ISSN 2347 - 3983
Volume 8. No. 8, August 2020
International Journal of Emerging Trends in Engineering Research
Available Online at http://www.warse.org/IJETER/static/pdf/file/ijeter59882020.pdf
https://doi.org/10.30534/ijeter/2020/59882020

Enhanced security in IoT Networks using ensemble learning methods-A Cognitive Radio Approach

V.Nallarasan[1], Dr.K.Kottilingam[2]
[1]Assistant Professor, Department of IT,SRM institute of Science and Technology,Chengalpattu,India.nallarav@srmist.edu.in
[2]Associate Professor, Department of IT,SRM institute of Science and Technology, Chengalpattu,India.
kottilik@srmist.edu.in

ABSTRACT

Ordinarily, to produce data to your Intrusion Detection System (IDS), then it's important to establish the true working environment to research all of the likelihood of strikes, that will be high priced. The Systems work to find attacks internet of the things using a cognitive radio approach in order to protect a system from attackers. The intrusion detection devices tries to develop a predictive model with the capacity of differentiating between "unwanted" relations, called attacks or strikes, and also "wanted" ordinary connections. To stop this issue in system businesses need to call whether the text is attacked or perhaps not from KDDCup99 data set employing machine learning methods. The intent is to research machine learning established processes for greater package connection moves calling by forecast contributes to most useful accuracy. In addition to compare and talk about the operation of various machine learning algorithms by the given data set with test classification file, identify the confusion matrix and also to categorizing data from the effect demonstrates that the accuracy of the used ML procedure helps in contrasting optimal accuracy with features of ML like Recall and F1 score in order to increase the efficiency of the internet of things using a cognitive radio network prediction system.

Keywords: Internet of things (IoT), Machine Learning , Cognitive radio, Intrusion Detection System (IDSs), Knowledge Discovery in Databases (KDD).

1.INTRODUCTION

1.1 Overview

Our suggested system uses machine learning (ML) supervised classification algorithms to provide a specific given network connection dataset and will extract possible out- comes or patterns which will help us in predicting whether the likely patient is affected or not, hence helping us in avoiding attacks and in making better decisions in the future. Different and multiple datasets from various sources would be collectively combined and generalized into generalised datasets[1]. After that various ML algorithms would be used and applied to extract outcomes to deduce patterns and to obtain results with the maximum of accuracy[2]. The construction of a predictive model can be done by the construction of different modules. Each of these modules will help us in creating a predictive model all together. Let us discuss each one by one. Our first module is Data Gathering or Data Collection. The collected dataset for the prediction of network attacks is Splitted into two fragments, where the first is the Training set followed by the Test set. In most cases, the ratio of 7:3 is applied to the split between Training set and Testset. The next module is Data pre - processing. Here the collected data might contain some missing values and hence leading to inconsistency. Data is preprocessed to improve the efficiency of the algorithm and to gain better results[3]. Firstly, the outliers are to be subtracted and variable conversion needs to be done. Plot diagrams under Data visualisation process can be used to identify the correlation amongst attributes. This phase consumes the maximum amount of time. For our easy analysis, the data is reduced to some minimum amount of records. The next module is Choose model where a particular model is selected to best fit the purpose and hence the procedure begins. Coming up next is Train model module where the collected datasets are trained to yield the best result possible with maximum efficiency. Next module is Test model where the trained datasets are tested to check the efficiency and the workflow of the collective model. Our next module is Tune model, it is the second last module of the process and finally Prediction module, it is used to predict the outcomes and patterns that will help us to make better decisions and to avoid attacks in the future[4].

2. PROPOSEDMETHODOLOGY

2.1 SystemArchitecture

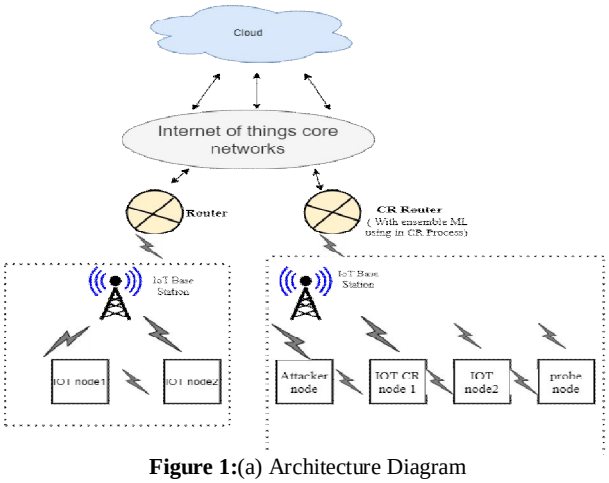


Figure 1:(a) Architecture Diagram

As we see in the figure 1.(a) architecture diagram from this internet of things core networks it can be connected and accessed by router and CR router with an enable of ensemble machine learning algorithms using in Cognitive radio process which is in order to protect the network from attackers. In IoT base station it can be consists of attacker node, IoT cognitive radio node 1, IoT node2, and probe node. By using its base station the attacker node has been accessed by the attacker and also these attacker has been trying to access by other IoT CR node1 and node2, probe node. But most probably IoT CR node 1 can be easy to detect the attacker with the help of ensemble ML algorithms and prevent that attacker but attacker node keeps on accessed by IOT node2. So we can use the IoT CR using ensemble learning methods for protecting our Network or System.

