Science and Technology 2016, 6(4): 89-92
DOI: 10.5923/j.scit.20160604.01

Accuracy of Machine Learning Algorithms in Detecting DoS Attacks Types

Noureldien A. Noureldien*, Izzedin M. Yousif

Department of Computer Science, University of Science and Technology, Omdurman, Sudan

Abstract Attaining high prediction accuracy in detecting anomalies in network traffic is a major goal in designing machine learning algorithms and in building Intrusion Detection Systems. One of the major network attack classes is Denial of Service (DoS) attack class that contains various types of attacks such as Smurf, Teardrop, Land, Back and Neptune. This paper examines the detection accuracy of a set of selected machine learning algorithms in detecting different DoS attack class types. The algorithms are belonging to different supervised techniques, namely, PART, BayesNet, IBK, Logistic, J48, Random Committee and InputMapped. The experimental work is carried out using NSL-KDD dataset and WEKA as a data mining tool. The results show that the best algorithm in detecting the Smurf attack is the Random Committee with an accuracy of 98.6161%, and the best algorithm in detecting the Neptune attack is the PART algorithm with an accuracy of 98.5539%, and on the average PART algorithm is the best algorithm in detecting DoS attacks while InputMapped algorithm is the worst.

Keywords DoS Detection, DoS Attacks, NLS-KDD, Machine Learning Algorithms, WEKA

1. Introduction

Intrusion detection and prevention systems are commonly used as a major security tool to detect and prevent networks from malicious attacks. Intrusion detection systems are classified either as misuse or anomaly detection systems [1, 2]. Misuse detection systems, also known as signature based systems, detect known attack signatures in the monitored resources, while anomaly detection systems identify attacks by detecting changes in the pattern of utilization or behavior of the system.

Anomaly based intrusion detection systems are categorized into three basic techniques, statistical based, knowledge based and machine learning based [3, 4].

In machine learning based intrusion detection systems, machine learning algorithms are trained to learn system behavior. The learning process or technique is classified either as supervised or unsupervised. In supervised learning, the data used in training is normally labeled as normal or malicious.

During training, the machine learning algorithm attempts to find a model between data features and their classes so that it can predict the classes of new data, usually known as testing data.

Several machine learning-based schemes have been applied. Some of the most important are Bayesian networks, Markov models, Neural networks, Fuzzy logic, Genetic algorithms, and Clustering and outlier detection [3].

On the other hand, networks attacks are generally classified into four classes [5]. Probes, which are attacks targeting information gathering. Denial of Service (DoS), which are attacks that either denies resource access to legitimate users or render system unresponsive. Remote to Local (R2L), these are attacks in which an attacker bypass security controls and execute commands on the system as legitimate user, and User to Root (U2R), the attacks in which a legitimate user can bypass security controls to gain root user privileges.

Out of these four classes, DoS is the known to be the most common and serious network attack. DoS attack class constitutes various attacks such as, Smurf, Neptune, Land, Back, teardrop, and TCP SYN flooding.

Accordingly, building DoS attacks intrusion detection systems becomes an interested research area and many machine learning based anomaly intrusion detection systems have been proposed.

To compare the detection accuracy of these systems, researchers compare the detection accuracy of machine learning algorithms deployed in the heart of these systems.

Most of the research work in the literature is centered on examining the performance of machine learning algorithms in detecting DoS attack as a class rather than focusing on a specific DoS attack type.

The possibility that one machine learning algorithm may out performs other algorithms in detecting a specific DoS

* Corresponding author:
noureldien@hotmail.com (Noureldien A. Noureldien)
Published online at http://journal.sapub.org/scit

attack type, is the motivation of this work.

In this paper, we provide a comprehensive set of simulation experiments to evaluate the performance of different machine learning algorithms in detecting different types of DoS attacks.

The rest of this paper is organized as follows; in Section 2 a related work is presented. In Section 3 the experimental environment is explained. Section 4 shows the experimental results and conclusions and future work are drawn in Section 5.

2. Related Work

Machine learning based systems use machine learning algorithms or classifiers to learn system normal behavior and build models that help in classifying new traffic. Machine learning techniques are based on establishing an explicit or implicit model that enables the patterns analyzed to be categorized.

