# Predictive Modeling for Attack Classification using Optimized Naïve Bayes using Weka

**Amrin Mansoori, Ankita Hundet, Babita Pathik, Shiv Kumar**

***Abstract:** The information security research that has been the subject of much attention in recent years is that intrusion detection systems. Intrusion-detection systems (IDS) intend at detecting attacks against computer systems and networks or, in general, against information systems. In fact, it is difficult to provide efficient IDS and to maintain them in such a secure state during their lifetime and utilization. Intrusion–detection systems have the task of detection of any insecure states. Machine learning in data mining field plays an essential role in the Network Intrusion Detection research area. Although there are several technological advancements in field of IDS still there are challenges. IDS are intended at detecting attacks against computer systems and networks or, in general, against information systems. The problem of developing an ability to detect novel attacks or unknown attacks based on audit data in IDS is still on verge. Also, the classification accuracy is one such inadequacy, the Weka tool is tested for the few machine learning techniques in this work. This paper presents comparison of K-NN, Decision tree, Naïve Bayes based classifiers using Weka tool, for IDS. This paper will provide an insight for the future research. The KDD CUP'99 data set is employed for experiment, result analysis and evaluation. The methods tested based on Detection rate and False Alarm rate.*

***Keywords:** Classification, Data Mining, Intrusion Detection System (IDS), Machine Learning techniques, Weka, KDD CUP'99 dataset.*

## I. INTRODUCTION

Intrusion-detection systems (IDS) intend at detecting attacks against computer systems and networks or, in general, against information systems. In fact, it is difficult to provide efficient IDS and to maintain them in such a secure state during their lifetime and utilization. Intrusion–detection systems have the task of detection of any insecure states. Figure 1 shows the intrusion detection system. They detect unusual attempts, active misuse of rights and attempts to exploit security vulnerabilities. It can processes information coming from the system to be protected. It can also investigate the requests coming from external sources. It uses three kinds of information:

1.) Long-term information (i.e. knowledge base of attacks) to detect intrusions.
2.) Configuration information (i.e. current state of the system).
3). Audit information (i.e. the events happening on the system).

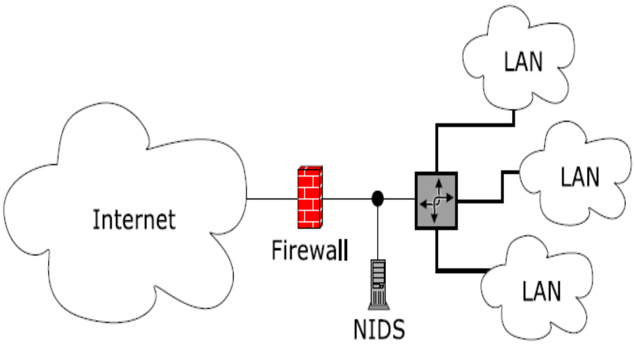


**Figure 1: Network Intrusion Detection System**

In conventional models of network and Internet security, a security boundary is deployed to build IDS to control the use intrusion in information systems. In such a model, the organization can control the storage and processing of the data depending on the organizational policies. Efficient intrusion detection is needed as a security cover against these malicious or suspicious and abnormal activities. Out of these major concern for organizations is Security. Although there are several technological advancements in field of IDS still there are challenges. IDS are intended at detecting attacks against computer systems and networks or, in general, against information systems. In fact, it is difficult to provide efficient IDS and to maintain them in such a secure state during their lifetime and utilization. IDS can processes information coming from the system to be protected. It can also investigate the requests coming from external sources. Intrusion–detection systems have the task of detection including:

- Detection of insecure states.
- Detection of unusual attempts.

The problem of developing an ability to detect novel attacks or unknown attacks based on audit data in IDS is still on verge. Misuse based IDS detect abnormal behaviour by analyzing the given traffic and go with several rules based on Analysis and comparison and thus detect only known attacks. On the other hand Anomaly IDS trying to detect anomalies when any difference occur from the normal system. Another problem of detecting unknown patterns in large amount of audit data is very difficult.