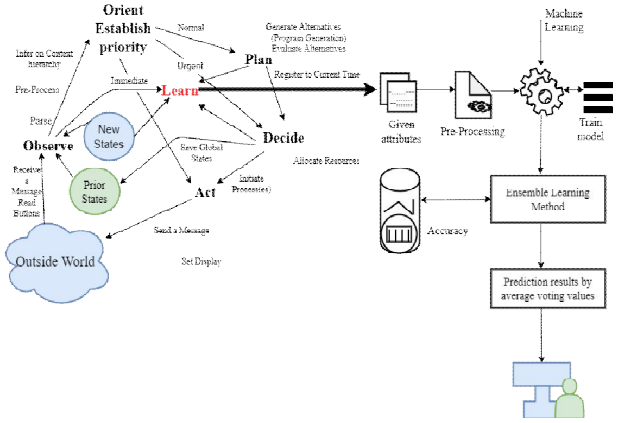


Figure 1: (b) CR Process flow Diagram

As we see process flow in figure 1. (b)its consists of the cognitive radio cycle that can observe the new states from the network or system and its transform the information to the learning environment process. It can be an analysis of that information in the given attributes and preprocessing the new states' information with the help of ensemble learning methods. If any unwanted or danger may happen means it can be predicted from the Train model and finally, CR decides to protect the network or system from the attacker.

2.2 DomainOverview

Machine learning is traditionally used for predicting the long run out of older information. Machine learning (ML) can be really a form of artificial intelligence (AI) which provides systems with the power of being able to investigate and accommodate without needing to be more specially designed. ML's focus is on growing programs which may be shifted when operate contrary to new data and also the smallest functionalities of ML, implementing of any conventional ML algorithm by means of the use of python. Method of education and forecasting involves usage of technical algorithms. It conducts on consuming education data for the algorithm then it uses this data to create forecasts to get a brand new test data [1-3].

Figure 2:ML Domain Overview

2.3 Kdd Cup99dataset

The Association for Computing Machinery (ACM) has a unique understanding of Knowledge Discovery and Data mining (KDD) which is the best recognized professional institution of data miners. The KDD organizes the yearly Data Mining competition known as KDDCup in various areas.The KDDCup99 data set originates from data gathered at MIT Lincoln Laboratory under the leadership of the Defense Advanced Research Projects Agency (DARPA) to study and judge the Intrusion Detection Systems(IDSs) of 1998 and 1999. The above data sets are called as DARPA98 and DARPA99, which has raw TCP dump data from a studied medium sized system of the US air force base. The KDDCup99 data set was given to the KDD Competition (and attached gathering) in 1999. This is a changed version of the DARPA TCP dump data, which has a set of features considered important for classification by the use of ML algorithms[2-3].

Table 1:KDD CUP Center of Attention

Year	Focused Area
KDD-CUP 1997	Direct marketing for lift curve optimization
KDD-CUP 1998	Direct marketing for profit optimization
KDD-CUP 1999	Computer network intrusion detection
KDD-CUP 2000	Online retailer website click stream analysis
KDD-CUP 2001	Molecular bioactivity and Protein locale prediction
KDD-CUP 2002	Bio Medical document and Gene role classification
KDD-CUP 2003	Network mining and usage log analysis
KDD-CUP 2004	Particle physics; plus Protein homology prediction
KDD-CUP 2005	Internet user search query categorization
KDD-CUP 2006	Pulmonary embolisms detection from image data
KDD-CUP 2007	Consumer recommendations
KDD-CUP 2008	Breast cancer
KDD-CUP 2009	Fast scoring on a large database

2.4 Ensemble Learning Techniques

Ensemble understanding helps improve machinelearning results from combining several models. This approach makes it possible for the creation of predictive operation compared to one version also it is the craft of mixing diverse group of students (specific versions) together to improvise to the stability and predictive capability of their version. In the realm of statistics and Machine Learning, Ensemble learning methods make an effort to make the performance of the predictive units improved by enhancing their own accuracy[5-8]. Ensemble learning model is one which helps in creating a system developed with finesse by the use of various learning models namely like the classifiers in order to solve problems pertaining to the network sets. Ensemble Understanding Techniques can be divided in to 3 Key kinds:-