Developing an optimum machine learning based detection systems directs research to examine the performance of a single machine learning algorithm or multiple algorithms to all four major attack categories rather than to a single attack category.

G. Meera Gandhi [6], examined the performance of four supervised machine learning algorithms in detecting the attacks in the four attack classes categories; DoS, R2L, Probe, and U2R. The results indicate that the C4.5 decision tree classifier outperforms in prediction accuracy the other three classifiers Multilayer Perception, Instance Based Learning and Naïve Bayes.

Nguyen and Choi evaluate a comprehensive set of machine learning algorithms on the KDD'99 dataset to detect attacks on the four attack classes [7].

Abdeljalil and Mara [8], have compared the performance of the three machine learning algorithm; Decision Tree (J48), Neural Network and Support Vector Machine. The algorithms were tested based on detection rate, false alarm rate and accuracy of four categories of attacks. From the experiments they found that the Decision tree (J48) algorithm outperformed the other two algorithms.

Sabhnani and Serpen, have assessed the performance of a comprehensive set of machine learning algorithms on the KDD'99 Cup intrusion detection dataset [9]. Their simulation results demonstrated that for a given attack category certain classifier algorithms performed better.

Zadsr and Daved [10], have compared the performance of two algorithms, an adaptive threshold algorithm and a particular application of the cumulative sum (CUSUM) algorithm for change point detection, in detecting SYN flooding attack.

Yogendra and Upendra [11], evaluates the performance of J48, Bayesnet, OneR, and NB algorithms, they conclude that J48 is the best algorithm with high true positive rate (TPR) and low false positive rate (FPR).

Unlike above studies, our work concentrates on examining detection accuracy of machine learning algorithms on different DoS attacks to determine which algorithm is better for specific DoS attack.

3. Experimental Environment

This section discusses the dataset used in experiments, and the measure used to evaluate algorithm's performance. All experiments were performed using a laptop with windows7 Ultimate operating system, Intel® Atom™ Cpun2700 processor, and 1.00 GB.

The KDD'99Cup dataset has been widely used for the evaluation of anomaly detection methods. KDD'99 is prepared and built based on the data captured in DARPA'98 IDS evaluation program [12, 13].

KDD dataset is divided into labeled and unlabeled records; labeled records are either normal or an attack. Each labeled record consisted of 41 attributes or features [14].

KDD'99 contains different types of attack classes. Each attack type is recognized by a set of features. Table (1) shows the attacks classes in KDD'99 dataset, and Table (2) shows the most relevant features of each DoS attack type in KDD'99 [5].

Table (1). Attack Classes in KDD'99

Attack Class	Attack Name
DoS	Smurf, Land, Pod, Teardrop, Neptune, Back
R2L	Ftp_write, Gess_pass, Imap, Multihope, phf, spy, warezmaster, warezclient
U2R	Perl, buffer_overflow, Rootket, Loadmodule
Probe	Ipsweep, nmap, portsweep.

Table (2). Most Relevant Features of DoS attacks on KDD Dataset

Class Label	Relevant Features
Land	7
Smurf	2,3,5,23,24,27,28,36,40,41
Neptune	4,25,26,29,30,33,34,35,38,39
Teardrop	8
Back	10,13

KDD99Cup data set has a huge number of redundant records for about 78% and 75% are duplicated in the train and test set, respectively [15].

To solve these issues, a new dataset, NSL-KDD was proposed, which consists of only selected records form the complete KDD dataset and does not suffer from any of the mentioned shortcomings [16]. NLS-KDD contains four files, the KDD Train+.txt file which contain full NSL-KDD train set including attack-type labels, the KDD train+_20Percent which is a 20% subset of the KDD Train+.txt file, the KDDTest+.txt which is the full NLS-KDD test set and KDDTest-21 which is a subset of KDDTest+.txt file.

Table (3) shows the number of classes' records in each of the four datasets, and Table (4) shows the number of records

of each DoS attack types in the four datasets.

