IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified
Vol. 6, Issue 2, February 2017

A Study on Data Mining & Machine Learning for Intrusion Detection System

Rashmi Ravindra Chaudhari[1], Sonal Promod Patil[2]

Second Year, ME CSE, GHRIEM, Jalgaon[1]

HOD CSE, GHRIEM, Jalgaon[2]

Abstract: This study paper describes a literature survey focused on machine learning (ML) and data mining (DM) methods for cyber analytics in support of intrusion detection. Descriptions of each ML/DM method are provided shortly. Based on the number of citations or the relevance of an emerging method, papers representing each method were identified, read, and summarized. Because data are so important in ML/DM approaches, some well-known data sets used in ML/DM are described. The complexity of ML/DM algorithms is addressed, discussion of challenges for using ML/DM for security is presented, and some suggestions on when to use a given method are provided.

Keywords: Machine learning, data mining, intrusion detection, etc.

1. INTRODUCTION

This report presents concepts from the field of Artificial Intelligence and Machine Learning are used to address some of the challenging problems faced in the computer security domain. Securing computer systems has become a daunting work these days with the rapid growth of the internet (both in terms of scalability and size) and the increasing complexity of communication protocols. The raging war between the security wrongdoer and information security professionals has become more strong than ever. New and complicated attack methods are being developed by attackers at an alarming rate by taking advantage of the complicated behaviour of today's networks. CERT has reported new 8064 vulnerabilities in the year 2006 and this figure has been significantly increasing over the past few years [CER07].

Although proactive means for achieving security have existed for a long time, these approaches have goal to prevent and/or detect of only known attacks. Novel attacks have been raising a long-standing problem in the field of information security have received considerable attention in the recent past. On the basis of analysis of basic behaviour of communication protocols (in the case of a network) or analysis of basic system calls (in the case of a single host) it is difficult to detect many novel attacks. For instance, an attack might be developed which operates in stealth mode, means it may hide its presence and avoid detection [WS02]. Also, increasing complexity of cryptographic mechanisms (like IPSec) has made this detection problem more intense. Another important problem in the computer security field that has been present for quite some time is that of insider threats. An employee at an organization is consider as a trusted user and the possibility of an attack from an insider is considered less probable. However, in the recent past, a study by CERT [CER05] showed that attacks made by insiders have been a cause of a lot of tangible and intangible losses to many institutions. Many of the insider threats may be unintentional; nevertheless it has become essential to ensure that insider behavior is in sync with the security policy of the organization.

Handling the above issues is quite expensive for organizations. Efficient rules for anomaly detection are formulated by security experts. They are also required to handle huge amounts of data. This adds an element of unreliability and also makes the entire process much slow. Also, checking whether compliance to security policy by insiders is being achieved, is a burdensome task for an administrator as normal behaviour changes over time.

In Computer Science, Machine Learning has been one of the most promising advancements in the past few decades and has found success in solving intricate data classification problems. It basically deals with constructing computer programs [Mit97] that instinctively improve with experience. The learning experience is given in the form of data and actual learning is achieved with the help of algorithms. Machine learning addresses the two main tasks that have the ability to grasp more from the given data and to make predictions about new data based on learning outcomes from the learning experience [Mal06]; both of which are complex and time-consuming for human analysts. Thus, machine learning is well-suited to problems that depend on rare, expensive and unreliable human experts. It has been successfully applied to the complex problems ranging from stellar analysis to medical diagnosis.

The work of detecting intrusions can be considered as a machine learning task as it involves the classification of behaviour into user and adversary behaviour. In this paper, we study some significant machine learning approaches towards solving some challenging computer security issues mainly relating to detecting intrusions which are described in later chapters.

IJARCCE
ISSN (Online) 2278-1021
ISSN (Print) 2319 5940
International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified
Vol. 6, Issue 2, February 2017

Intrusion Detection
Intrusion Detection is used to detect violation of a security policies of an organization. These violations may be caused by people external to the organization (i.e. attackers) or by employees of the organization (i.e. insiders). Even though this field has progressed to detect violations by attackers, insider violations are difficult to detect.

Motivations behind Intrusion Detection
Because of the following reasons, Intrusion Detection has received considerable motivation:

1. If an intrusion is detected quickly enough, an intruder can be identified quickly and removed from the system before any damage is done or any data are compromised. Even if the detection is not sufficiently timely to preempt the intruder, as early the intrusion is detected, the amount of damage done is that less and more quickly that recovery can be achieved.
2. An effective intrusion detection system can serve as a deterrent, so acting to prevent intrusion.
3. Intrusion detection allows the collection of information about intrusion techniques that can be used to strengthen the intrusion prevention facility.