**Manuscript published on 30 November 2017.**
***Correspondence Author(s)**

**Amrin Mansoori,** M.Tech Scholar, Department of Computer Science & Engineering, Lakshmi Narain College of Technology & Excellence, Bhopal (M.P), India. E-mail: amrinmansoori7@gmail.com

**Ankita Hundet,** Assistant Professor, Department of Computer Science & Engineering, Lakshmi Narain College of Technology & Excellence, Bhopal (M.P), India.

**Babita Pathik,** Assistant Professor, Department of Computer Science & Engineering, Lakshmi Narain College of Technology & Excellence, Bhopal (M.P) India.

**Dr. Shiv Kumar,** Professor & Head, Department of Computer Science & Engineering, Lakshmi Narain College of Technology & Excellence, Bhopal (M.P), India.

So various works in IDS suggest use of Machine-learning as a solution as it seems well-suited to overcome this problem and can therefore be used to discover unknown patterns. The adaptive and dynamic nature of machine-learning makes it a suitable solution for this situation.

One of the complementary trends in intrusion detection is to use the knowledge accumulated about attacks and look for evidence of the exploitation of these attacks, and is often referred to as misuse detection.

To evaluate the efficiency of an intrusion-detection system three major parameters are: Accuracy, Performance and Completeness. In recent years, Machine Learning (ML) techniques established much interest to overcome the limitation of traditional IDSs by increasing accuracy and detection rates.

Classification and Machine learning techniques will be the main focus of this research. Classification is assigning a class label to a set of unclassified objects described by a fixed set of attributes or features. In literature, numbers of Intrusion detection systems are developed based on many different machine learning techniques such as Neural Networks, Support Vector Machine (SVM) and genetic algorithms etc. These techniques are developed as classifiers, which are used to classify or distinguish whether the incoming Internet access is the normal access or an attack. In our work, we did performance evaluation of three different classifiers based on 3 different categories of ML algorithms. The categories are Decision Tree, Naïve Bayes and Neural Networks.

## II. REVIEW OF LITERATURE

The authors [1] proposed a new approach based on ML techniques including artificial neural networks and SVM. Authors applied this approach to the KDD CUP'99 data set. the proposed approach provided high performance, especially for U2R attacks and R2L type.

According to the authors [2], neural networks, SVM and decision trees are the admired schemes for IDS. In this paper [3] three techniques are compared by applying ML techniques on KDD CUP'99 data set. The techniques are supposed to be good for identifying the anomalies detection, but the performance may differ in terms of different algorithms.

Chi Cheng et al. [3] proposed Extreme Learning Machines methods to classify binary and multi-class network traffic for intrusion detection. The performance of ELM in both binary- and multi-class data are examined, and compared to SVM based classifiers. Simulation results on KDD CUP'99 data set demonstrated that the proposed method can detect intrusions for large datasets also with optimal training and testing times.

The work [4] presents a neural-network-based active learning procedure for computer network intrusion detection. A comparison of the with a C4.5 decision tree indicated that the actively learned model had better generalization accuracy.

The authors [5] assessed the performance of a ML algorithm called Decision Tree and compared with two other ML algorithms namely SVM and Neural Network. The algorithms were analyzed on basis of detection accuracy and rate, false alarm rate of all four attack types. From the experiments conducted, authors found that the Decision tree algorithm outperformed the other two algorithms. Jingbo Yuan et al. [6] first introduces the basic structure of the IDS, then analyzed the Intrusion Detection Techniques Based on ML methods, including the Bayesian based method, the Neural Network and SVM based method.

The authors in [7], aim to use data mining techniques including classification tree and SVM for intrusion detection. As their results indicate, C4.5 algorithm is better than SVM in detecting network intrusions and false alarm rate in KDD CUP 99 dataset.

Based on the survey, in this paper we evaluate the performance of a comprehensive set of classifier algorithms using KDD CUP’99 dataset. Based on evaluation results, we intend to compare the efficiency of three classifiers based on Neural Networks, SVM and Decision Tree algorithms against KDD-cup dataset, in this research.

