Optimal Data Mining Techniques for Security Means

Pawade Abhijeet S[1] , Pawade Amit S[2] , Mhamane Sanjeev C[3]
NB Navale Singhgad Engg College [2]VVPIET Solapur [3]
abhijeet.s.powade@gmail.com[1] , sanjeev.mhamane4@gmail.com[2]

Abstract— **generally, data mining is called data or knowledge discovery. The process of analyzing data from dissimilar perspectives and summarizing it into serviceable information. Information that can be used to expand revenue, cuts costs or both. Data mining software is one of the analytical tools for analyzing data. It permits users to analyze data from numerous different dimensions or angles, categorize and summarize the relationships identified. Technically, data mining is the process of finding patterns among dozens of fields in large relational databases.**

Keywords: Database mining, Database security, Intrusion detection, Oblivious Decision Trees

I. Introduction

Security is a region that deals with protecting data from intrusions, malwares, frauds and many criminal activities that are rise in digital media with a very fast rate and to maintain non-repudiation, confidentiality and integrity of data. Security is a vital part for securing information of systems and critical infrastructures. With an increased use of computer usage and internet applications, no matter how much the systems and data are secured there are always some vulnerabilities that arise due to the proliferated use of these applications. More recently with the advancements in the field of security, data mining techniques have found their place in this area.

Data mining is extraction of hidden, useful and precious data from large dimensional databases. It was introduced with a goal to support large databases that are used in several business applications for predicting upcoming trends, analyzing data and building proactive decisions. Data mining has emerged as tool that produces its users to recognize the vulnerabilities and guide in providing a defensive tools against various threats to the data systems. The applications of data mining have also increased extremely to many other areas of information security and are not restricted to just intrusion detection and prevention systems.

This paper addresses the various security areas and the application of various data mining techniques in those

Problem Statement:

Most data mining applications run under the assumption that all the data is available at a single central store, called a data warehouse. This presents a huge privacy problem because breaking only a single store's security exposes all the data.
The ultimate aim of this project is to enable a smooth merge of *data mining* and a simple solution to the problem is identify all malicious executable files from the data warehouse and then classifying it with alert message.

areas, which will help researchers to identify and register various data mining techniques to operate with the security issues that arise in the computer systems. The paper focuses on providing a comprehensive study that identifies the various applications of data mining in phrase of security be it the information contained in the systems or establishment of various security policies or security techniques in communication of various systems.

Objectives:

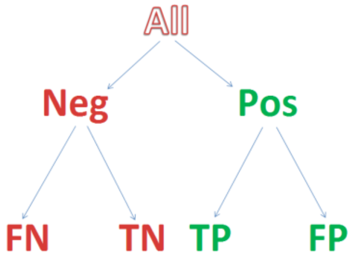
1.Identifying security objectives is an iterative process that is initially driven by an examination of the application's requirements and usage scenarios.
2.Analyse behaviours like call headers, call codes, statistical anomaly and the content of the malicious executable from data warehouse.

Methodologies:

1. The aim of this projects to work and explore a number of standard data mining techniques to determine exact detectors for new unseen executable .We collect a large set of .exe files from public sources and distinguish the problem into two sets: *malicious* and *non-malicious* executable.
2. We split the dataset such that it must be in resulting into two subsets: the *training set* and the *test set*. While generating the rule sets the data mining algorithms use the training set. We use a test set to scan the accuracy of the classifiers over unseen particular example.

3. The framework supports various methods for feature extraction and variant data mining classifiers.
4. To quantitatively indicate the presentation of our method we represent tables with the tally for *true positives* (TP), *true negatives* (TN), *false positives* (FP), and *false negatives* (FN). A true positive is a malicious example that is correctly tagged as malicious, and a true negative is a non-malicious example that is correctly classified. A false positive is a non-malicious example that has been mislabelled by an algorithm as a malicious example, while a false negative is a malicious executable that has been misclassified as a non-malicious example.

   Following is the graphical representation of induction method for better understanding:



5. To estimate the performance, we figure out the false positive rate and the detection rate. The false positive rate is the number of non-malicious examples that are mislabelled as malicious divided by the total number of non-malicious examples. The detection rate is the number of malicious examples that are caught divided by the total number of malicious examples.

   Some of the well-known data mining methods as follow:

1. Decision Tree and Rules.
2. Nonlinear Regression and Classification Methods.
3. Probabilistic Graphical Dependency Models.