Goals of Intrusion Detection
The goals of intrusion detection can be summarized as below:

1. Detect as many type of attacks as possible including those by attackers and those by insiders.
2. Detect as correctly as possible thereby minimizing the number of false alarms.
3. Detect the attacks in the shortest possible time.

Types of Intrusion Detection
The above requirements have triggered the development of different types of intrusion detection techniques that satisfy the above properties to an extent in the past few decades. These techniques can be classified on the basis of their functionality, as follows:

1. Signature-based detection (also known as Misuse detection):
In this technique, the behaviour of user is compared with known attack patterns. If a match is found, an alert is raised. This type is capable of detecting only known attacks.

2. Specification-based detection:
This technique is just opposite to the signature-based approach [Wol06]; legitimate behaviour is specified and any deviation from legitimate behaviour raises an alert. The complexities of many programs and the task of modeling such complex behaviour with precision are included as the challenges in this approach. On the contrary, rough specification reduces the sensitivity of the detector.

3. Anomaly detection:
Anomaly detection is the most promising technique of intrusion detection as it aims to detect novel attacks in addition to known attacks. In this type, the models for normal system operation [Lia05] are built by using the observable behaviours of a system. These behaviours may include audit logs, network sensors, system calls, etc. While building a model and also while classifying new instances, various statistical techniques are used. The disadvantage of this approach is the definition of normal behaviour. Expert domain knowledge may be required while making such profiles of normal behaviour.
Intrusion Detection systems can also be classified as network-based (which monitor network traffic) or host-based (which monitor operating system events). In [MM01] and [Bac99] a more detailed description of these can be found. As evident from the previous sections, anomaly detection is one of the most vital and challenging tasks in the computer security domain. The remainder of this report focuses on how machine learning and related ideas can be used to address this problem. In particular, we will see how decision making algorithms are able to identify the anomalous behaviour by intelligent analysis of previous network behaviour.

Machine Learning
This chapter describes the basics of machine learning. We first discuss the machine learning concept and thereafter describe the components and representation of a machine learning task in more formal terms.

Basic Concepts
Learning
In computer science, every computer action can be shown as a function with sets of inputs and outputs. A learning task may be considered as an estimation of this function by observing the sets of outputs and also inputs. The function estimating process usually consists of a search in the hypo report space i.e. the space of all such possible functions that might show the input and output sets under consideration.
The authors in [Nil96] officially describe the function approximation process. Consider a set of input instances X = (x1, x2, x3, ..., xn). Let ‘f’ be a function which is to be deduced by the learner. Let h be the learner’s hyporeport about f. Also, we assume a priori that both f and h belong to a class of functions H. The function f maps the input instances in X as, X h2H ! h(X)
A machine learning task may thus be defined as a search in this space H. This search results in approximating the relevant h is based on the training instances i.e. the set X. The approximation is then examined against a set of test instances which are then used to indicate the accurateness of h. The search requires algorithms which are efficient and which are best-fit for the training data [Mit97].

Knowledge Representation and Utilization
Machine learning may be divided into decision trees, neural networks, probability measures or other such

IJARCCE
ISSN (Online) 2278-1021
ISSN (Print) 2319 5940
International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified
Vol. 6, Issue 2, February 2017

representations, depending on the way in which the learned knowledge may be represented. On the basis of the type of input and the way in which the learned knowledge is utilized, is the more basic way to divide machine learning, as identified in [DL03]. This division involves:

• Learning for Classification and Regression: This is the most widely used method of learning which involves the classification and regression. Classification involves assigning a new instance into one of the fixed classes from a finite set of classes. Regression involves the detection of the new value on the basis of some continuous variable or attribute.

• Learning for Acting and Planning: In this case, the learned knowledge is used for selecting an action for an agent. In a purely reactive way, the action may be chosen, ignoring any past values. Alternatively, the output of classification or regression may be used to select an action based on the description of the current world state, by the agent. These approaches are useful for problem solving, planning and scheduling.

• Learning for Interpretation and Understanding: This type focuses on the the explaination and understanding of situations or events rather than just the accurate prediction of new instances. To derive this understanding, which is known as abduction, many separate knowledge elements are used.