## III. EXPERIMENTAL FRAMEWORK (DATASET)

### 3.1. KDD CUP 1999 Dataset

KDD CUP 1999 Data (KDD99) is the dataset used in the evaluate machine learning technique. The full KDD99 dataset Contain 4,898,431 records and each record contain 41 features [8]. Due to the computing power, we do not use the full dataset of KDD99 in the experiment but a 10% portion use of it. This 10% KDD99 dataset contains 494,069 records (each with 41 features) and 4 categories of attacks. The details of attack categories and specific types are shown in Table1. The four attack types are [2, 5, and 7]:

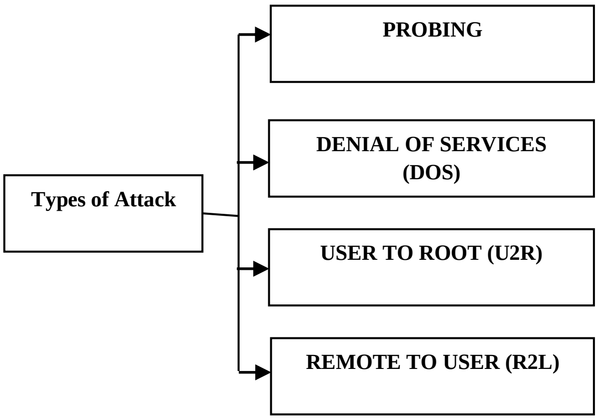


**Figure 2: Types of Attack**

**Probing:** Scan networks to gather deeper information

**Denial of service (DoS):** This attack can easily exhaust the computing and communication resources of its victim within a short period of time.

**User to Root (U2R) Attacks (U2R):** Illegal access to gain super user privileges

**Remote to User (R2L):** Illegal access from a remote machine

Every attack categories contain some specific attack types [2].

*Published By:*
*Blue Eyes Intelligence Engineering*
*and Sciences Publication (BEIESP)*

For example, DoS has 6 specific attack types (e.g. back, land, neptune), R2L has 8 specific attack types (e.g. ftp write, guess password, imap). There are totally 22 specific attack types within the 10% KDD99 dataset, while the full KDD99 dataset has 39 specific attack types.

Although the number of specific attack types is different between 10% KDD99 dataset and full KDD99 dataset, we believe that there are no negative effects on our evaluation purpose.

### 3.2 WEKA Tool Description

WEKA is a data mining system developed by the University of Waikato in New Zealand that implements data mining algorithms. WEKA is a state-of-the-art facility for developing machine learning (ML) techniques and their application to real-world data mining problems. It is a collection of machine learning algorithms for data mining tasks. The algorithms are applied directly to a dataset. WEKA implements algorithms for data preprocessing, classification, regression, clustering, association rules; it also includes a visualization tools. The new machine learning schemes can also be developed with this package. WEKA is open source software issued under the GNU General Public License.

WEKA Explorer preprocessing, classification, clustering, association, attribute selection, and visualization tools.

### 3.3 Experimental Framework

You can launch Weka from C:\Program Files directory, from your desktop selecting icon, or from the Windows task bar 'Start' Æ 'Programs' Æ 'Weka 3-4'. When 'WEKA GUI Chooser' window appears on the screen, you can select one of the four options at the bottom of the window:

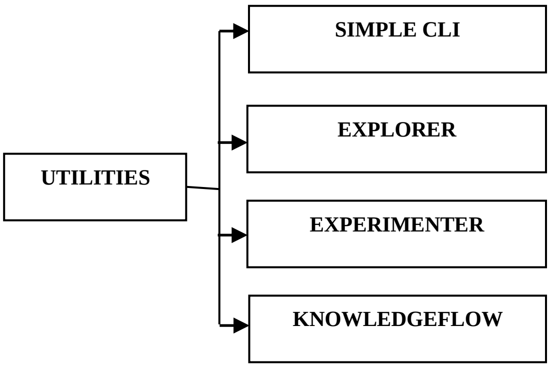


**Figure 3: Architecture of Weka**

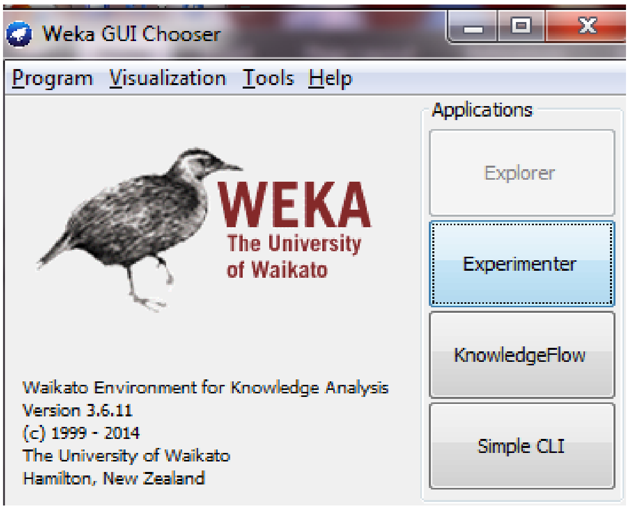
**Simple CLI:** provides a simple command-line interface and allows direct execution of Weka commands.

**Explorer:** is an environment for exploring data.

**Experimenter:** is an environment for performing experiments and conducting statistical tests between learning schemes.

**Knowledge Flow:** is a Java-Beans-based interface for setting up and running machine learning experiments.

The GUI Chooser consists of four buttons one for each of the four major. On starting Weka application the interface comes with four menus and buttons. We choose 'Explorer' which provides an environment for performing experiments and conducting statistical tests between learning schemes.

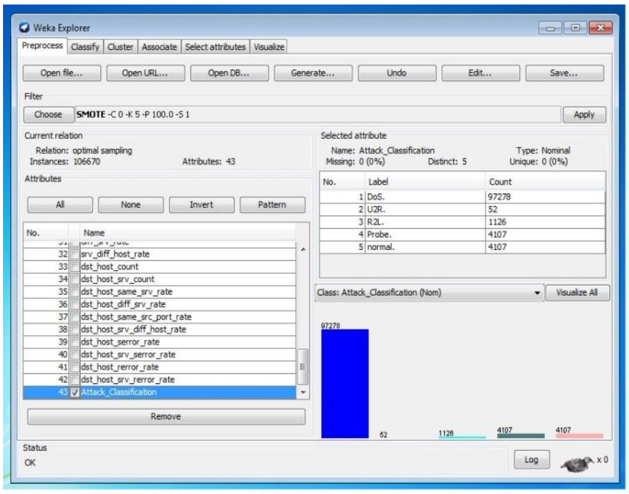


Waikato Environment for Knowledge Analysis
Version 3.6.11
(c) 1999 - 2014
The University of Waikato
Hamilton, New Zealand

**Figure 4: GUI Screen of Weka**

Firstly, we build the experiment environment in Weka for evaluation, with major steps: environment setup, data preprocessing, choosing the classifier. Figure 3 and 4 shows the Experimental framework for this work. Secondly, we select a comprehensive set of most popular classifier algorithms, three distinct widely used classifier algorithms were selected so that they represent a wide variety of fields: Bayesian, decision trees, and lazy functions.

Finally, we come up with the performance comparison between the selected classifiers in next section. To verify the efficiency of said classifiers for the field of intrusion detection, we will use the KDD cup 99 dataset. KDD training dataset consists of approximately 4,900,000 single connection vectors each of which contains 41 features and is labeled as either normal or an attack, with exactly one specific attack type. The dataset is split into training and testing sets of 30 and 70 percent respectively. The training set split is used to build the model using ML and the model built is tested on 70% of dataset (test) .We are using 10% of that dataset which is having 494021 instances. For this purpose we used Weka 6.0 tool, a brief description of which is given in section 3.2.



