Journal of Harbin Engineering University
ISSN: 1006-7043
Vol 45 No. 5
May 2024

Protecting the Internet: How Smart Computers Detect Online Threats using Intrusion Detection System (IDS)

[1] Badisa Naveen, [1] Jayanth Krishna Grandhi, [1] Kallam Lasya, [1] Eda Mokshita Reddy,
[2] Nulaka Srinivasu, [3] Suneetha Bulla
[1] Department of Computer Science, Koneru Lakshmaiah Education Foundation Guntur, India,
[2] Professor, Koneru Lakshmaiah Education Foundation,Vaddeswaram,A.P,India,
[3] Associate Professor, Koneru Lakshmaiah Education Foundation,Vaddeswaram,A.P,India.

Abstract

Introduction— In today's internet-driven world, there's a growing threat of cyberattacks. To keep the internet safe, we need a powerful Intrusion Detection System (IDS). This system helps spot and stop online attacks. In this paper, we suggest a new way to do this using smart computer programs called Machine Learning Algorithms. We also use a special dataset called KDD-CUP-99 to see how well these programs can find internet attacks.Our tests show that boosting Algorithms do a great job compared to other computer programs. This research can help make the internet safer by improving our ability to detect and defend against online threats.

Objectives: Evaluate how well various machine learning algorithms can detect several kinds of cyber threats with the KDD-CUP-99 dataset or other similar data sets. This task will compare how boosting algorithms perform against traditional methods.

Methods: To improve discriminatory ability of selected features, this study took a multifold approach on feature selection using correlation-based, Principal Component Analysis (PCA)-based, Information Gain ratio-based modeling, and redundancy minimization methods. We used several classifiers such as: Decision Tree; Random Forest; Gaussian Naïve Bayes; Supervised Machine Learning Model (SVM); XGBoost; and Gaussian Naïve Bayes among others for intrusion detection testing. Decision Tree classifying algorithms had more interpretable results whereas Random Forests enhanced accuracy through ensemble learning while Gaussian Naïve Bayes was computationally efficient. In the process SVM was striving to determine best class division hyperplane.

Results: A variety of machine learning algorithms were experimented on to determine which were best at keeping out intruders, and the results showed that some worked well while others didn't. Most notably, Gradient Boosting and XGBoost showed the best performance among them all.Research also showed that when compared to Support Vector Classifier (SVC),Logistic Regression,Decision Tree,Gaussian Naive Bayes (NB),Bernoulli Naive Bayes (BNB),Random Forest, and Light GBM, both SV and XG were able to detect intruders more effectively .These two methods ,particularly, demonstrate the importance of boosting in Intruder Detection System (IDS) performance enhancement by making it easier for attacks to be detected and stopped accurately.

Conclusions: Gradient Boosting and XGBoost are among the boosting algorithms used in the field of Intrusion Detection Systems. The two algorithms were found to be significantly powerful. Their power lies in their ability to effectively learn intricate styles and ensemble learning, which in turn help improve detection accuracy and make them more resistant to different kinds of cyberattacks. The most effective and consistent defense mechanisms against unwanted intrusions into networked devices are the advanced Machine Learning tools that boost algorithms like Boosting. In the future there should be more investigations on whether it would be possible for one to invest time as well as money(or other resources) on something so as not only ensure security but enhance safety too especially when it comes down for identification system such as mean square error.

Keywords: Decision Tree, Gaussian Naïve Bayes, Gradient Boosting, intrusion detection (IDS), KDD-CUP-99 Dataset, Logistic Regression, machine learning, Random Forest, XGBoost.

1. Introduction

The Internet has a number of challenges to secure data and prevent attacks on the internet. Intrusion detection is used to determine the detecting attacks. Intrusion Detection is nothing, but it is a process that inspects the data for malicious and inaccurate or anomalous activities. As we know, they are two basic levels in the Intrusion detection system. They are Host and Network based intrusion systems. Also, we have two methods in an intrusion detection system. They are Anomaly and misuse. The misuse is used to identify the signatures of attacks on the monitored resources whereas Anomaly works on the knowledge of behavior and deviations. We are all aware that Anomaly detection is very famous for finding or detecting new attacks.