Inputs and Outputs

The inputs and outputs to a machine learning task may be of different kinds. Generally, the inputs and outputs are in the form of numeric (both discrete and real-valued) or nominal attributes. Numeric attributes (including both discrete and real-valued) may have incessant numeric values whereas nominal values may have values from a pre-defined set. For instance, if temperature used as a numeric attribute, may have values like 25^{o}C, 28^{o}C, etc. On the other side, if it is used as a nominal attribute, it can take values from a fixed set such as high, medium, low. In many cases, the output may also be a Boolean value.

Defining a Machine Learning task

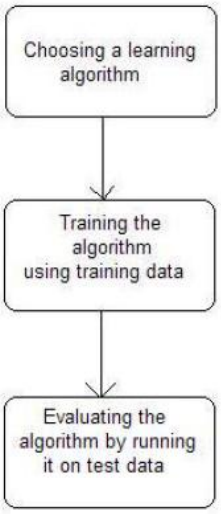
Generally, machine learning task can be defined in terms of three elements, and that are the learning experience E, the tasks T and the performance element P. [Mit97] elaborates a learning task more exactly as, a computer program is said to learn from experience E with respect to some class of tasks T and performance measure P, if its performance at tasks in T, as measured by P, improves with experience E.

This depiction of a machine learning task clearly states the requirements. It gives the concept of what the machine learning problem is, and what are its learning goals. It also tells how these goals can be measured so that the effectiveness of the task can be decided.

Life Cycle of a Machine Learning task

The life cycle of a machine learning task generally follows the process as

1. Choosing a learning algorithm.
2. Training the algorithm using a set of instances (referred to as the training set).



3. Evaluating the performance by running the algorithm on another set of instances (referred to as the test set).

Different types of algorithms may be chosen at different times depending on the nature of the knowledge to be learned. Also, the type of inputs and outputs are also useful in choosing an algorithm. These include rule inferring algorithms, algorithms based on statistical models, covering algorithms, divide and conquer approaches, algorithms for mining association rules, algorithms based on linear models, algorithms on instance-based learning and clustering algorithms.

Benefits of Machine Learning

The field of machine learning has been found to be extremely useful in the following areas relating to software engineering [Mit97]:

1. Data Mining gives problem where large databases may contain valuable implicit regularities that can be discovered automatically.
2. Difficult to understand domains where human beings might not have the knowledge to embellish effective algorithms.
3. Domains in which the program is required to adapt to dynamic conditions. In the case of traditional intrusion detection systems, the human analyst who evaluates them and takes a suitable action analyzes the alerts generated. However, this is an extremely burdensome task as the number of alerts generated may be quite large and the environment may change continuously [Pie04]. It makes machine learning more appropriate for intrusion detection.

2. LITERATURE SURVEY

A comparison of eager, lazy and hybrid learning algorithms for enhancing intrusion detection has been done in [Tes07]. A dataset produced from the honeypots of Tilburg University was used for this study. IB1 [AKA91] and RIPPER [Coh95] were the respective lazy

IJARCCE
ISSN (Online) 2278-1021
ISSN (Print) 2319 5940
International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified
Vol. 6, Issue 2, February 2017

and eager learning methods used to learn intrusions from the data set. Attribute selection was done using InfoGain [Mit97] and CFS [Hal99] mechanisms. as RIPPER has an inherent feature selection capability, this attribute selection was done only during the IB1 selection. A hybrid learning model was also used by combining the above two approaches. It was found that the most effective mechanism for classifying new intrusions on the concerned data set was RIPPER.

A new algorithm called LERAD was invented for learning useful system call attributes [TC05]. This is a conditional rule learning algorithm. It generates a small number of rules. The rules can be effectively used to detect more attacks with a reasonable space and time overhead. In this research, it was described that analysis of system call attributes is extremely vital and useful in detecting attacks and it performs better than systems which analyze just the sequence of system calls. This methodology mainly aimed at making the IDS deterrent to mimicry attacks [WS02].

In [IYWL06], a combination of signature-based and machine learning-based intrusion detection systems is shown as very useful. The authors showed that when a signature-based ID is used as the main system and the machine learning-based IDS is used as a supporting system. The supporting system can filter out the false alarms and also can validate the decisions of the main system. Snort was used as the signature-based IDS in the machine learning-based IDS. An extended IBL (Instance-based Learner) [AKA91] was also used as the algorithm in the machine learning-based IDS. Another important research work was done in [Lia05], where the task of detecting intrusions was related with a text mining task. The frequency of system calls was an important element in this report. The k-Nearest neighbor learning methodology [Mit97] was used to categorize intrusions based on the frequency of system calls. A cost-based model for checking the interdependence between the IDS and the attacker was also presented, which was shown to be quite effective.