Filter
SMOTE -C 0 -K 5 -P 100.0 -S 1

Current relation
Relation: optimal sampling
Instances: 106670 Attributes: 43

Attributes

| No. | | Name |
|---|---|---|
| 32 | □ | srv_diff_host_rate |
| 33 | □ | dst_host_count |
| 34 | □ | dst_host_srv_count |
| 35 | □ | dst_host_same_srv_rate |
| 36 | □ | dst_host_diff_srv_rate |
| 37 | □ | dst_host_same_src_port_rate |
| 38 | □ | dst_host_srv_diff_host_rate |
| 39 | □ | dst_host_serror_rate |
| 40 | □ | dst_host_srv_serror_rate |
| 41 | □ | dst_host_rerror_rate |
| 42 | □ | dst_host_srv_rerror_rate |
| 43 | ☑ | Attack_Classification |

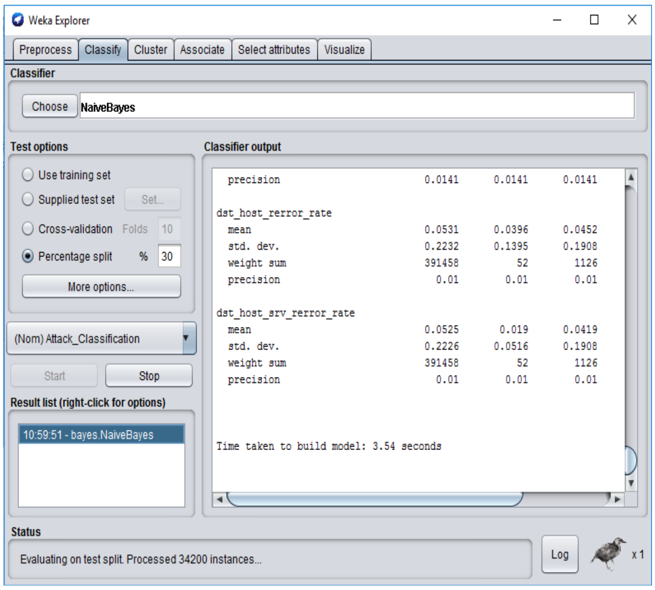
Selected attribute
Name: Attack_Classification Type: Nominal
Missing: 0 (0%) Distinct: 5 Unique: 0 (0%)

| No. | Label | Count |
|---|---|---|
| 1 | DoS. | 97278 |
| 2 | U2R. | 52 |
| 3 | R2L. | 1126 |
| 4 | Probe. | 4107 |
| 5 | normal. | 4107 |

Class: Attack_Classification (Nom)

Status
OK

**Figure 5: Snapshot of Our Experiment**



NaiveBayes

Folds 10

% 30

(Nom) Attack_Classification

Classifier output

```
  precision                     0.0141    0.0141    0.0141

dst_host_rerror_rate
  mean                          0.0531    0.0396    0.0452
  std. dev.                     0.2232    0.1395    0.1908
  weight sum                    391458        52      1126
  precision                       0.01      0.01      0.01

dst_host_srv_rerror_rate
  mean                          0.0525     0.019    0.0419
  std. dev.                     0.2226    0.0516    0.1908
  weight sum                    391458        52      1126
  precision                       0.01      0.01      0.01



Time taken to build model: 3.54 seconds
```

Result list (right-click for options)

10:59:51 - bayes.NaiveBayes

Status

Evaluating on test split. Processed 34200 instances...

**Figure 6: Snapshot of Our Experiment**

Figure 3 and 4 showing snapshot of our experiment. The suitable sampling, normalization and filtering techniques are used in our work.

## IV. METHODOLOGY USED IN IMPLEMENTATION

The Model is presented in Figure 7, which actually employs a Optimized Naïve Bayes ML algorithm for faster and accurate convergence of model. The model also employs filters for faster evaluation and lesser overall time. The pre-processing methods and application of filters affect a lot in final evaluation results of classifiers (ML based models). The feature extraction methods, conversion of nominal to binary and cleaning are few of those filters.