Table (3). Numbers of classes records in datasets

Dataset	Normal	Dos	U2R	R2L	Probe	Total
Train+	67343	45927	993	54	11656	125973
Train+_ 20Per	13449	9234	206	12	2289	25190
Test+	9711	7458	2421	533	2421	22544
Test-21	2152	4342	2421	533	2402	11850

Table (4). Number of DoS attacks records in datasets

Dataset	Back	Neptune	Smurf	Teardrop	Land
Train+	956	41214	8649	892	18
Train+_20Per	196	8282	529	188	1
Test+	359	4657	665	12	7
Test-21	359	1579	627	12	7

The experiments will be carried out using Train+20 percent for training and Test-21 for testing. To test and evaluate the algorithms we use 10-fold cross validation to ensure that algorithms will perform on unseen data [17, 18].

We use WEKA-3.6 as a data mining tool to select and evaluate accuracy of algorithms. Table (5) shows the algorithms that we select to use in the experiments.

Table (5). The Selected Machine Learning Algorithm

Category	The Selected algorithm
Rule	PART
BayesNet	BayesNet
Lazy	IBK
Function	Logistic
Tree	J48
Meta	Random Commitee
Misc	Input Mapped Classifer

To measure the detection accuracy of algorithms, we use the correctly and incorrectly classified instances, which show the percentage of test instances that were correctly and incorrectly classified.

4. Results

Table (6) and Fig (1) below shows a summary of testing experiments of algorithms against different types of attacks. The percentage of correctly classified instances is reported.

From table (6) we deduce that:

1- All algorithms perform poorly and equally in detecting Land, Teardrop and Back attacks. This result is due to the fact that these attacks are recognized with only one and two features with very few records in the training and testing dataset. Thus the few numbers of features and records in the dataset conceal the individual characteristics of classification algorithms.

2- The Random Committee algorithm is the best algorithm in detecting Smurf attack with 98.6161% accuracy, with insignificant difference from PART and J48 with 98.5495% and 98.5362% respectively, while PART algorithm is the best algorithm in detecting Neptune attack with 98.5539% accuracy, with significant difference from Random Committee and J48 with 88.5069% and 88.3251% respectively.

3- On average PART is the best algorithm in detecting DoS attacks and InputMapped is the worst.

Table (6). Algorithms Percentages of Correctly Classified Instances

Algorithm	Land	Smurf	Neptune	Teardrop	Back
PART	56.9242	98.5495	98.5539	57.0529	57.1017
BayesNet	56.9242	95.0763	80.283	56.9819	56.044
IBK	56.9242	97.3873	93.3641	57.0529	57.1194
Logistic	56.9242	97.3873	93.3641	57.0529	57.1194
J48	56.9242	98.5362	88.3251	57.0529	57.1017
Random-Committee	56.9242	98.6161	88.5069	57.0529	57.1194
Input Mapped	56.9242	56.9242	56.9242	56.9242	56.9242

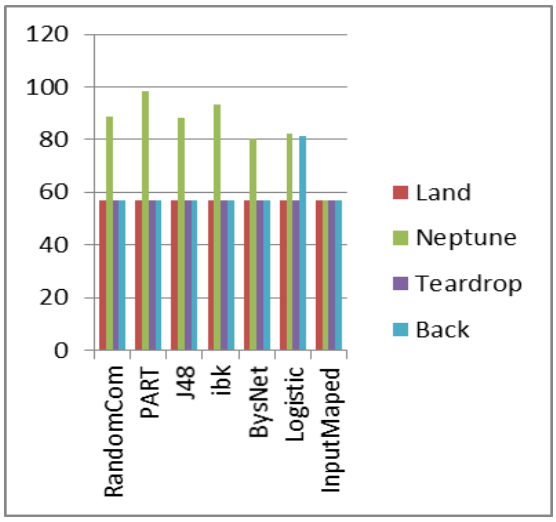

Figure (1). Algorithms Detection Performance

5. Conclusions and Recommendations for Future Work

The experimental work in this paper shows that the machine learning algorithms perform differently in detecting DoS attacks, and their performance is directly affected by the amount of attack features and records in the testing dataset.

On average, the PART algorithm is the best algorithm to be implemented by DoS attack intrusion detection systems, while InputMapped algorithm is the worst.

