DOI: http://dx.doi.org/10.26483/ijarcs.v12i4.6748

ISSN No. 0976-5697

Volume 12, No. 4, July-August 2021

International Journal of Advanced Research in Computer Science

RESEARCH PAPER

Available Online at www.ijarcs.info

COMPARATIVE EVALUATION OF MACHINE LEARNING METHODS FOR NETWORK INTRUSION DETECTION SYSTEM

Sunil Kumar Rajwar
Assistant Professor
University Dept. of Computer Applications,
Vinoba Bhave University ,Hazaribag,Jharkhand

Dr. I Mukherjee
Assistant Professor
Department of Computer Science & Engineering,Birla Institute of Technology, Mesra, Ranchi ,Jharkhand , India

Dr. Pankaj Kumar Manjhi
Assistant Professor
University Department of Mathematics,
Vinoba Bhave University,Jharkhand,India

***Abstract*:** Cyber security is becoming more sophisticated, and as a result, there is an increasing challenge to accurately detect intrusions. Lack of intrusion prevention can degrade the credibility of security services, namely data confidentiality, integrity and availability. Many intrusion detection methods have been suggested in the literature to address threats to computer security, which can be broadly classified into signature-based intrusion detection (SIDS) and anomaly-based intrusion detection systems. (AIDS). This research presents the contemporary taxonomy of IDS, a comprehensive review of important recent work, and an overview of commonly used datasets for assessment purposes. It also presents detail analysis of different machine learning approach for intrusion detection.

Keywords: KDD99 Datasets, Weka, Network Anomalies

1. INTRODUCTION

Outlier detection refers to the problem of finding patterns in the data that do not meet the expected normal behaviour [1]. These anomalous patterns are often referred to as anomalies, inconsistent observations, exceptions, glitches, defects, noise, errors, or contaminants in various application domains. Outlier detection is a widely researched problem and finds great use in a wide range of application domains such as credit cards, insurance, tax fraud detection, cyber security intrusion detection, critical security system flaw detection, military surveillance for enemy activities and many more.

. The importance of Outlier detection system from the fact that anomalies in the data translate into meaningful information across a wide range of application domains. For example, an abnormal traffic pattern on a computer network can cause a infected computer to send confidential data to an unauthorized destination. In public health data, outlier detection techniques are widely used to detect abnormal patterns in patients' medical records that may be symptoms of a new disease. Similarly, migrants in credit card transaction data can indicate theft or abuse of credit cards [2]. Our research aim is to provide the basic understanding of network anomalies and their different type of detection system. We also evaluate different machine learning approach for network anomaly detection and their results based on standard network dataset with machine learning tools.

2. NETWORK ANOMALIES:

Network anomalies generally refer to circumstances in which network operations deviate from normal network behaviour [3]. Network anomalies can arise for a variety of reasons, including malfunctioning network devices, network overload, malicious denial-of-service attacks, and network outages that interfere with the normal delivery of network services. These anomalous events will interfere with the normal behaviour of some measurable network data.

Defining normal network behavior for the measured network data depends on a number of network-specific factors, such as the dynamics of the network under study in terms of traffic volume, type of network data available, and types of applications running on the network. Precise modeling of normal network behavior remains an active area of research, especially online modeling of network traffic [4]. Commercially available network management systems today continuously monitor a set of measured indicators to detect anomalies in the network.

A human network administrator observes alarm conditions or threshold violations generated by a group of individual indicators to determine the health of the network. These alarm conditions indicate a deviation from normal network behavior and can occur before or during abnormal events. These deviations are often related network performance degradation.