2.4.1SequentialEnsemblelearning

Boosting is a types of ML meta form which is mostly used for reducing the bias and for keeping the variance of the data model under check which converts the poor accuracy results to perform better and yeild better results for bettering the fundamental results of the algorithms. It basically acts like an averaging system. It is the most widely used algorithm form which utilizes ensemble learning and its underlyong properties and concepts in the best manner tobring about the best fruition in terms of results. This method was mainly developed to help in the classification algorithm types but has since found its roots quite well in regression types as well

2.4.2Parallel Ensemble Learning

Bagging is a form of ML meta forms which is mainly used in order to develop resistance of a model to attacks and to in-crease their overall working abilities by increasing their accuracy levels in predictions of an attack. Moreover, it also reduces the fluctuations shown by the data in their levels commonly referred to as data variance and thereby ensures that no over-fitting of data takes place and the limits of data and realised with ease. It is an efficient model that works quite easily by using multiple sets of training versions by the use of bootstrap and thereby samples the data by substitution but it cannot be combined with other models working on the same data sets.

2.4.3 Stacking and Blending:

Stacking is a form which works by the combination of multiple models in order to touch upon the concepts of meta learners. This method isn't quite trusted like the other methods because of several underlying problems liketaking into account the majority count which may be wrong in case of faulty implication of the system under study. It's main usage is found in the combination of multiple models to produce better results.

2.5 Machine LearningConcepts

2.5.1Voting based Ensemblelearning

Voting is among the most straightforward Ensemble understanding techniques by which predictions from multiple versions are all combined. The method starts off with developing a few different models with the exact very same data set. Afterwards the Voting established Ensemble model can be used to wrap both the last versions and mixture the forecasts of both those versions. After the Voting established Ensemble version is assembled, it can be utilised to make a prediction on new info. The predictions made by the sub-models can be delegated weights. Stacked aggregation can be just a technique that is often used to find out how to take those predictions at the best way possible[9].

2.5.2 Sensitivity(Recall)

By definition, sensitivity can be termed as a measure of the proportion of real time positive cases that have been predicted affirmative or true positive. Another term that can be used to denote sensitivity is known as Recall. Therefore, conclusion made is that there can be further more proportion of real time positive cases, which would get wrongly predicted as negative and thus, could be coined as false negative as well. The information above can also be represented in a unique form of false negative rate. When added, the sum of recall and false negative rate must be 1. Let us take a real time example and understand this with the help of a model used for predicting if a person suffers from a disease or not. Recall is basically the measure of the proportion of people who suffer from a disease and got correctly predicted as the individuals who are actually suffering under the disease. In simple words, an individual who is not healthy indeed got predicted as unhealthy. Mathematically, Sensitivity can be found out by using: Sensitivity = (True Positive) / (True Positive + False Negative) The following best explain True Positive and False Negative used in the above equation: True Positive(TP) = People who were predicted as sufferers of the disease (or sick) are indeed ill from the disease (sick); In other words, the true positive denotes the actual number of people who are ill and are predicted as sick or not healthy. False Negative (FN) = People who are indeed sufferers of the disease (or sick) are predicted not to be sufferers of the disease (healthy). In other words, the false negative denotes the number of people who are ill and got wrongly predicted as healthy. Generally, we always seek the model to consist low false negatives because it can lead to result in life hazardous or business hazardous situations[9-10].

2.5.3 Specificity

Specificity is denoted as the measure of proportion of actual negatives, who got predicted as negative (or true negative). Therefore, that there will be further more proportion of real time negative, who got predicted as positive and can be coined as false positives. Another term for this proportion could be called as a false positive rate. When added, the sum of specificity and false positive rate should always be 1. Let's take a real time example and understand this with a model used for predicting if a person is a sufferer of the disease or not. Specificity is a measure of the proportion of people who do not suffer from the disease and got predicted correctly as the individuals who are not sufferers of the disease. In other words, the person who is healthy actually got predicted as healthy itself is known as specificity. Mathematically, Specificity can be found out by using: Specificity = (True Negative) / (True Negative + False Positive)

The following is the details in relation to True Negative and False Positive used in the above equation: True Negative(TN) = People who are predicted as not the sufferers of the disease (or healthy) are indeed found to be not sufferers of the disease (healthy); In other words, the true negative denotes the number of people who are healthily well and are predicted as healthy.

False Positive(FP) = People who are predicted as sufferers of the disease (or ill) and are indeed found to be not the sufferers of the disease (healthy). In other words, the false positive denotes the number of people who are healthily well and got predicted as ill[9-12].

2.5.4 Accuracy

Accuracy can be termed as the proportion of the total number of predictions that are correct or in an overall view, how often does the model predict correct defaulters and not defaulters. Logistic regression is an algorithm that uses a linear equation with in- dependent predictors to predict a specific value. The range of the predicted value can be anywhere between negative to positive infinity. The output of the algorithm must be classified variable data. We can achieve higher accuracy by comparing best accuracy using logistic regression model.