We can use Machine Learning algorithms in anomaly detection. The Machine Learning algorithms are well-trained, and we can apply on directly to hidden inputs to detect network attacks. Also, Intrusion detection system (IDS) monitors the network traffic to identify the issues and alerts the system when suspicious activities are found. In this paper, we used an Intrusion detection system to network with a variety of feature selection techniques and classifiers to study the combination of feature selection techniques.

In the paper, the study reveals which feature classifier technique is used to build an accurate Intrusion Detection System.

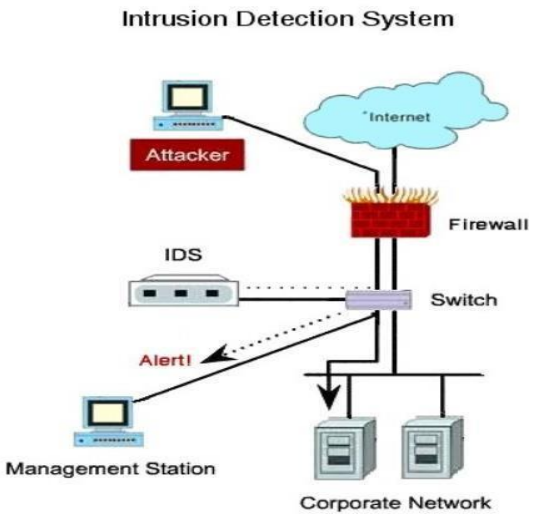

Fig. 1. Working of Intrusion detection system (Source: Internet)

2. Literature Survey

Many technological security flaws are leading to computer system intrusions which are increasing in number day by day, which are becoming economically expensive for the manufacturers to resolve [1]. To counter this problem many machine learning techniques are developed and many more are going through that development, Chih-Fong Tsai explored and understood the use of such ML algorithms in addressing the intrusion detection issues by going through related studies and analyzing 50+techniques [2]. A simple and yet effective approach is proposed by Phurvit for utilizing supervised Machine learning approach in detecting real time intrusions [3]. The most often used dataset for assessing these detections is KDDCUP'99 [4]. After studying the dataset carefully and selecting the right features, characteristics from the dataset result in greater rates of correct detection with minimal false alarms [5]. One of the most used methods for designing the IDS is artificial neural network approach. Akshadeep created an ANN based classification system and was trained upon a small dataset. It was tested on five distinct subjects of the KDD99 dataset [7], and like it Jirapummin came up with an IDS using hybrid neural network [6]. Wang suggested an efficient feature augmented SVM-based intrusion detection system. The feature-augmented approach is applied in the detection framework to offer the SVM classifier with brief, high quality training data, which not only enhances the SVM's detection capabilities but also shortens the training period [11]. Li worked with Naïve Bayes and on all the data sets he analyzed, the proposed approach obtained a detection rate of almost 100% and a false positive rate of practically 0% [8], While Subba observed that the binary class and the multiclass classification problems are successfully solved by intrusion detection models based on LDA and LR. That they have outperformed the Naive Bayes-based model while as effective as the C4.5-based intrusion detection model and SVM models. However, the conclusion was that the SVM based model had substantially greater computational cost when compared with LDA and LR-based models [12]. Manish noticed the benefit of using decision trees models which is generalization accuracy. As with many intrusion detection models are

developed, there will also be some new attacks upon the system with slight variations of the well-known methods. Because decision trees generalization accuracy is so good, it is easy to identify these new invasions [9]. By random forest, features are eliminated repeatedly while the important value of each feature is computed. Using the acquired features, a Deep Multilayer Perceptron (DMLP) structure can identify intrusions with a 91% accuracy rate [10]. Upadhyay obtained a detection rate of 98.5% that accurately distinguishes between non-attack and attack vectors with just 6.7% and 3.7% of false positives for three class and binary classification, respectively using gradient boosting [13]

3. Methods

In Feature selection, a representative collection of attributes is chosen from the initial set of attributes of the larger dataset. As the number of characteristics and features are reduced, the algorithm requires shorter time to train, creates additional generic classifier as it eliminates unnecessary features for the initial set. This representative set retains only the pertinent and significant attributes. The choice of features aids in both data understanding and visualization. The next section gives a quick overview of few of the common feature selection strategies employed in the study.