Network anomalies can be classified into two categories:

2.1. Network failures and performance Anomalies.

Typical examples of network performance anomalies are file server failures, network-wide localization, broadcast storms, babbling nodes, and transient congestion [5], [6]. For example, file server failures, such as web server failures, can occur when the number of ftp requests to that server increases. Network paging errors occur when an application program reaches the memory limits of the workstation and when it begins to call a network file server. This anomaly may not affect the individual user, but it does affect other users on the network by causing a shortage of bandwidth on the network. Broadcast storms refer to situations where broadcast packets are heavily used to shut down the network. A babbling node is a situation where a node sends small packets in an infinite loop to verify certain information, such as status reports. Congestion occurs on short timescales due to hot spots in the network that can result in connection failures or excessive traffic load at that point in the network. In some cases, software problems can manifest as network anomalies, such as a protocol implementation error that triggers increased or decreased traffic load characteristics. For example, a receive protocol error on a super server results in reduced network access, which in turn affects network traffic loads.

2.2. Security related Anomalies.

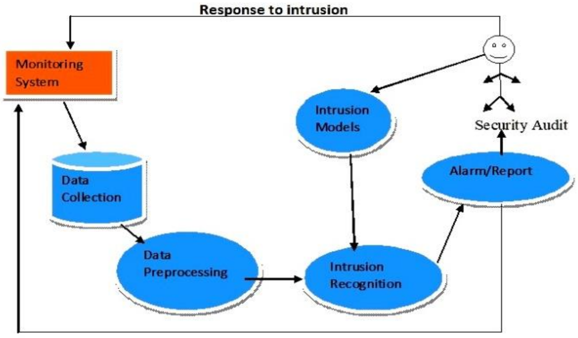
Denial of service attacks and network disruption are examples of these types of anomalies. Denial of service attack occurs when the services offered by a network are hijacked by some malicious entity. The offending party may disable vital services such as domain name server search (DNS) and virtual network shutdown [7], [8]. For this incident, the anomaly can be characterized by very poor performance. In the event of network disruption, the malicious entity may hijack network bandwidth by causing unnecessary flooding of the network, avoiding other legitimate users [9], [10]. This anomaly would result in a large volume of traffic.

3. NETWORK ANOMALY DETECTION

The enormous growth in the use of computers on a network and the development of applications that run on multiple platforms is highlighting network security. This paradigm exploits security vulnerabilities in all technically difficult and expensive computer systems. Therefore, intrusion is used to compromise the integrity, availability and confidentiality of a computing resource [11].

The Intrusion Detection System (IDS) plays a critical role in detecting anomalies and attacks on the network. The concept of data mining is integrated with IDS to effectively identify relevant, hidden data of user interest and with less execution time [12].

Figure 1 shows the general architecture of IDS. It is centrally located to capture all incoming packets transmitted over the network. The data is collected and sent for pre-processing to remove noise; irrelevant and missing attributes are added. The preprocessed data are then analyzed and classified according to their severity measure. If the log is normal, it does not require further changes or is submitted for reporting to generate alarms. Based on the status of the data, alarms are triggered so that the administrator can handle the case in advance. The attack is modeled to allow the classification of network data. The whole above process continues as soon as the transmission starts.



3.1. Intrusion Detection System (IDS)

4. NETWORK ANOMALIES DETECTION TECHNIQUES:

An intrusion detection system (IDS) revolves around the assumption that user behavior is observable and that normal user behavior is different from intrusive behavior [13]. At the core of intrusion detection is the ability to distinguish between acceptable normal system behavior and abnormal (possibly indicating unauthorized activity) or actively harmful [14]. It is possible to distinguish between two approaches to this problem, with some IDS applying a combination of both approaches.

While detecting anomalies in the network appears to be very simple, we need to obtain the data that does not follow normal patterns of behavior. Despite the large number of techniques available, the research challenges are as follows:

- Lack of a universally applicable anomaly detection technique; for example, a wireless network may not make much use of an intrusion detection technique.
- Noise in data is usually a real anomaly and therefore difficult to separate.
- Lack of a publicly available tagged data set that is used to detect a network failure.
- Since normal behaviors are constantly evolving and may not be forever, current invasion detection techniques may not be useful in the future . Need for newer and more sophisticated techniques because the invaders already know the prevailing techniques.