True Positive Rate (TPR) = TP / (TP + FN)
False Positive Rate (FPR) = FP / (FP + TN)
Accuracy = (TP + TN) / (TP + TN + FP + FN)

2.5.5 Precision

Precision can be termed as the ratio of correctly predicted affirmative observations to the total number of predicted affirmative observations. Higher precision implies a low false positive rate.

Precision = TP / (TP + FP)

F1Score

F1 Score is measured as the weighted average of Precision and Sensitivity. Hence, this score undertakes both, the false positives and false negatives into consideration. In gen- eral, it is not as easy of a task to understand as accuracy, but F1 is usually much useful than accuracy, especially in the case if you have an uneven class distribution. Accuracy dose the best work if false positives and false negatives have alike cost. If the cost of false positives and false negatives are very apart from each other, it's best to look at both Precision and Recall[12-13]. General Formula: F- Measure = 2TP / (2TP + FP + FN) F1-Score Formula: F1 Score = 2* (Recall * Precision) / (Recall + Precision)

2.5.6 ConfusionMatrix

A confusion matrix can be termed as a matrix which is used to understand the productivity of a classification model (or classifier) on a specific dataset or set of data for testing of the true values that have already been looked into. It also allows the representation of the performance of an algorithm. In the field of Machine Learning and the issue of statistical classification of data, it is also known as an error matrix which is in a typical table format which allows us to understand the performance of an algorithm, generally a supervised learning individual. Creation of a Confusion Matrix is of use in calculating the values of Recall and Precision which in turn facilitates in the calculation of the F1 Score and Accuracy by the use of which we can learn the production percentage of an algorithm which can then be represented graphically to contrast the accuracy and effect of different algorithms on a specific data[13-14].

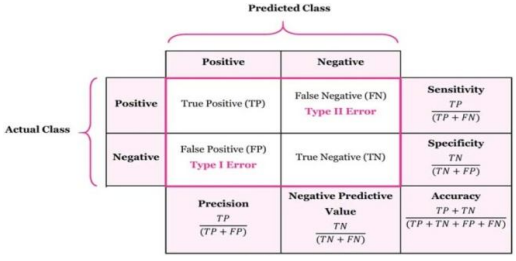


Figure 3: Confusion Matrix

3. MODULEDESCRIPTION

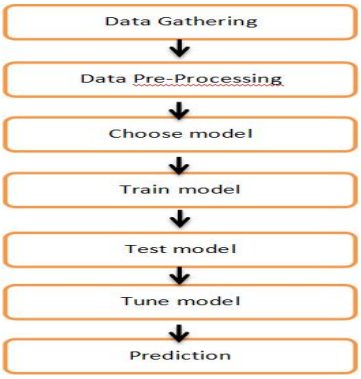


Figure 4: Machine Learning Model

The functionality can be understood in the above figure.4 Furthermore after the pre- processing of data, prediction of the various DoS, U2R, R2L and Probe attacks takes place. After this a combined model is produced that predicts the overall accuracy and the kind of attack that takes place in the overall scenario. Then we have the usage of Ensemble Learning Techniques to improve the accuracy percentage of the model and give even more prominent results. Next we have a GUI based prediction that predicts the kind of attack that takes place in a particular scenario on the basis of the Source and Destination Loads, Flag Offsets and kind of Service running[15-18].

3.1 Data Validation andPreprocessing

Data is usually set in the form of huge datasets that comprise of multiple rows and columns or structure oriented tables. In other scenarios data also exist as audio or video files. Thus when we talk about ML processes, Data Preprocessing is the step that helps the users to transform or change data into such states that are easy to render and can be easily be used to apply the various features available under ML concepts. In the data preprocessing the main objective is to prepare the data set by ensuring that the data set in question can be used in order to implement the various algorithms to get the result in form of various bar charts that help us in comparing the percentage of accuracy of the particular given algorithm against the given set of data and also acts as an indicator to understand which algorithm stands with the highest accuracy. The data is trimmed and cleaned to reduce redundancies and invalid rows which have no effect in determining the kind of attack taking place are removed.

3.2 Prediction of DoSAttack

Under this the main functionality is to use the concepts of Machine Learning and the various terminologies involved like Specificity, Sensitivity and Weighted Average to create a Confusion Matrix that is used to get the accuracy percentages of the various different algorithms and plot it in form a bar chart that helps us in comparison of the various accuracy percentages and drawing conclusions from the same. The module works only on the attacks known under the category of DoS attacks prominently.

3.3 Prediction of U2RAttack

For the module the objective is to apply ML and the various underlying concepts involved like Recall, Precision etc. to create a Confusion Matrix that will help us in getting the Accuracy percentages of the different algorithms which in turn helps us in plot the same as bar graph that helps in comparing and contrasting the performance levels of the different algorithms. The conditionality of this module is such designed that the focus is on the various significantly known attacks under the category of U2R attacks.