In [SS03], it was shown that some specific attack categories are better detected by some specific algorithms. A multi-classifier machine learning model using these individual algorithms was built and to predict the attacks in the KDD 1999 Cup Intrusion Detection dataset. Various nine algorithms representing a variety of fields were selected for this analysis. False alarm rate and probability of detection were used as the performance measures. Empirical results showed that a noticeable performance improvement was achieved for certain probing, DoS and user-to-root attacks.

A. Proposed Architecture

The framework consists of three Levels:

Level 1: In this level the basic features are produced from network traffic ingress to internal network. At internal network the proposed servers resides in and are used to form the network traffic records for well-defined time period. Monitoring and analyzing network to decrease the malicious activities only on relevant inbound traffic.

To provide a best protection for a aimed internal network, this also allows our detector to provide protection which is the best fit for the aimed internal network because legitimate traffic profiles used by the detectors are developed for a smaller number of network services.

Level 2: In this step the Multivariate Correlational Analysis is applied in which the Triangle Area Map Generation module is applied to extract the correlation between two separate features within individual traffic record.

The distinct features are come from level 1 or "feature normalization module" in this step. All the extracted correlations are stored in a place called Triangle Area Map (TAM), are then used to substitute the original records or normalized feature record to represent the traffic record. It's differentiating between legitimate and illegitimate traffic records.

Level 3: The anomaly based finding mechanism is adopted in decision making. Decision making involves two phases as

Training phase.

Test phase

Normal profile generation module is work in "Training phase" to generate a profile for various types of traffic records and the generated normal profiles are stored in a database. In the "test phase", the "Tested Profile Generation" module builds the profiles for individual observed traffic records. Then at the end, the tested profiles are handed over to "Intruder Detection" module it compares tested profile with stored normal profiles. This distinguishes the Intruder from legitimate traffic.

This needs the expertise in the targeted detection algorithm and it is manual task. Particularly, two levels (i.e., the Training Phase and the Test Phase) are included in Decision Making. The Normal Profile Generation module is operated in a Training Phase [1] to generate profiles for various types of legal records of traffic, and the normal profiles generated are stored in the database. The tested profile generation module is used in a Test Phase to build profiles for the each observed traffic documentation. Next, the profiles of tested are passed over to an Intruder Detection part, which calculates the tested profiles for individual with the self-stored profiles of normal. A threshold based classifier is employed in the Intruder Detection portion module to differentiate Intruders from appropriate traffic [8].

B. Multivariate Correlation Analysis

Intruder traffic treat differently from the appropriate traffic of network and the behaviour of network traffic is reflected by its geometric means. To well describe these statistical properties, here a novel multivariate correlation analysis (MCA) moves toward in this part. This multivariate correlation analysis approach use triangle area for remove the correlative data between features within a data object of observed (i.e. a traffic record).

IJARCCE
ISSN (Online) 2278-1021
ISSN (Print) 2319 5940
International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified
Vol. 6, Issue 2, February 2017

C. Detection Mechanism

In this section, we present a threshold based on anomaly finder whose regular profiles are produced using purely legal records of network traffic and utilized for the future distinguish with new incoming investigated traffic report. The difference between an individual normal outline and a fresh arriving traffic record is examined by the planned detector. If the variation is large than a pre-determined threshold, then a record of traffic is coloured as an attack otherwise it is marked as the legal traffic record.

D. Algorithm for Normal Profile Generation

In this algorithm [1] the normal profile Pro is built through the density estimation of the MDs between individual legitimate training traffic records (TAM normal, i, lower) and the expectation (TAM normal, lower) of the g legitimate training traffic records.

In the training phase, we employ only the normal records. Normal profiles are built with respect to the various types of appropriate traffic using the algorithm describe below. Clearly, normal profiles and threshold points have the direct power on the performance of the threshold based detector. An underlying quality usual shape origins a mistaken characterization to correct traffic of network.

E. Algorithm for Intruder Detection

This algorithm is used for classification purpose.

Step1: Task is to classify new packets as they arrive, i.e., decide to which class label they belong, based on the currently existing traffic record.

Step2: Formulated our prior probability, so ready to classify a new Packet.

Step 3: Then we calculate the number of points in the packet belonging to each traffic record.

Step 4: Final classification is produced by combining both sources of information, i.e., the prior and to form a posterior probability.