4.1. Misuse Detection

Misuse detection model capable to find abnormal behaviors and compare network traffic to a signature base for known attacks [15] any match clearly indicates system abuse. For example, an HTTP request referring to the cmd.exe file may indicate an attack. Misuse detection technique reduced false alarms compared to anomaly detection. The anomaly and misuse detection techniques differ from each other in a way that anomaly detection uses the standard data model to detect anomalous activities, while the signature abuse detection model uses several known attacks and seeks to occur in network data. The advantage of detecting misuse over anomaly detection is higher accuracy and fewer false alarms for known attacks. The problem with misuse detection models is how to show the signatures of each potential attack and how to write signatures that are very different from the normal data pattern. Another implicit problem in the misuse detection model is how to update the signature base when new attacks appear on the platform.

4.2. Anomaly Detection

An anomaly detection model attempts to model normal behavior. This technique looks at user behavior over time and creates the model that faithfully reflects a legitimate (normal) user behavior. Events that are very different from this model are considered suspicious. For example, a passive public web server trying to open links with a large number of addresses can be an indication of a worm infection. Anomaly detection generates an alert for any activity that looks unusual, making it ideal for detecting zero attacks. The problem with the anomaly detection model is how to define a model for normal behavior and how to handle a user's evolutionary normal behavior. Another disadvantage of the anomaly detection system is the return of high false positives. This is a result of their inability to change and adapt over time [16].

4.3. Hybrid Approach

Signature and anomaly detectors are often used together to complement each other. This combination of signature and anomaly detection techniques has resulted in a hybrid approach. This hybrid approach combines the positive benefits of both techniques. A survey shows that a hybrid technique works better than any technique. The problem with a hybrid approach is the added complexity of putting the two approaches together to form a complex system, the order in which both must process the data.

5. MACHINE LEARNING

Machine learning is a sub-area of computer science that has emerged from the study of pattern recognition and the theory of computer learning in artificial intelligence. It examines the construction and study of algorithms that can learn and predict data [17]. These algorithms work by building a model from sample inputs to make predictions or decisions based on data, rather than following completely static programming instructions.

Machine learning schemes are widely deployed and developed in today's intrusion detection community to improve anomaly detection performance. In particular, neural networks [18], support vector machines [18], decision trees [20] are significant and meaningful schemes used in anomaly detection systems to improve classification speed and performance.

Machine learning is the study of algorithms that improve performance with experience and intended to computerize exercises; the machine performs all the necessary steps in a consummate and well maintained manner. It is a type of artificial intelligence that gives computers the ability to learn without being explicitly programmed [19]. It includes various learning techniques classified into supervised, unsupervised and reinforced learning based on the presence or absence of labeled data.

5.1. **Supervised learning** : It trains the program with labeled examples; therefore, the trained program can predict similar unlabeled samples. It includes prediction tasks, information extraction and compression.

5.2. **Unsupervised learning:** It works on the principle of finding the hidden pattern of the data by grouping or grouping similar data. It includes work such as pattern recognition and outline detection.

This article focuses on the relative examination of the topic of intrusion detection by applying various prediction procedures under supervised learning. This document focuses on various prediction techniques used for the experiment, including J48, Random Forests, Zero R, One R, NaïveBayesUpdatable, Naïve Bayes, Multilayer Perceptron, K star, AdaBoost, M1, and Bagging.

6. KDD CUP 1999 DATASETS

The KDD Cup 1999 [21] is an intrusion detection reference dataset. In this dataset, the connection between two network hosts in a record is expressed in terms of 41 attributes; of which 38 are continuous and numeric discrete and 3 are categorical attributes. All records are labeled as ordinary or as a specific type of attack. Attacks fall into one of four categories: Denial of Service (DoS), Remote Locally (R2L), User to Root (U2R), and Probe.