3.4 Prediction of R2LAttack

The module's objective is to apply ML theories and the various concepts that are the foundings of the field of ML like Sensitivity etc. to create a Confusion Matrix on the basis of which the Accuracy percentages are obtained for the multitude of algorithms which have been used in the findings which is then helpful in plotting of a bar graph that is helpful in visualizing the performance levels of the various algorithms. The design of this module helps in determining of significantly known attacks under the category of R2L attacks.

3.5 Prediction of Probe Attack

In this module the objective is to apply ML and the various underlying concepts involved like F1 Score etc. to develop a Confusion Matrix that helps us in acquiring the Accuracy percentage of the various algorithms which further helps us in converting this achieved result in the form of a bar graph that helps in drawing facts from the performance level of the different algorithms. The working is such designed that the thought process has been put on the multiple known attacks which can be found under Probe type.

3.6 Performance measurements of Overall network attacks

The main usage of this part of the code is to use the concepts of ML and its associated properties like Weighted Average to develop a Confusion Matrix which helps in the obtaining of the Accuracy percentages of the various different algorithms and form a bar graph that helps in comparison of the various accuracy percentages and drawing conclusions from the same. The major difference of this module with respect to the other modules is that it does not find the types of attacks reserved to one of the major attack types explained in the above four modules, instead it focuses on getting the results as an overall accuracy percentage and deriving results for the respective algorithms and that helps in deriving conclusions based on the accuracy percentages. Performance measurements of Overall network attacks using Ensemble Learning Technique In Ensemble Learning Technique we use Voting Classifier algorithm to derive results on the basis of the best possible algorithms that provide us with the highest accuracy percentages by using Cross Validation technique setting its value as 10. By taking into account the best possible algorithms, we try to achieve the True Positive Rates and True Negative Rates and get the final result as a percentage higher than the percentage of accuracy achieved by the overall network attack predictions.

3.7 GUI based prediction result of Network attack types by Voting Classifier

This gives us a prediction of the kind of attack on the basis of the Source File Size, Destination File Size, Protocol Type, Flag type and Services involved and gives us the type of attack that takes place in the scenario on the basis of the data set.

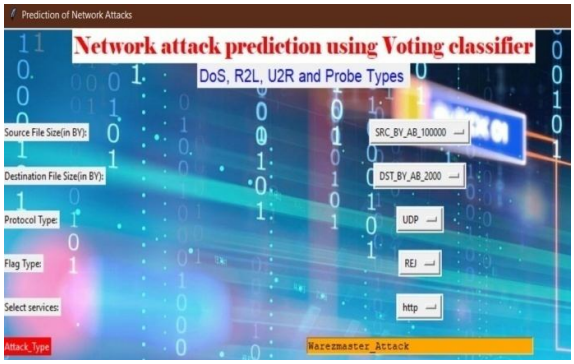


Figure 5: Results from application of EnsembleLearning

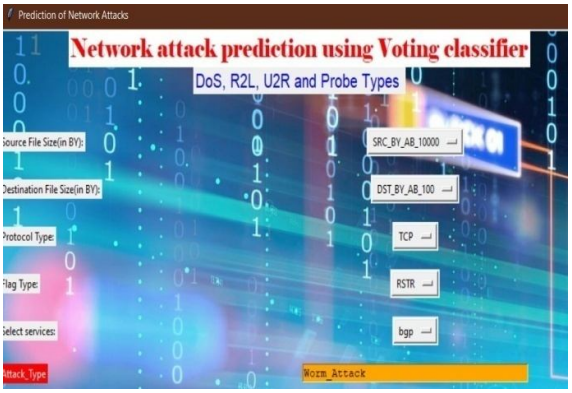


Figure 6: Results from application of EnsembleLearning

4.IMPLEMENTATION AND ITS RESULTS:

4.1 Prediction of DoSAttack

```
Classification report of Decision Tree Results:
              precision    recall  f1-score   support

           0       0.81      0.96      0.88       681
           1       0.71      0.29      0.41       219

    accuracy                           0.80       900
   macro avg       0.76      0.62      0.64       900
weighted avg       0.78      0.80      0.76       900

Confusion Matrix result of Decision Tree is:
 [[655  26]
 [156  63]]

Sensitivity :  0.9618208516886931

Specificity :  0.2876712328767123

Cross validation test results of accuracy:
[0.76723277 0.795      0.77977978]

Accuracy result of Decision Tree is: 78.06708490041824
```