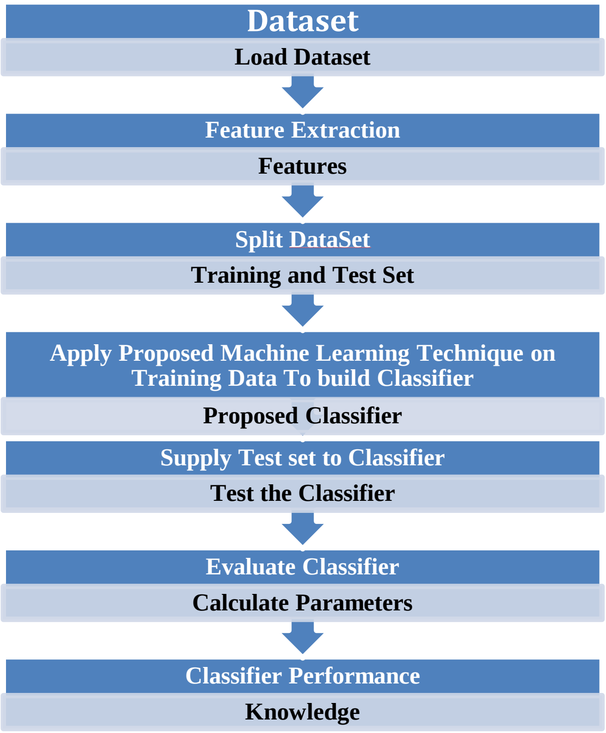


**Figure 7: Proposed Data Mining Framework for Classification**

The following figure 8 explains the intrusion / attack prevention framework employed by proposed model. Generally an algorithm or multiple algorithms detects/classifies the attack, which has more convergence time. So, for faster convergence application of Attack Prevention Framework is suggested. The major steps involve application of filters to incoming traffic followed by application of Proposed Naïve Bayes algorithm. If prediction model detects the attack as per algorithm followed then traffic is denied else allowed.

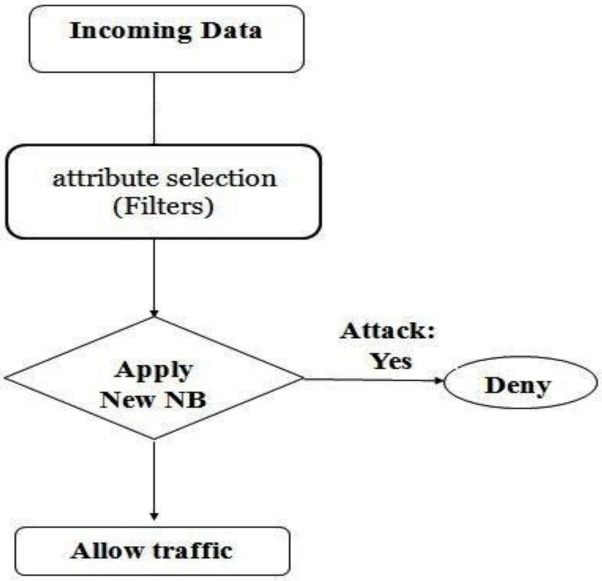


**Figure 8: Attack Prevention Framework**

## V. RESULT AND PERFORMANCE ANALYSIS

Authors follow following steps during execution of their implementation:

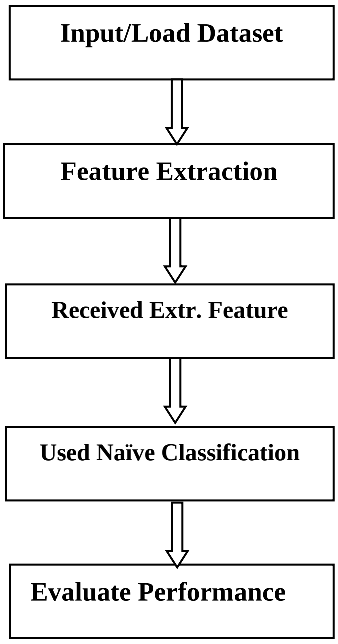
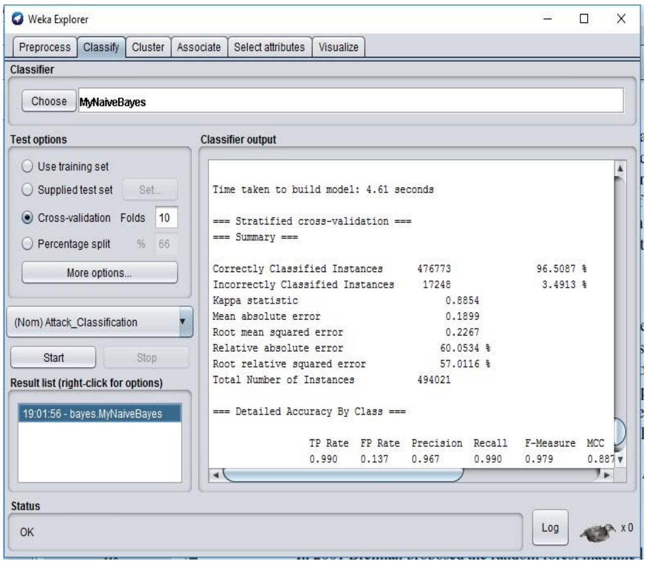