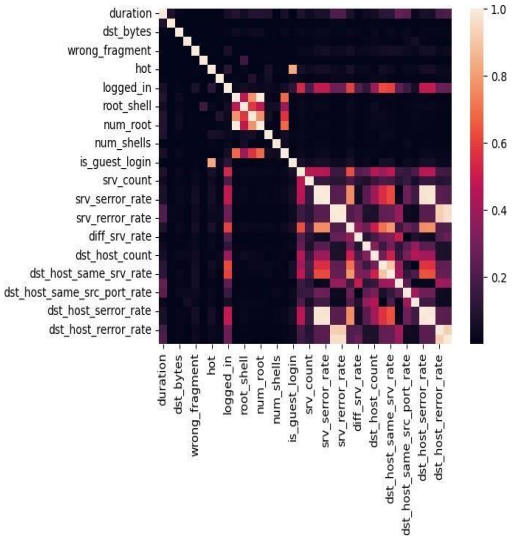

Fig. 2. Heat Map of selected features

A. Methods for selecting features based on correlation

CFS's working hypothesis states that good feature subsets have properties that are substantially linked with the class but uncorrelated with one another.

Algorithm:

1. Choose the dataset that will be pre-processed.
2. Determine the relationships between features and feature's classes.
3. Browse the feature subspace and determine the merit-based feature subset.

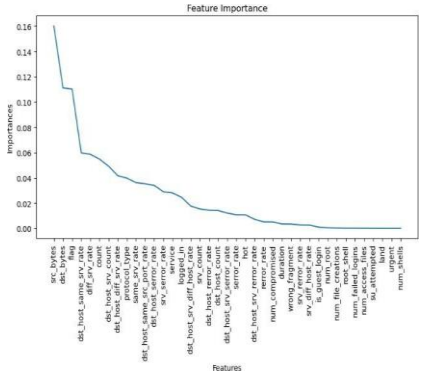

Fig. 3. Graph Representing Feature Importance

B. Principal Component Analysis

It is a statistical way to determine main components which are uncorrelated attributes.

Algorithm:

1. While taking the entire-dimensional dataset ignore the class labels.
2. The mean vector in dimension d is computed.
3. The covariance matrix is located for the entire dataset.
4. Eigen values and associated eigen vectors were calculated.
5. To create a d dimensional matrix M, the eigen vectors with the largest eigen valuesare selected after the eigen vectors with the lowest eigen values that are sorted.

C. Selection of features based on Information Gain ratio

It is a Machine Learning technique that has multiple values that are more likely to be chosen if only information gain is considered. It sometimes refers to the dimensionality curse. This disadvantage is eliminated by Information Gain Ratio (IGR) based Feature Selection, which takes

into consideration the splitting information of attribute. As the value of the split information rises, the gain ratio of the attribute falls.

Algorithm:

1. Begin the set of attributes (set including all characteristics from the dataset) with a null selected feature set.
2. Determine each attribute's IG ratio.
3. Pick the one highest information gain ratio from theentire list of attributes.
4. Depending on the attribute values, divide the dataset into sub- datasets.
5. Include the attribute and exclude it from attributeset.
6. Produce the chosen feature set.

D. Redundancy at a Minimum Highest Relevance

This approach tries to reduce the usefulness of features by penalizing its duplication. The total values between a dependent feature ftsx and the class cls define the relevance of feature set S for the class cls.

Equation illustrates it
Maximum B(A, cls),B=1 |A|P ftsx ∈ A I(ftsx; cls)
The mean value of all the features ftsx and ftsy representsthe redundancy of features in the set A.
Minimum N(A),N=1|A|2 P ftsx, ftsy ∈ AI(ftsx; ftsy)
Combining two metrics mentioned, the m RMR criterion is indicated in the equation.
Maximum
D(B,N),D=1|A|Pftsx∈AI(ftsx;cls)−1|A|2Pftsx,ftsy∈AI(ftsx;ftsy)
Near-optimal features specified by D are situated using incremental searching techniques. The features from the collection will be optimized in the below condition if we have sm1,the feature collection with (s-1).
maxftsy∈U−sm−1(PI(ftsy;cls)−1s−1Pftsx∈ss−1I(ftsy; ftsx))

Algorithm:

1. From the list of candidates, choose the characteristic ftsx that has the highest value formation.
2. Subtract fi from the candidate list andadd it to the subset.
3. From the candidate list,
   choose the following feature ftsy so it will maximize the condition (5).
4. Subset of the chosen feature ftsy and detach it from the candidate list.
5. Continue going through steps 3 to 4 until the subset can contain no more features.