The KDD Cup 1999 data set, which is used to compare intrusion detection problems, is used in our experiments. The data set is a raw TCP simulation dump data collection over LAN over a 9 week period. Training data was processed on approximately 5 million connection records from seven weeks of network traffic and fifteen days of test data received approximately 2 million connection records. A set of tagged labels is provided for training purposes, and as soon as the classifier is trained, it is checked for effectiveness on a different set of unlabeled records. The training and test data are taken from the different distribution with test data containing some records that are not in the training set [20]. The training data consists of 22 different types of attacks and 39 attacks are present in the test data [21]. This is the standard data set for detecting interferences and for the last decade and a half and in this work, this data set has been used to assess the feasibility of various procedures.

Table-1: Attack Distribution on Data Sets

Dataset	DoS	U2R	R2L	Probe	Normal	Total
10% KDD	391458	4107	52	1126	97277	494020
Whole KDD	3883370	41102	52	1126	972780	4898430

Table-2: Attack Types of Dataset

Category	Attack Type
Probe	nmap, mscan, ipsweep, portsweep,satan, saint
DoS	Back, apache, mailbomb, land, neptune, pod, teardrop, smurf, teardrop, udpstorm
U2R	Perl, rootkit, ps, buffer_overflow, loadmodule, xterm attack
R2L	Guess_password, imap, ftp_write, imap, multihp, named, phf, snmpgetattack, warezmaster, worm, xsnoop, httptunnel, snmp_guess

7. MACHINE LEARNING TOOLS:

WEKA

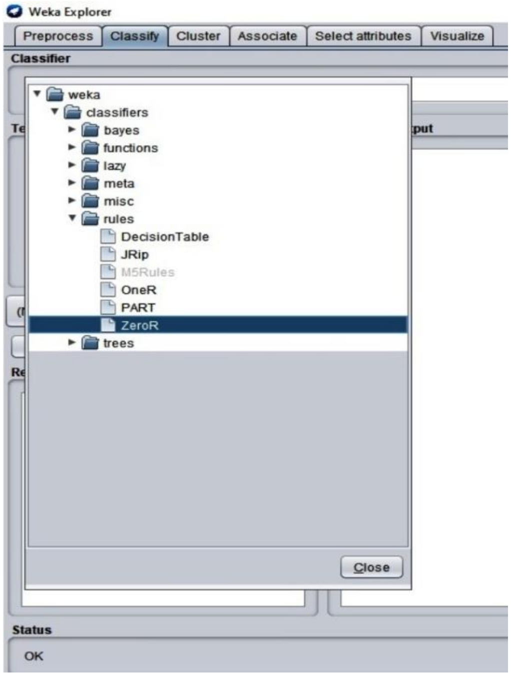
Weka is the collection of machine learning techniques for discovering information, which can be applied directly to data or called from Java code. Weka is open source software and free to use for most researchers in the field of data mining and knowledge discovery.

WEKA provides implementation of learning algorithms that we can easily apply on our dataset. It also includes various tools for changing datasets, such as algorithms for discrepancy and sampling. We can pre-process a dataset, enter it into a learning schema,and analyze the resulting classifier and its performance, all without writing any programming code fully.

The workbench includes methods for major data mining problems: regression, classification, grouping, association rules mining, and attribute selection. The data is for information only an integral part of the job, with many data visualization facilities and data pre-processing tools provided. Each algorithm takes their input in the form of one relative table that can be read from a file or generated by a database query.

One way to use WEKA is to apply a learning method to a dataset and analyze its output. Another method is to use learned models to generate predictions about new ones cases. The third is to apply a number of different learners and compare their performance in order select one for prediction.

Figure: SnapSnot of WEKA Explorer:



In the snapshot given WEKA uses different machine learning techniques under different tabs.

8. FEATURE SELECTION:

The feature or attribute in a dataset is an important element that can affect the performance of machine learning techniques. A total of 41 feature of each record in the KDD99 dataset.

For better and effective result in the experiment, we choose weka tools to eliminate some of the non descriptive attributes. Table given illustrates several evaluation methods and search techniques to reduce non descriptive attributes.