From above mentioned methods decision tree and rules is the simplest and easy to understand for users as comparative to others two.
Since it is top to bottom rule indication method we are using this technique for effective results.:

Decision Tree:
A decision tree is a classifier illustrate as a recursive partition of the instance space. The decision tree contains of nodes that form a rooted tree, means it is a directed tree with a node called “root” that has no incoming edges. Each and every node has exactly one incoming edge. A node with outgoing edges is known asinternalor test node. All other nodes are known leaves also known as terminal or decision nodes. In a decision tree, each and every internal node splits the instance space into two or more sub-spaces following to a certain discrete function of the input attributes values.

The region is of noteworthy importance because it permits modeling and cognition extraction from the infinity of data available. Both technicians and programmers are frequently seeking methods to make the procedure more efficient, cost-effective and correct. The decision trees are based on the strategy "divide and conquer"

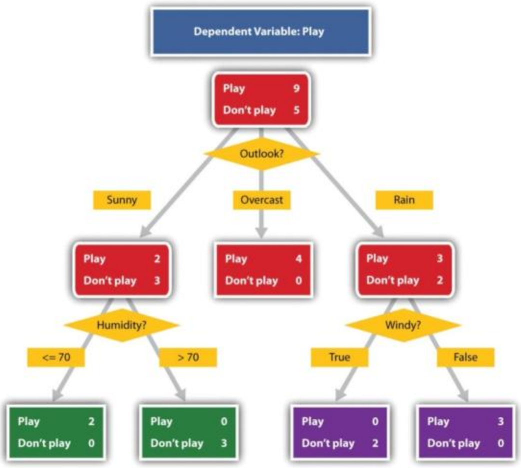
Tree Induction

- Verify how to split the records.
- Prove when to stop splitting.
- Nodes with similar class distribution are favoured.
- How to state the attribute test situation?

E.g:- **Left nodes: -X <=1**

OR

Right nodes: - X>1 Where ‘X’ is a continuous attribute; consider all possible splits and find the best cut.

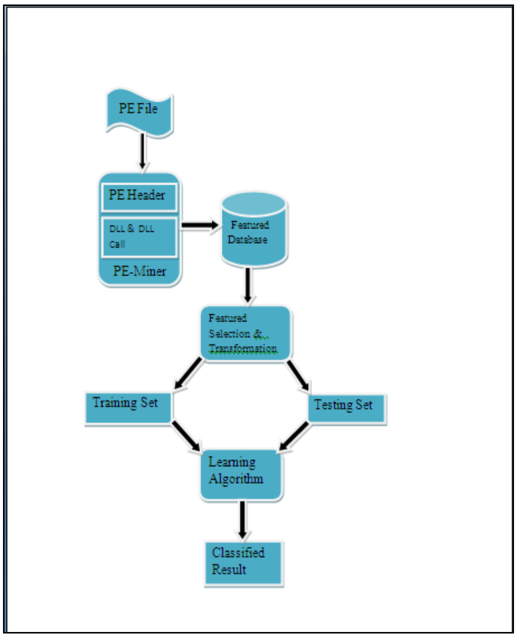


).

System Architecture:

In this paper we explained Multi Naive Bayes algorithm of detecting a malicious executable files. These malicious executable files are formed at the huge amount every year in the typical field and produce a major security threat. The antivirus systems attempt to find these malicious files with strategies designed by hand. This method is often less effective and costly. Hence to overcome this drawback by presenting a strategy in data mining that detects new malicious executable files automatically and more effectively. The data mining platform helps to detects patterns automatically from a huge data set and used these detected patterns to classify a set of new malicious executable files. This new detection technique provides comparison and classification of both the current detection rates for unseen malicious files.

Figure below shows the architecture of system detecting malware. The system mainly involves these of modules Program Executable files (PE-Files), feature selection and transformation lower-cased.

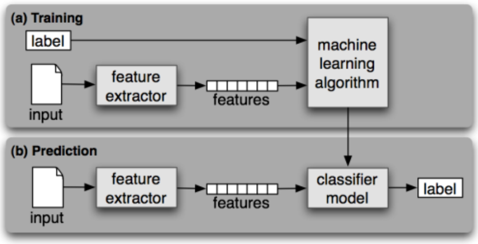


This has extremely strong security features and secrecy problems to user specifically via software of malicious nature that got access to the solution given by the devices and gathers the personal data. So hence some cases supported by best methods designed to provide better security architecture presently in use by these devices.