4. Classifiers

Classifiers are used for sorting the classes. Many algorithms use this classifier which help the algorithm to sort the data into categories or classes which contains labels. It classifies the data into more than one category or class. Classifiers help in understanding the patterns.

A. Decision Tree

This algorithm is used for both regression and classification purposes. It has root nodes, internal nodes, and leaf nodes. Root nodes are also called as the parent node which is the starting of the decision tree. Internal nodes denote the characteristics of the dataset and the branches which divide the nodes of the tree represents the decision rules. Leaf nodes represent the output based on the conditions.

Algorithm:

1. Importing required libraries.
2. Importing dataset.
3. Splitting of dataset into the test set and training set.
4. Training the Decision Tree Classification model based on training set.
5. Predicting test set results.
6. Comparing predicted values with the actual values.
7. Calculation of confusion Matrix and Accuracy.
8. Visualize the Decision Tree. Decision algorithm has good accuracy. It is 99% accurate. Decision Tree, when done with correlation-based feature selection has more accuracy when compared with the other feature selection approaches.

B. Random Forest

This algorithm is one of the popular supervised ML algorithms. Random Forest follows regression and classification. This algorithm merges multiple classifiers to resolve the hard problem and to enhance the model's performance.

Algorithm:

1. From training the dataset, randomly select K data points.
2. From the data points which are selected, build the decision trees.
3. Repeat 1 and 2 steps.
4. Find predictions of every DT and all of the latest data points, for latest data points, find the predictions of each DT.

C. Gaussian Naïve Bayes

Naïve Bayes is an ML algorithm used for classification purposes. In this algorithm, we calculate the probability function. For normal distribution or gaussian distribution we calculate standard deviation and average for training data.

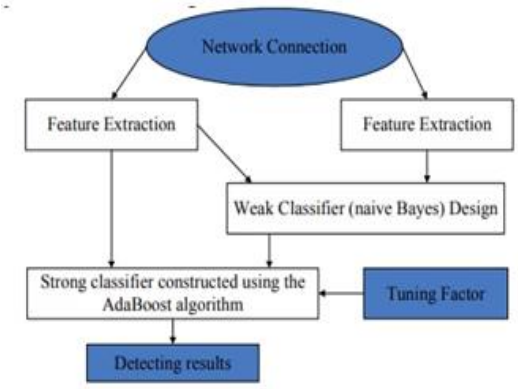

Fig. 4. Naïve Bayes Model for getting Results

Naïve Bayes is more accurate with Information Gain Ratio when compared with the other feature selection approaches

D. Supervised Machine Learning Model (SVM)

SVM is the most popular ML algorithm which is used for classification. This algorithm has a hyperplane which classifies the data into different categories or classes. The main aim of the SVM is to find out the hyperplane which classifies the data points into categories.

Algorithm:

1. Import libraries and dataset.
2. Extract x and y variables separately.
3. Divide the data set into test and train.
4. Initialize SVM classifier model.
5. Fitting the SVM classifier model.
6. Predicting the model.
7. Evaluating the performance of the model.

Support Vector Machine has produced high accuracy when it is done with PCA feature selection whereas the other feature selection approaches have less accuracy when compared with the PCA.

E. Logistic Regression

Logistic regression models are used for predicting analytics for multi-class classification. This algorithm helps to classify the variables into different classes. It is mainly used for finding the probability of a event which may be success or failure.

Algorithm:

1. Import libraries and dataset.
2. Train the dataset based on train and test approach.
3. Predicting the model.

F. XGBoost

Extreme Gradient Boosting is an extension of Gradient Boosted DT. A sequential form of decision trees is created in this algorithm. In XgBoost algorithm, weights play a vital role. This algorithm is specially meant to improve speed and performance.

To avoid overfitting, regularized terms are used to smooth final weights. Because of its distributed computing, Xgboost is the fastest algorithm when compared with the other algorithms.

