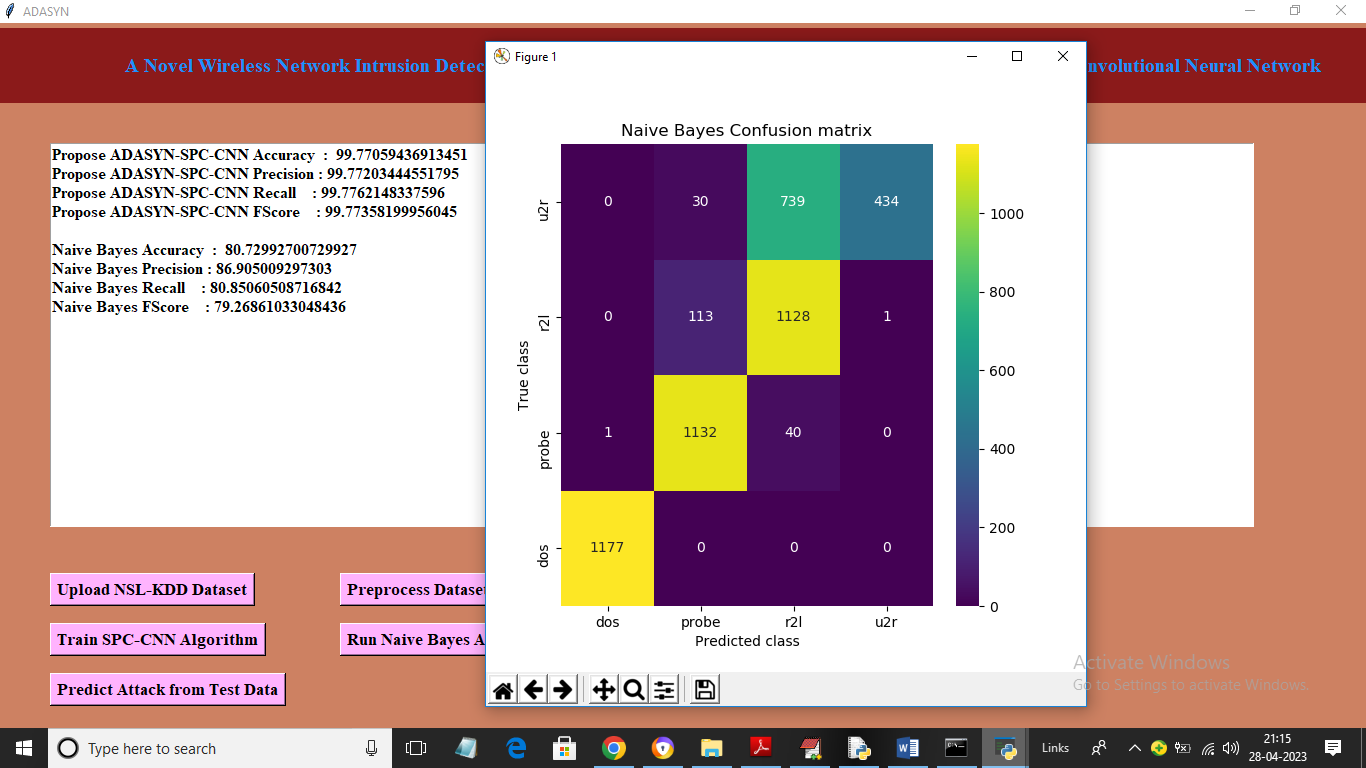
In above screen after augmentation all class labels are having equal number of instances and now click on ‘Dataset Train & Test Split’ button to split dataset into train and test and then will get below output