Figure 7: Decision Tree Accuracy result under DoS Attack

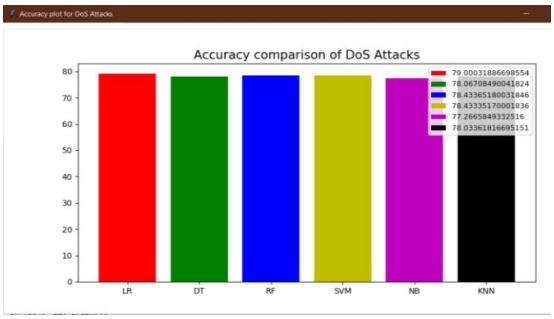


Figure 8: Graphplotted for accuracy comparison of DoS Attack

4.2 Prediction of U2R Attack

```
Classification report of Random Forest Results:

              precision    recall  f1-score   support

           0       0.79      0.97      0.87       681
           1       0.69      0.20      0.31       219

    accuracy                           0.78       900
   macro avg       0.74      0.58      0.59       900
weighted avg       0.77      0.78      0.73       900


Confusion Matrix result of Random Forest is:
 [[662  19]
 [176  43]]

Sensitivity :  0.9720998531571219

Specificity :  0.1963470319634703

Cross validation test results of accuracy:
[0.77222777 0.799      0.78178178]

Accuracy result of Random Forest is: 78.43365180031846
```

Figure 9: Random Forest Accuracy result under U2R Attack

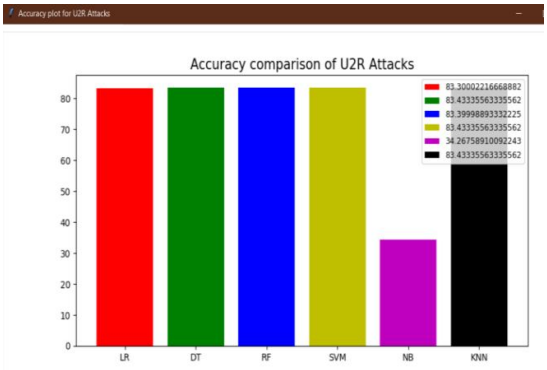


Figure 10: Graphplotted for accuracy comparison of U2R attack

4.3 Prediction of R2LAttack

```
Classification report of Support Vector Classifier Results:

              precision    recall  f1-score   support

           0       0.79      0.99      0.87       681
           1       0.78      0.16      0.27       219

    accuracy                           0.79       900
   macro avg       0.78      0.57      0.57       900
weighted avg       0.78      0.79      0.73       900


Confusion Matrix result of Support Vector Classifier is:
 [[671  10]
 [183  36]]

Sensitivity :  0.9853157121879589

Specificity :  0.1643835616438356
Cross validation test results of accuracy:
[0.77522478 0.802      0.77577578]

Accuracy result of Support Vector Classifier is: 78.43335170001836
```

Figure 11: Support Vector Classifier Accuracy result under R2L

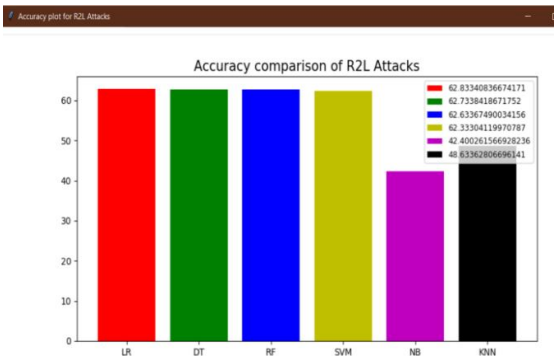


Figure 12: Graphplotted for accuracy comparison of R2L attack

4.4 Prediction of Probe Attack

```
Classification report of Naive Bayes Results:

              precision    recall  f1-score   support

           0       0.77      0.99      0.87       681
           1       0.78      0.10      0.17       219

    accuracy                           0.77       900
   macro avg       0.78      0.54      0.52       900
weighted avg       0.77      0.77      0.70       900


Confusion Matrix result of Naive Bayes is:
 [[675   6]
 [198  21]]

Sensitivity :  0.9911894273127754

Specificity :  0.0958904109589041

Cross validation test results of accuracy:
[0.77522478 0.77       0.77277277]

Accuracy result of  Naive Bayes is: 77.2665849332516
```

Figure 13: Naive Bayes accuracy result under Probe Attack

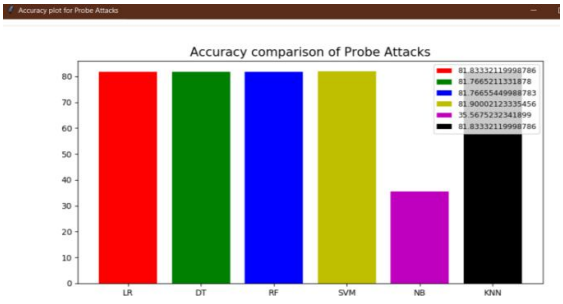


Figure 14: Graphplotted for accuracy comparison of probe attack

4.5 Prediction of Overall network attacks

```
Classification report of K-Nearest Neighbor Results:

              precision    recall  f1-score   support

           0       0.80      0.79      0.80       681
           1       0.38      0.38      0.38       219

    accuracy                           0.69       900
   macro avg       0.59      0.59      0.59       900
weighted avg       0.70      0.69      0.70       900


Confusion Matrix result of K-Nearest Neighbor is:
 [[541 140]
 [135  84]]

Sensitivity :  0.7944199706314243

Specificity :  0.3835616438356164

Cross validation test results of accuracy:
[0.76823177 0.796      0.77677678]

Accuracy result of  K-Nearest Neighbor is: 78.03361816695151
```