G. Gaussian Naïve Bayes

Gradient Boosting is a ML algorithm which is used to construct predictive models. This algorithm is very useful when working with large datasets. This algorithm has high accuracy in predicting the models.

Algorithm:

1. Construct a model for the base.
2. Calculate the values of observed– predicted.
3. Find out outcomes of each DT.

5. Intrusion Detection System (IDS)

For classification, important features are selected from the dataset. A dataset which contains significant features will have the great model's accuracy. After the selection of the features, the

classifiers are trained. Using the same features, the test dataset is then tested. This test is done to determine the attacks on the data. It may be normal data or an attacked data.

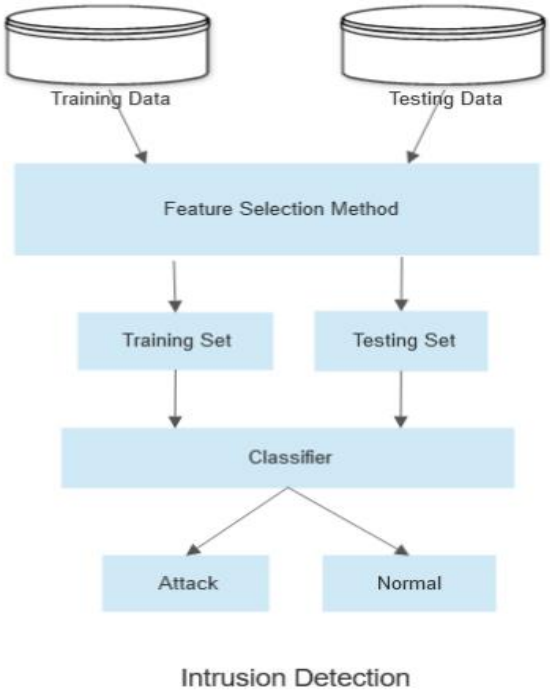

Fig. 5. Flow of Intrusion detection model

6. Experiment Setup

Jupiter notebooks, Google Colas, and the WEKA software, KDD-CUP-99 dataset we conducted this experiment.

7. Experimental Analysis

KDD-CUP-99 Dataset is used for this research. This dataset is used to get findings after five folds of cross validation.40% of data is trained and 60%is used for testing.

Decision Tree when done with correlation-based feature selection has more accuracy when compared with the other feature selection approaches.

Naïve Bayes is more accurate with Information Gain Ratio when compared with the other feature selection approaches.

Support Vector Machine has produced high accuracy when it is done with PCA feature selection whereas the other feature selection approaches have less accuracy when compared with the PCA.

8. Result of the Experiment

The experiments make use of KDP-CUP-99 Dataset. This dataset is exceptionally huge, with more than 40 columns and over 26000 rows of data. Huge datasets are challenging to work with since they raise the cost of computation.

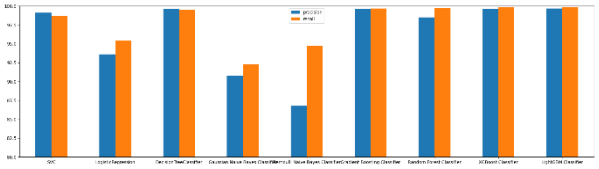

Fig. 6. Results of Precision and recall for all the 9 algorithms

The above graph represents the average of precision of all the algorithms what we compared with Recall of All Algorithms.

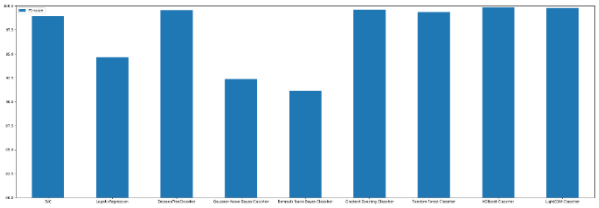

Fig. 7. F1 score analysis of the 9 algorithms

The Above graph shows the accuracy of Intrusion detection using all the Algorithms. It is observed from the experiment that boosting algorithms give 100 percent accuracy when compared with other algorithms