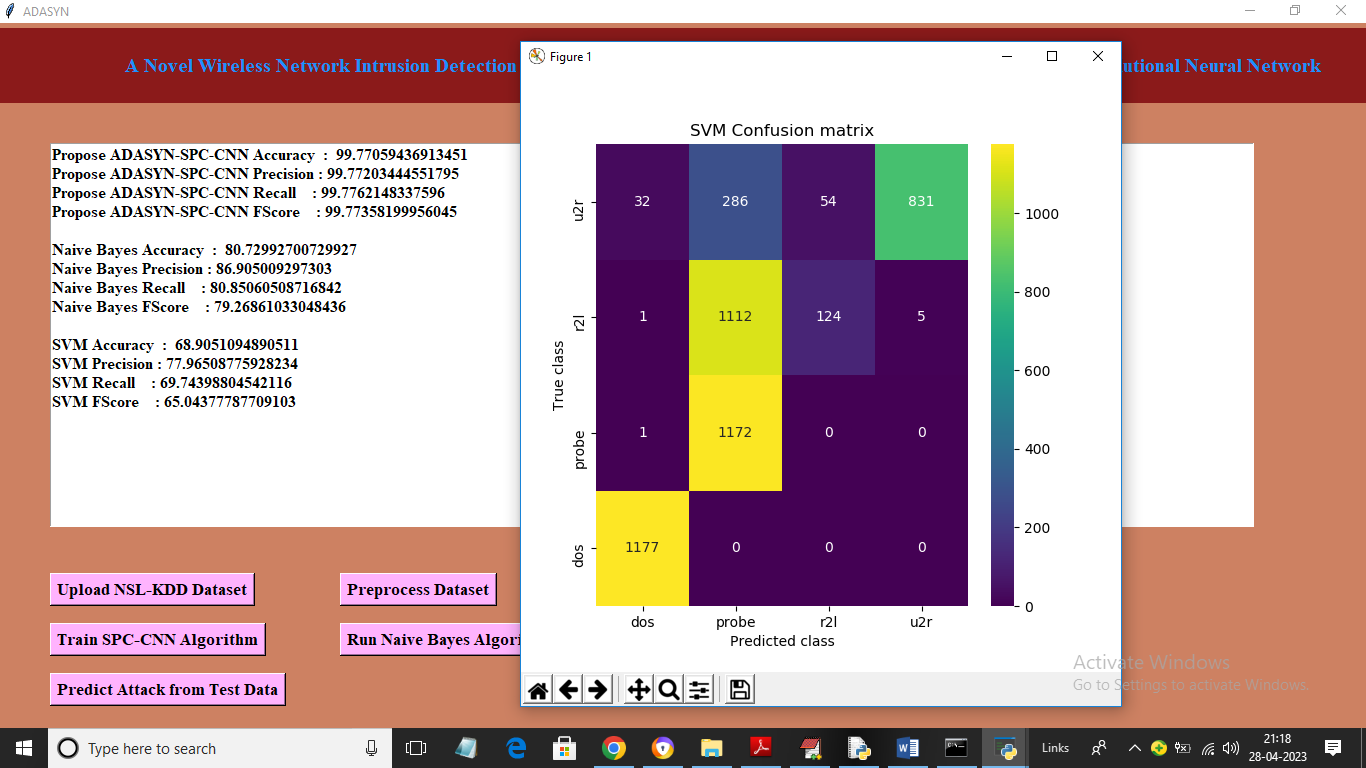
In above screen we can see application using 80% dataset for training and 20% for testing and now click on ‘Train SPC-CNN Algorithm’ button to train SPC-CNN propose algorithm and get below output



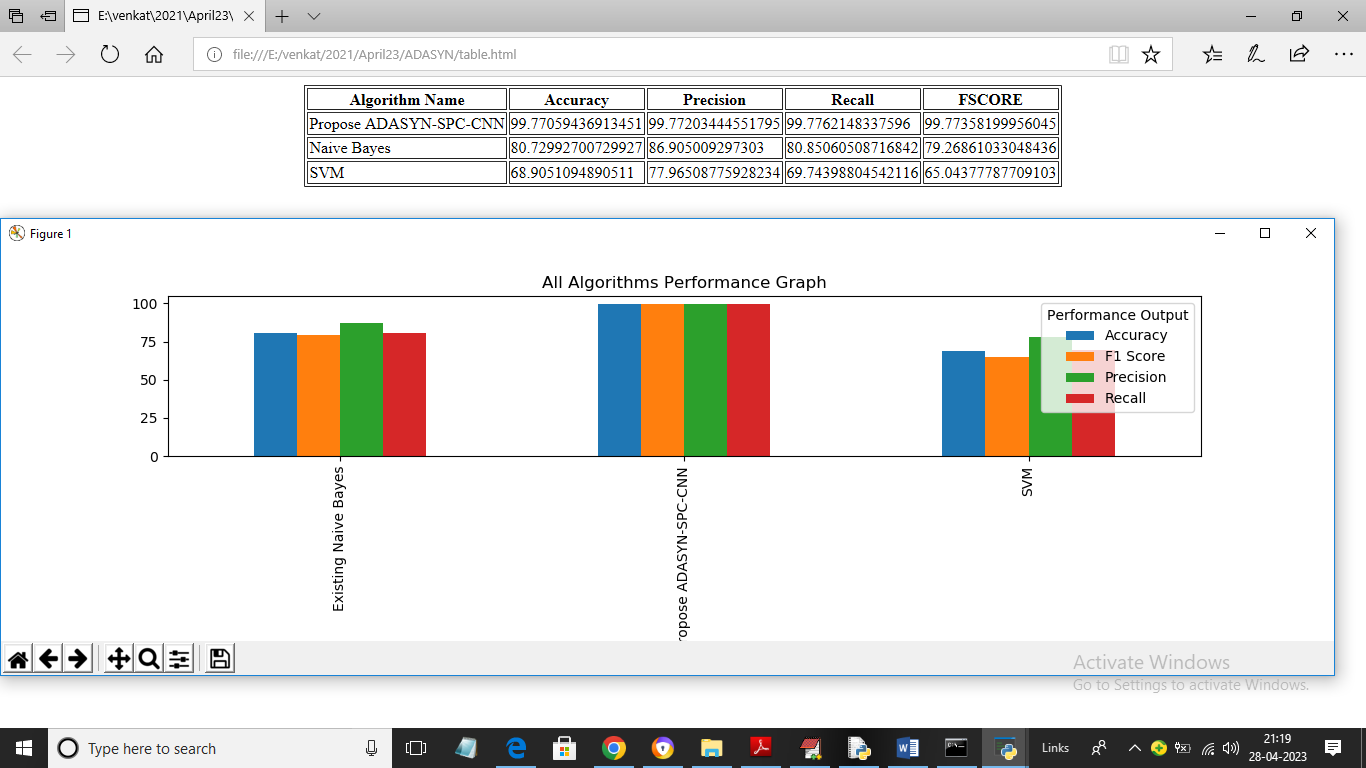
In above screen with propose SPC-CNN algorithm we got 99.77% accuracy and we can see other metrics also and in above confusion matrix graph x-axis represents Predicted Labels and y-axis represents True Labels and all yellow boxes contains correct prediction count and all blue boxes contains incorrect prediction count which are very few and now close above graph and then click on ‘Run Naïve Bayes Algorithm’ button to train Naïve Bayes and get below output