This Architecture is the undertaking process of using information from old data to examine the output of a specific situation that may occur. Data mining work and focuses to identify the data stored in data warehouses that are being used to store that data that has been examined. The particular data can arrive from all businesses, from the production house to the management. In this work, we are required to make the system on a network of computers for assessing the performance to make it more efficient in terms of time and precise in real world domain. It is needed to make the learning algorithms more efficient in time and space. Presently, the Naive Bayes techniques have to run on a computer with one gigabyte of RAM. Finally we need to plan testing these techniques over a huge set of malicious codes.

Multi Naive Bayes Algorithm:

Multi Naive Bayes Algorithm was basically a group of Naive Bayes algorithm that elect on an altogether classification. Example as each Naive Bayes algorithm classified the samples in the test set as malicious or non-malicious code and this computed as a vote. Further these votes were gathered by the Multi-Naive Bayes algorithm to finalized classification for all the Naive Bayes.



This kind of function was needed because even after using a system with one gigabyte of RAM, to suit into memory the bulk of binary data was too vast. The Naive Bayes algorithm needs a bench ofeach strings or bytes to assess its probabilities. To error free this problem we splitthis into smaller pieces that would suit in memory and instruct a Naive Bayes algorithm over each of the sub problems. We divide the data free from variations into various place by positioning each *i*th line in the binary into the (i mod n)th set where n isthe integer of sets. For each and every set we instruct a Naive Bayes classifier. Our assumption for a binary is the outcome of the projections of the n classifiers.

The multiplication of all the projection of the Naive Bayes classifiers is the projection of the Multi-Naive Bayes algorithm.In certain, we will allot with the next machine learning classifiers, to be specified, Naive Bayes Classifier, Maximum Entropy Classifier and Support Vector Machines. All of these classifiers need training data and hence these methods drop under the class of supervised classification.

Results and Analysis:

We estimate our results over new data by using validation. Validation is the standard procedure to estimate likely predictions over unseen data in Data Mining. For all set of binary profiles we divided the data into 5 equal groups. We used 4 of the subdivision for training and then evaluated the rule set over the remaining divisions. Then we repeated this process for 5 times leaving out a different partition for testing each time. This gave us a very authentic count of our method's accuracy over unseen data. We find median results of these five tests to acquire a good count of how the algorithm performs execute the complete set.

To evaluate our system measures we were interested in several quantities:

1. True Positives (TP), the number of malicious executable examples classified as malicious executables
2. True Negatives (TN), the number of benign programs classified as non-malicious.
3. False Positives (FP), the number of benign programs classified as malicious executables
4. False Negatives (FN), the number of malicious executables classified as benign binaries.

in the false positive rate. This was the percentage of benign strategy which was tagged as malicious, also called false alarms or alerts.

Conclusion:

The very first problem with these traditional anti-malicious executable detection techniques is that in direct to detect a new malicious executable, the program needs to be study and a signature extracted from it and incorporate in the anti-malicious executable software database.

On the basis of detection rate, accuracy, execution time and false alarm rate, the paper has analyzed various classification and clustering data mining techniques for malicious code detection. According to given mandatory parameter ,Multi-Naive Bayes method had the highest accuracy and detection rate of any algorithm over unspecified programs, 97.76%, over double the detection rates Also decision tree has high detection rate in case of huge dataset.

This has the potential to stop some malicious executables in the network and prevents huge database from Data ware house attacks by malicious executables.

References:

[1] Jeffrey O. Kephart and William C. Arnold. Automatic Extraction of Computer Virus Signatures. *4th Virus Bulletin International Conference*, pages 178-184, 1994.

[2] Wenke Lee, Sal Stolfo, and KuiMok. A Data Mining Framework for Building Intrusion Detection Models. *IEEE Symposium on Security and Privacy*, 1999.

[3] Mrutyunjaya Panda and ManasRanjanPatra, “A Comparative Study Of Data Mining Algorithms For Network Intrusion Detection”, First International Conference on Emerging Trends in Engineering and Technology, pp 504-507, IEEE, 2008.

[4] International Journal of Computer Science Trends and Technologyhttp://www.ijcstjournal.org/volume-3/issue-1/IJCST-V3I1P12.pdf, 2015.

[5] International Journal of Advanced Research in Computer and Communication Engineering Vol. 3, Issue 6, June 2014.