Figure 15: K-Nearest Neighbor Accuracy result under Overall Attack

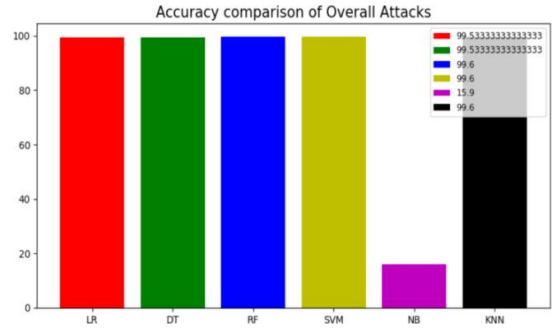


Figure 16: Graph plotted for accuracy comparison of overall attacks

4.6 Performance measurements of Overall network attacks using Ensemble Learning Technique

```
Classification report of voting classifier Results:
              precision    recall  f1-score   support

           0       0.00      0.00      0.00         4
           1       1.00      1.00      1.00       896

    accuracy                           1.00       900
   macro avg       0.50      0.50      0.50       900
weighted avg       0.99      1.00      0.99       900

Confusion Matrix result of voting classifier is:
 [[   0    4]
 [   0 896]]

Sensitivity :  0.0

Specificity :  1.0

True Positive : 896
True Negative : 0
False Positive : 4
False Negative : 0

True Positive Rate : 1.0
True Negative Rate : 0.0
False Positive Rate : 1.0
False Negative Rate : 0.0

Positive Predictive Value : 0.9955555555555555
Negative predictive value : nan
```

Figure 17: Results from application of Ensemble Learning

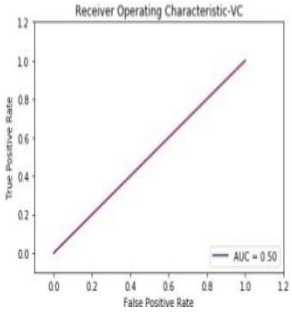
```
In [20]: from sklearn.metrics import roc_curve, auc
         def plot_roc_curve(y_test, predictR):
             false_positive_rate, true_positive_rate, thresholds = roc_curve(y_test, predictR)
             print ("False Positive rate: ",false_positive_rate)
             print ("True Positive rate: ",true_positive_rate)

             roc_auc = auc(false_positive_rate, true_positive_rate)

             plt.title('Receiver Operating Characteristic-VC')
             plt.plot(false_positive_rate, true_positive_rate, 'b',
             label='AUC = %0.2f'% roc_auc)
             plt.legend(loc='lower right')
             plt.plot([0,1],[0,1],'r--')
             plt.xlim([-0.1,1.2])
             plt.ylim([-0.1,1.2])
             plt.ylabel('True Positive Rate')
             plt.xlabel('False Positive Rate')
             plt.show()

In [21]: plot_roc_curve(y_test, predictR)

         False Positive rate:  [0. 1.]
         True Positive rate:  [0. 1.]
```




```
In [18]: def plot_confusion_matrix(cm1, title='Confusion matrix-VC', cmap=plt.cm.Blues):
             target_names=['Predict','Actual']
             plt.imshow(cm1, interpolation='nearest', cmap=cmap)
             plt.title(title)
             plt.colorbar()
             tick_marks = n.arange(len(target_names))
             plt.xticks(tick_marks, target_names, rotation=45)
             plt.yticks(tick_marks, target_names)
             plt.tight_layout()
             plt.ylabel('True label')
             plt.xlabel('Predicted label')

In [19]: cm1=confusion_matrix(y_test, predictR)
         print('Confusion matrix-VC:')
         print(cm1)
         plot_confusion_matrix(cm1)

         Confusion matrix-VC:
         [[   0    4]
          [   0 896]]
```

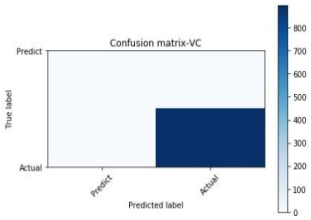


Figure 18: Results from application of Ensemble Learning

5. CONCLUSION

The process starts with cleaning and processing of data and then moves onto cleaning values which are missing for investigation and eventually developing a model and evaluation stages are carried out. The best accuracy of the public test set is the highest accuracy score that is found out by comparing each algorithm with the types of all network attacks as well as for future predictions these results have a huge role because we can conclude the type of attack if the conditions of attack is found similar to a previously found out data and can conclude that the following data in question has the same attack type.