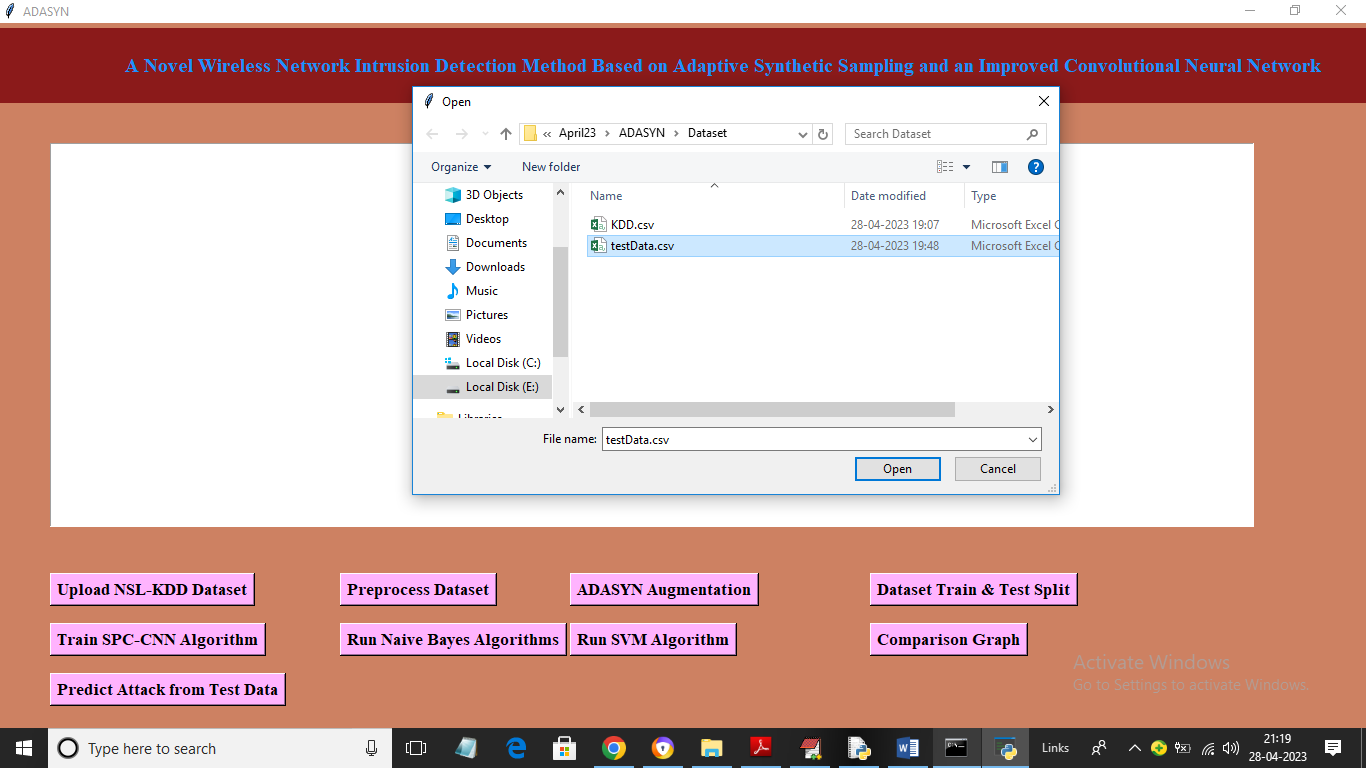
In above screen with Naïve Bayes we got 80% accuracy and now click on ‘Run SVM Algorithm’ button to train SVM and get below output