**Figure 9: Steps Follow During Execution**

*Published By:*
*Blue Eyes Intelligence Engineering*
*and Sciences Publication (BEIESP)*



Classifier

Choose: MyNaiveBayes

Test options

○ Use training set
○ Supplied test set — Set...
● Cross-validation — Folds: 10
○ Percentage split — %: 66

More options...

(Nom) Attack_Classification

Result list (right-click for options)

19:01:56 - bayes.MyNaiveBayes

Classifier output

```
Time taken to build model: 4.61 seconds

=== Stratified cross-validation ===
=== Summary ===

Correctly Classified Instances      476773              96.5087 %
Incorrectly Classified Instances     17248               3.4913 %
Kappa statistic                          0.8854
Mean absolute error                      0.1899
Root mean squared error                  0.2267
Relative absolute error                 60.0534 %
Root relative squared error             57.0116 %
Total Number of Instances           494021

=== Detailed Accuracy By Class ===

                 TP Rate  FP Rate  Precision  Recall   F-Measure  MCC
                 0.990    0.137    0.967      0.990    0.979      0.887
```

Status

OK

**Figure 10: Results of Simulation**

A representative and frequent task in the area of machine learning is comparing two or more learning procedures with one another. This can be done to study the improvements that can be obtained by new procedures, and also simply be used to select a suitable technique for IDSs. In this section we will show how this can be done with Weka. The detection of attacks can be measured by following metrics [5, 7]:

### 5.1. Performance Metrics

**True Positive:** When, the number of found instances for attacks is actually attacks.

**False Positive:** When, the number of found instances for attacks is normal.

**True Negative:** When, the number of found instances is normal data and it is actually normal.

**False Negative:** When, the number of found instances is detected as normal data but it is actually attack.

The accuracy of IDS is measured generally on basis of following parameters:

**Detection Rate:** Detection rate refers to the percentage of detected Attack among all attack data, and is defined as follows:

$$\text{Detection rate} = \frac{TP}{TP + TN} * 100$$

With this formula detection rate for different types of Attacks can be calculated.

**False Alarm rate:** It refers to the percentage of normal data which is wrongly recognized as attack. The formula represented below as:

$$\text{False Alarm rate} = \frac{FP}{FP + TN} * 100$$

**Table 2: Performance comparison of three Classifiers**

| Classifier Category | Classifier Algorithm | | DoS | Probe | U2R | R2L |
|---|---|---|---|---|---|---|
| **Bayes** | Naïve Bayes | TP | 93.9 | 92.0 | 86.5 | 36 |
| | | FP | 0.12 | 0.62 | 0.017 | 0.001 |
| **Trees** | J48 | TP | 100 | 97.8 | 40.5 | 95.0 |
| | | FP | .0001 | 0.0 | 0.0 | 0.0 |
| **Lazy** | k-NN | TP | 96.7 | 73.51 | 23 | 8.15 |
| | | FP | 0.71 | 0.2 | 0.1 | 0.52 |

### 5.2 Comparison with Different Machine Learning Techniques

An overview of how specific values of these algorithms were identified as well as their detection performance will be given. In results it is shown that no single algorithm could detect all attack classes with a high detection rate and a low false alarm rate. It reinforce our belief that different algorithms should be used to deal with dissimilar types of network attacks. Results show that certain algorithms have superior detection performance compared to others. For DoS category, most algorithms provide very high TP rates – averagely 96%. Naïve Bayes is the only one that lags as it gives a TP at 93.9%. But for Probe attacks, Decision Tree outperforms the others with its TP at 97.8%; Decision Tree has impressive performance for this category at 92.0%. In U2R attacks, Naïve Bayes and Decision Tree are the best two classifiers with FP at approximately 0.0001.