Our future work is to build intrusion detection systems using different machine learning algorithms and to punish mark these systems using various DoS attacks.

REFERENCES

[1] V. Chandola, A. Banerjee and V. Kumar. Anomaly detection: A Survey, ACM Computing Surveys, 41(3), pp.1-58, 2009.

[2] C. F. Tsai, Y. F Hsu and C. Y. Lin, Intrusion detection by machine learning: A review, Expert Systems with Applications, 36(10), pp.11994-12000, 2009.

[3] P. Garcıa-Teodoroa, J. Diaz-Verdejo, G. Macia-Fernandez and E. Vazquez, Anomaly-based network intrusion detection: Techniques, systems and challenges, ELSEVIER, computers & security 28 (2009) 18–28.

[4] V. Jyothsna and V. Rama, A Review of anomaly based intrusion detection systems, International Journal of Computer Applications, Vol. 28– No.7, September 2011.

[5] A. O. Adetunmbi, S. O. Adeola, and O. A. Daramola, Analysis of KDD'99 intrusion detection dataset for selection of relevance features, Proceedings of the World Congress on Engineering and Computer Science, Francisco, USA, Vol I WCECS pp. 20-22, 2010.

[6] G. Meera Gandhi, Machine learning approach for attack prediction and classification using supervised learning algorithms, International Journal of Computer Science & Communication, Vol. 1, No. 2, pp. 247-250. July-December 2010.

[7] N. Huy and C. Deokjai, Application of data mining to network intrusion detection: Classifier Selection Model, APNOMS 2008, LNCS 5297, pp. 399–408, Springer-Verlag Berlin Heidelberg 2008.

[8] K. AbdJalil, and S. Mara, Comparison of machine learning algorithms performance in detecting network intrusion, In Proceedings of Networking and Information Technology (ICNIT), pp. 221 – 226, Manila 2010.

[9] S. Maheshkumar, and S. Gursel, Application of machine learning algorithms to KDD Intrusion detection dataset within misuse detection context, In proceedings of MLMTAP conference, pp. 209-215, 2003.

[10] B. Zadsr, and G. Daved, Anomaly detection algorithms for detecting SYN flooding attacks, Computer Communications, Vol. 29, Issue 9, pp. 1433–1442, 2006.

[11] K. Yogendra and Upendra, An efficient intrusion detection based on decision tree classifier using feature reduction, International Journal of Science and Research Publication, vol. 2, issue 1, January 2012.

[12] S. J. Stolfo, W. Fan, W. Lee, A. Prodromidis, and P. K. Chan, Cost based modeling for fraud and intrusion detection: Results from the jam project, discex, vol. 2, pp. 1130, 2000.

[13] R. P. Lippmann, D. J. Fried, I. Graf, J. W. Haines, K. R. Kendall. McClung, D. Weber, S. E. Webster, D. Wyschogrod, R. K. Cunningham, and M. A. Zissman, Evaluating intrusion detection systems: The 1998 DARPA off-line intrusion detection evaluation, discex, vol. 2, p. 1012, 2001.

[14] W. Lee, S. J. Stolfo, and K. W. Mok, "Mining in a data-flow environment: Experience in network intrusion detection," In Proceedings of the 5 th ACM SIGKDD, SanDiego, CA, pp. 114-124, 1999.

[15] KDD Cup 1999. Available on: http://kdd.ics.uci.edu/databases/kddcup 99/kddcup99.html,

[16] Nsl-kdd data set for network-based intrusion detection systems. Available on: http://nsl.cs.unb.ca/kdd/nslkdd.html

[17] H. Günes, A. NurZincir-Heywood, and M. Heywood, Selecting features for intrusion detection: A feature relevance analysis on KDD'99 intrusion Detection datasets, Proceedings of the third annual conference on privacy, security and trust, New Brunswick, Canada, October 2005.

[18] K. T. Chui, K. F. Tsang, C. K. Wu, F. H. Hung, H. R. Chi, H. S. -H. Chung, K. F. Man, and K. T. Ko, Cardiovascular diseases identification using electrocardiogram health identifier based on multiple criteria decision making, Expert Systems with Applications, vol. 42, no. 13, pp. 5684-5695, Aug. 2015.