Table 3: Feature Selection with WEKA:

S.no	Evaluation Methods	Search	Attributes Selected	Total
1.	CfsSubsetEval	BestFirst	2,3,4,5,6,7,8,14,23,30,36	11
		GreedyStepwise	2,3,4,5,6,7,8,14,23,30,36	11
2.	InfoGainAttributeEval	Ranker	5,23,3,24,36,2,33,35,34,30,29,4,6,38,25,39,26,12,32,37,31,40 ,41,27,28,1,10,13,8,22,16,19,17,11,14,7,18,9,15,20,21	41

The Table 3 given below presents the results of machine learning techniques on the reduced dataset consisting of 11 features. In the second set of experiments 11 features given by CfsSubsetEval and Best First Search are provided as input to each of the technique and results of each technique are documented

.Table 4: Results on Reduced Dataset:

Sno	Group	Technique	CC	TP Rate	FP Rate	Precision	Recall	F-Measure	RA	KS	MAE
1	**Rule Based**	**DT**	**99.745**	**0.997**	**0.001**	**0.997**	**0.9997**	**0.997**	**1.000**	**0.9957**	**0.0016**
		CR	**78.537**	**0.785**	**0.061**	**0.677**	**0.785**	**0.713**	**0.937**	**0.6318**	**0.0203**
		ZeroR	**56.837**	**0.568**	**0.568**	**0.323**	**0.568**	**0.412**	**0.500**	**0.000**	**0.0514**
		OneR	**98.081**	**0.981**	**0.005**	**0.978**	**0.981**	**0.978**	**0.988**	**0.9675**	**0.0017**
		PART	**99.946**	**0.999**	**0.000**	**0.999**	**0.999**	**0.999**	**1.000**	**0.9991**	**0.0001**

2	**Bayes Rule**	**BayesNet**	**99.718**	**0.997**	**0.000**	**0.998**	**0.997**	**0.997**	**0.997**	**0.9952**	**0.0003**
		NaiveBayes	**96.164**	**0.962**	**0.000**	**0.99**	**0.962**	**0.973**	**0.999**	**0.9539**	**0.0037**
		NBUpdatable	**96.164**	**0.962**	**0.000**	**0.99**	**0.962**	**0.973**	**0.999**	**0.9359**	**0.0037**
3	**Functions**	**MLP**	**99.279**	**0.993**	**0.001**	**0.993**	**0.993**	**0.991**	**0.999**	**0.988**	**0.001**
		SMO	**99.255**	**0.993**	**0.007**	**0.993**	**0.993**	**0.991**	**0.999**	**0.9874**	**0.793**
4	**Lazy Learners**	**IBk**	**99.869**	**0.999**	**0.000**	**0.999**	**0.999**	**0.999**	**1.000**	**0.9978**	**0.0001**
		Kstar	**99.768**	**0.998**	**0.000**	**0.998**	**0.998**	**0.998**	**1.000**	**0.996**	**0.0003**
		LWL	**98.041**	**0.98**	**0.008**	**0.964**	**0.98**	**0.972**	**0.999**	**0.9664**	**0.0038**
5	**Tree**	**DecisionStump**	**78.538**	**0.785**	**0.061**	**0.677**	**0.785**	**0.713**	**0.973**	**0.0203**	**0.6318**
		J48	**99.944**	**0.999**	**0.000**	**0.999**	**0.999**	**0.999**	**1.000**	**0.999**	**0.0001**
6	**Meta-Algorithm**	**AdaboostM1**	**97.592**	**0.976**	**0.006**	**0.959**	**0.976**	**0.967**	**0.993**	**0.959**	**0.0477**
7	**Misc**	**InputMappedClassifier**	**56.837**	**0.568**	**0.568**	**0.323**	**0.568**	**0.4214**	**0.500**	**0.000**	**0.0514**

9. CONCLUSION

In this paper, a comparative analysis of various machine learning strategies for network intrusion detection was performed. The experiments were carried on benchmark dataset (KDDCUP99) for intrusion detection. We have performed two sets of experiments, one on a full dataset having 41 features and one on the reduced one with only 11 elements attributes. Experiments showed that classification algorithms doesn't depend all the 41 features thus the technique could easily get better results with an appreciable cutback in resources needed by working on the same dataset with reduced number of attributes.