This helps us in concluding some major insights about diagnosis of the network at- tack of each new live connection. It also presents a solution for the development of a prediction model with the help of AI to improve results beyond the normal human accuracy and also provides testers and detection systems the scope of early detection. Theinferencesthataredrawnfromthismodelarethatareainvestigationanduseof ML algorithmsarehelpfulinthedevelopmentofpredictionmodelsthatcanbeofusetothe internet of things using cognitive approach sectors in reducing the lengthy processes involved in the findings of an attack and removal of any and all human error.

6. FUTURE ENHANCEMENT

Major enhancement is the inclusion of Artificial Intelligence with the various components of the network attacks and creating a strong system which is less prone to attacks. This can be done by putting up several layer of firewalls and encryptions with the said medium to ensure that data is secured and not prone to attacks.

REFERENCES

1. Grynov Rostyslav,Vitalii Martovytskyi,**"A Method for Identifying and Countering HID Attacks - Virus Detection in BMP Images",***International Journal of Emerging Trends in Engineering*

*Research,Volume*8,No.7,July 2020.

2. W. Fan, F. Geerts, J. Li, and M. Xiong, **"Discovering conditional functional de- pendencies,"** *IEEE TKDE,* vol. 23, no. 5, pp. 683–698, 2011.

3. W. Fan and F. Geerts**,"Foundations of Data Quality Management,ser. Synthesis",** Lectures on *Data management. Morgan and Claypool Publishers,* 2012.

4. Singh WN, Marchang N. **A review on spectrum allocation in cognitive radionetwork**. *Int J Commun Networks DistribSyst* 2019;23(1):

5. Khan AA, Rehmani MH, Rached**A.Cognitive-radio-based internet of things: applications, architectures, spectrum related functionalities, and future research directions.** *IEEE WirelCommun* 2017;24(3):17–25. https://doi.org/10.1109/MWC.2017.1600404

6. S. TABBANE, **''IoT systems overview Present the different IoT systems andtheir classifications,"** no. October, 2019.

7. X. Liu et al., **''Overview of Spintronic Sensors with Internet of Things for SmartLiving,"***IEEE Trans. Magn.*, vol. 55, no. 11, 2019.

8. Wu Q et al. **Cognitive internet of things: a new paradigm beyond connection**.*IEEE Internet Things* J. 2014;1(2):129–43.

9. K. Katzis and H. Ahmadi, **''Challenges Implementing Internet of Things (IoT)Using Cognitive Radio Capabilities in 5G Mobile Networks,"** vol. 8, pp. 55–76,2016.

10. Kim S. **Inspection game based cooperative spectrum sensing and sharingscheme for cognitive radio IoT system.** *Comput. Commun*. 2017;105:116–23.

11. Han R, Gao Y, Wu C, Lu D. **An effective multi-objective optimization algorithmfor spectrum allocations in the cognitive-radio-based internet of things.** *IEEEAccess* 2018;6(January):12858–67.

12. H. B. Salameh, S. Almasri, E. Benkhelifa, and J. Lloret, **''Spectrum Assignment in Hardware-constrained Cognitive Radio IoT Networks under Varying Channelquality Conditions,"***IEEE Access*, pp. 1–1, 2019.

13. TianhengXu, Ting Zhou, JinfengTian, Jian Sang, and HonglinHu ,**''Intelligent Spectrum Sensing: When Reinforcement Learning Meets Automatic Repeat Sensing in 5G Communications",** *IEEE Wireless Communications* , February 2020.

14.P. Vijayakumar, V. Nallarasan and A. Joshua Jafferson,**Fuzzy Logic based Reliable Spectrum Sensing-A SDR Implementation**, *Journal of Advanced Research in Dynamical and Control Systems,*oct 2019,,Volume 11 | Issue 11,Pages: 137-142.

15.P. Vijayakumar, S. Malarvizhi**, "Self-Diagnosis of Cognitive Relay on the Joint Impact of Hardware Impairment and Channel Estimation Error",** *Int. J. of Systems, Control and Communications,* 2017, Vol.8, No.4, pp.335 – 347.

16.KottilingamKottursamy, Gunasekaran Raja, JayashreePadmanabhan, Vaishnavi Srinivasan, **"An improved database synchronization mechanism for mobile data using software-defined networking control"** ,*Computers & Electrical Engineering*,2017,vol.57.,pp.99-103.

17.Gunasekaran Raja, Kottilingam Kottursamy, SajjadHussain Chaudhary, Ali Hassan, Mohammed Alqarni, **"SDN assisted middlebox synchronization mechanism for next generation mobile data management system"**,*IEEE*,2017,pp.1-7

18. D. Tarek, A. Benslimane, M. Darwish et al **"Survey on spectrum sharing/allocation for cognitive radio networks Internet of Things,"***Egyptian Informatics Journa*l, https://doi.org/10.1016/j.eij.2020.02.003.

19. Anchal, Pooja Mittal**,"IoT Based Intelligent Modeling of Smart Home Parking Environment",** *International Journal of Emerging Trends in Engineering* Research,Volume 8,No.7,July 2020.