Table-1: Results of the Model

Model	f1-score	accuracy
SVC	0.99	0.99
Logistic Regression	0.93	0.93
Decision Tree	0.99	0.99
Gaussian NaiveBaye	0.91	0.92
Bernoulli NaiveBayes	0.89	0.90
Gradient Boosting	1.00	1.00
Random Forest	0.99	0.99
XGBoost	1.00	1.00
LightGBM	0.99	0.99

9. Conclusion

In this paper, we proposed an Intrusion Detection System Model to compare performances between the most popular Machine Learning algorithms. Some Famous Machine Learning algorithms that we used in this paper are SVC, Logistic Regression, Decision Tree, Gaussian Naïve Baye, Bernoulli Naive Baye, Gradient Boosting, Random Forest, XG Boost, and LightGBM. The experimental results show that Gradient Booting and XGBoost performed well when we compared them with other ML algorithms. These two algorithms are belonged to the Boosting Algorithms Family. So, this study concludes that Boosting algorithms are effective for Intrusion Detection Systems (IDS). In Future work we are planning to work on boosting algorithms as a future Scopus and we want to implement by adding degree like mean validations, standard deviations to improve the accuracy of Intrusion Detection System (IDS).

Refrences

[1] The 1994 paper by Landwehr, C.E., Bull, A.R., McDermott, and Choi a classification of vulnerabilities in computer software. 26(3), pp. 211-254, ACM Computing Surveys (CSUR).

[2] 2009 study by Tsai, C.F., Hsu, Y.F., Lin, C.Y., and Lin, W.Y. Machine learning for intrusion detection: A review of expert systems with applications, 36(10), pp. 11994–12000.

[3] Sangkatsanee, P., Wattanapongsakorn, N. and Charnsripinyo, C., 2011. Practical real-time intrusion detection using machine learning approaches. Computer Communications, 34(18), pp.2227-2235.

[4] M. Tavallaee, E. Bagheri, W. Lu, and A. A. Ghorbani, 2009, July a thorough examination of the KDD CUP 99 data set 2009 IEEE conference on uses of computational intelligence for security and defence (pp. 1-6). Ieee.

[5] M. Alshawabkeh, M. Moffie, F. Azmandian, J. A. Aslam, J. Dy, and D. Kaeli, December 2010. effective feature selection on extremely unbalanced data for virtual machine monitor 6 intrusion detection. The Ninth International Conference on Machine Learning and Applications was held in 2010 (pp. 823-827). IEEE.

[6] Kanthamanon, P. and Jirapummin, C. 2002. a system using hybrid neural networks for intrusion detection. In the IEEK Conference Proceedings (pp. 928-931). The Institute of Electronic and Computer Engineers.

[7] An ANN classifier-based feature-reduced intrusion detection system, Manzoor, I. and Kumar, N., 2017. 88, pp. 249–257 of Expert Systems with Applications.

[8] Li, W., and Li, Q. November 2010. enhancing network anomaly intrusion detection with naive Bayes and AdaBoost. third international conference on intelligent networks and systems was held in 2010 (pp. 486-489). IEEE.

[9] M. Kumar, M. Hanumanthappa, and T. S. Kumar, November 2012. Decision tree algorithm-based intrusion detection system. IEEE's 14th global conference on communication technology will be held in 2012 (pp. 629- 634). IEEE.

[10] S. Ustebay, Z. Turgut, and M. Aydin, December 2018. recursive feature elimination intrusion detection method with random forest and deep learning classifier. The IBIGDELFT, an international conference on big data, deep learning, and counterterrorism, was held in 2018. (pp. 71-76). IEEE.

[11] In 2017, Wang, H., Gu, J., and Wang, S. a powerful SVM-based featureaugmentation system for intrusion detection. pp. 130–139 in Knowledge-Based Systems, vol. 136.

[12] B. Subba, S. Biswas, and S. Karmakar, December 2015. technologies for detecting intrusions that use logistic regression and linear discriminant analysis. 2015 IEEE India Conference (INDICON) Annual Conference (pp. 1-6). IEEE.

[13] Upadhyay, D., Manero, J., Zaman, M. and Sampalli, S., 2020. Gradient boosting feature selection with machine learning classifiers for intrusion detection on power grids. IEEE Transactions on Network and Service Management, 18(1), pp.1104-1116