3. CONCLUSION

The paper describes the literature review of ML and DM methods used for cyber. Special emphasis was placed on finding example papers that describe the use of different ML and DM techniques in the cyber domain, both for misuse and anomaly detection. Unfortunately, the methods that are the most effective for cyber applications have not been established; and given the richness and complexity of the methods, it is impossible to make one recommendation for each method, based on the type of attack the system is supposed to detect.

REFERENCES

[1] A. Mukkamala, A. Sung, and A. Abraham, “Cyber security challenges: Designing efficient intrusion detection systems and antivirus tools,” in Enhancing Computer Security with Smart Technology, V. R. Vemuri,Ed. New York, NY, USA: Auerbach, 2005, pp. 125–163.

[2] M. Bhuyan, D. Bhattacharyya, and J. Kalita, “Network anomaly detection: Methods, systems and tools,” IEEE Commun. Surv. Tuts., vol. 16, no. 1, pp. 303–336, First Quart. 2014.

[3] T. T. T. Nguyen and G. Armitage, “A survey of techniques for internet traffic classification using machine learning,” IEEE Commun. Surv. Tuts., vol. 10, no. 4, pp. 56–76, Fourth Quart. 2008.

[4] P. Garcia-Teodoro, J. Diaz-Verdejo, G. Maciá-Fernández, and E. Vázquez, “Anomaly-based network intrusion detection: Techniques, systems and challenges,” Comput. Secur., vol. 28, no. 1, pp. 18–28, 2009.

[5] A. Sperotto, G. Schaffrath, R. Sadre, C. Morariu, A. Pras, and B. Stiller, “An overview of IP flow-based intrusion detection,” IEEE Commun. Surv. Tuts., vol. 12, no. 3, pp. 343–356, Third Quart. 2010.

[6] S. X. Wu and W. Banzhaf, “The use of computational intelligence in intrusion detection systems: A review,” Appl. Soft Comput., vol. 10, no. 1, pp. 1–35, 2010.

[7] Y. Zhang, L. Wenke, and Y.-A. Huang, “Intrusion detection techniques for mobile wireless networks,” Wireless Netw., vol. 9, no. 5, pp. 545–556, 2003.

[8] U. Fayyad, G. Piatetsky-Shapiro, and P. Smyth, “The KDD process for extracting useful knowledge from volumes of data,” Commun. ACM, vol. 39, no. 11, pp. 27–34, 1996.

[9] C. Shearer, “The CRISP-DM model: The new blueprint for data mining,” J. Data Warehouse., vol. 5, pp. 13–22, 2000.

[10] A. Guazzelli, M. Zeller, W. Chen, and G. Williams, “PMML an open standard for sharing models,” R J., vol. 1, no. 1, pp. 60–65, May 2009.

[11] Joo, D., Hong, T., and Han, I. "The neural network models for IDS based on the asymmetric costs of false negative errors and false positive errors." Expert Systems with Applications 25, 69–75 2000.

[12] Lee, W., and Stolfo, S.J... "Data Mining Approaches for Intrusion Detection." Seventh USENIX Security Symposium (SECURITY _98), San Antonio, TX. (1998)

[13] Li, X.B. "A scalable decision tree system and its application in pattern recognition and intrusion detection." Decision Support System 41,112-130. (2005)

[14] Lippmann, R.P., and Cunningham, R.K. "Improving Intrusion Detection Performance Using Keyword Selection and Neural Network." Computer Network 34,597-603. (2000)

[15] Lippmann, R.P., Haines, J.W., Fried, D.J., Korba, J., and Das, K. "The 1999 DARPA off-line intrusion detection evaluation." Computer Networks 34,579-595. (2000)

[16] Liao, Y., and Vemuri, V.R. "Use of K-Nearest Neighbor classifier for intrusion detection." Computer & Security 21,439-448. (2002)

[17] Liu, Y., Chen, K., Liao, X. and Zhang, W. "A genetic clustering method for intrusion detection" Pattern Recognition 37,927-942. (2004)

[18] Mukkamala, S., Sung, A.H., and Abraham, A. "Intrusion detection using an ensemble of intelligent paradigms." Computer Applications 28,167-182. (2005)

[19] Nikulin V "Threshould-based clustering with merging and regularization in application to network intrusion detection." Computational Statistics & Data Analysis. (2005)

[20] Ozyer, T., Alhajj, R., and Barker, K. "Intrusion detection by intelligent boostering genetic fuzzy classfier and data mining criteria for rule pre-processing." Journal of Network and Computer Applications. (2005)