## VI. CONCLUSION AND FUTURE WORK

Intrusion-detection systems (IDS) intend at detecting attacks against computer systems and networks or, in general, against information systems. In recent years ML techniques proved useful and attracted researchers in the NIDS research area. Seeming the need of correct classification of correct attack types, the Weka tool is tested for the few machine learning techniques including: Decision tree, K-NN and Bayesian is demonstrated in this work.

The performance comparison of three machine learning techniques on KDD Cup 1999 Intrusion Data (KDD99) using Weka tool is presented. Also, as most of the research works using tools like MATLAB, WEKA etc. are available. The purpose of this work is to test and evaluate the machine learning techniques on Weka. The work also indicated that the actively learned model had better classification accuracy. The classifiers were tested based on Detection rate and Accuracy. Results also show that for a given attack category, certain algorithms shows superior detection performance compared to others.

At the same time, the factor such as ever growing amount of data for classification and constraints on response time, have made DM tasks a challenging job in the IDSs domain. So, to override the constraints on size of data to be classified and computational performance, the choices of various platforms for Intrusion detection is available. When making scientific predictions, Machine Learning (ML) has unique ability to evaluate large number of variables than a human possibly could do. In future, researchers can propose Classification Frameworks for Network Intrusion Prediction on various cloud platforms. This work will definitely provide an important reference for the future research.

## REFERENCES

1. Hua TANG, Zhuolin CAO "Machine Learning-based Intrusion Detection Algorithms" Journal of Computational Information Systems5:6(2009) 1825-1831 Available at http://www.JofCI.org
2. YU-XIN MENG "The Practice on Using Machine Learning For Network Anomaly Intrusion Detection" 2011 IEEE.

3. Chi Cheng, Wee Peng Tay and Guang-Bin Huang “Extreme Learning Machines for Intrusion Detection” - WCCI 2012 IEEE World Congress on Computational Intelligence June, 10-15, 2012 - Brisbane, Australia
4. Naeem Seliya , Taghi M. Khoshgoftaar “Active Learning with Neural Networks for Intrusion Detection” IEEE IRI 2010, August 4-6, 2010, Las Vegas, Nevada, USA 978-1-4244-8099-9/10/$26.00 ©2010 IEEE
5. Kamarularifin Abd Jalill, Mohamad Noorman Masrek “Comparison of Machine Learning Algorithms Performance in Detecting Network Intrusion” 201O International Conference on Networking and Information Technology 978-1-4244-7578-0/$26.00 © 2010 IEEE
6. Jingbo Yuan , Haixiao Li, Shunli Ding , Limin Cao “Intrusion Detection Model based on Improved Support Vector Machine” Third International Symposium on Intelligent Information Technology and Security Informatics 978-0-7695-4020-7/10 $26.00 © 2010 IEEE
7. Mohammadreza Ektefa, Sara Memar, Fatimah Sidi, Lilly Suriani Affendey “Intrusion Detection Using Data Mining Techniques” IEEE 2010.
8. Mahbod Tavallaee, Ebrahim Bagheri, Wei Lu, and Ali A. Ghorbani “A Detailed Analysis of the KDD CUP 99 Data Set” Proceedings of the 2009 IEEE Symposium on Computational Intelligence in Security and Defense Applications (CISDA 2009)
9. Megha Aggarwal, Amrita “Performance Analysis Of Different Feature Selection Methods In Intrusion Detection” international journal of scientific & technology research volume 2, issue 6, june 2013
10. Huy Anh Nguyen and Deokjai Choi “Application of Data Mining to Network Intrusion Detection: Classifier Selection Model” Weka, University of Waikato, Hamilton, New Zealand