10. REFERENCES.

[1] Varun Chandola, ArindamBanerjee, Vipin Kumar, Outlier Detection: A Survey, ACM Computing Surveys, 2009.

[2] Prasanta Gogoi, D.K. Bhattacharyya, B. Borah, Jugal K. Kalita, A Survey of Outlier Detection Methods in Network Anomaly Identification, The Computer Journal (Volume: 54, Issue: 4, Apr. 2011)

[3] Ansam Khraisat, Iqbal Gondal, Peter Vamplew and Joarder Kamruzzaman, Survey of intrusion detection systems: techniques, datasets and challenges, Khraisat et al. Cybersecurity(2019)

[4] T. Ye, S. Kalyanaraman, D. Harrison, B. Sikdar, B. Mo, H. T. Kaur, K. Vastola, and B. Szymanski, "Network management and control using collaborative on-line simulation," in Proc. CNDSMS, 2000

[5] M. Thottan and C. Ji, "Using network fault predictions to enable ip traffic management," J. Network Syst. Manage., 2000.

[6] R. Maxion and F. E. Feather, "A case study of ethernet anomalies in a distributed computing environment," IEEE Trans. Reliability, vol. 39, pp. 433–443, Oct. 1990.

[7] G. Vigna and R. A. Kemmerer, "Netstat: A network based intrusuion detection approach," in Proc. ACSAC, 1998.

[8] J. Yang, P. Ning, X. S. Wang, and S. Jajodia, "Cards: A distributed system for detecting coordinated attacks," in Proc. SEC, 2000, pp. 171–180.

[9] H. Wang, D. Zhang, and K. G. Shin, "Detecting syn flooding attacks," in Proc. IEEE INFOCOM, 2002.

[10] S. Savage, D. Wetherall, A. R. Karlin, and T. Anderson, "Practical net- work support for ip traceback," in Proc. ACM SIGCOMM, 2000, pp. 295–306.

[11]Dhruba Kumar Bhattacharyya ,Jugal Kumar Kalita, Network Anomaly Detection:A Machine Learning Perspective , ISBN 9781466582088, Published July 5, 2013 by Chapman and Hall/CR,366 Pages

[12] Effective approach toward Intrusion Detection System using data mining techniques, Nadiammai, M.Hemalatha Egyptian Informatics Journal (2014) 15,

[13] Stallings William. Network and internetwork security: Principles and practice. Englewood Cliffs: Prentice Hall.

[14] Verwoerd, Theuns, Ray Hunt. Intrusion detection techniques and approaches. 15, s.l.: Elsevier, Computer Communications. 2002;25:1356-1365.

[15] Anonymous. Intrusion detection FAQ. May 19; 2010. Available:http://www.sans.org/
Available:http://www.sans.org/security- resources/idfaq/

[16] Shun Julian, Heidar Malki. Network intrusion detection system using neural networks. s.l.: IEEE, ICNC'08. Fourth International Conference. 2008.

[17] Machine learning.[Online]August6;2015.Available:https:// en.wikipedia.org/wiki/ Machine learning.

[18]Dong Ling Tong and Robert Mintram, "Genetic Algorithm-Neural Network (GANN): a study of neural network activation functions and depth of genetic algorithm search applied to feature selection", International Journal of Machine Learning and Cybernetics, Vol. 1, No. 1-4, pp. 75-87, 2010.

[19] Peddabachigari S., Abraham A., Thomas J., "Intrusion Detection Systems Using Decision Trees and Support Vector Machines", International Journal of Applied Science and Computations, Vol.11, No.3, pp.118-134, 2004.

[20] Sindhu, Siva S Sivatha, Geetha S, Kannan, A Decision tree based light weight intrusion detection using a wrapper approach. 1, s.l. : Elsevier, Expert Systems with applications. 2012;39:129-141.

[21] "KDD Cup 1999 Data." [Online]. Available: http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html