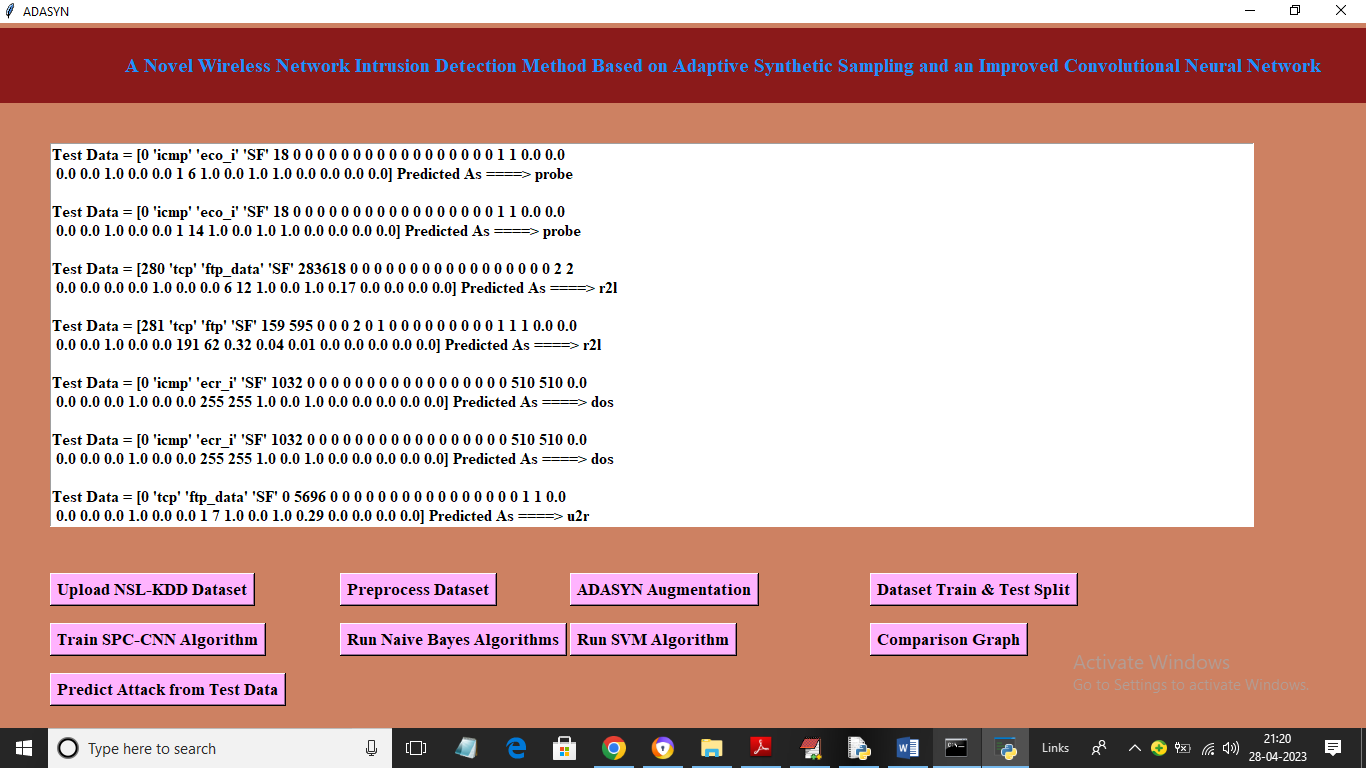
In above screen with SVM we got 68% accuracy and now click on ‘Comparison Graph’ button to get below graph.



In above graph x-axis represents algorithm names and y-axis represents accuracy and other metrics in different colour bars and in all algorithms propose SPC-CNN got high accuracy and in above screen in tabular format also we can see all algorithms performance. Now click on ‘Predict Attack from Test Data’ button to upload test data and get below output



In above screen selecting and uploading ‘testData.csv’ file and then click on ‘Open’ button to load test data and get below predicted output



In above screen in square bracket we can see test data values and after arrow symbol =🡺 we can see predicted attack names.

Similarly by following above screens you can run code and get output