IJREAT International Journal of Research in Engineering & Advanced Technology, Volume 4, Issue 2, April - May, 2016
ISSN: 2320 – 8791 (Impact Factor: 2.317)
www.ijreat.org

An Efficient Classification Model Based On Hybrid Approach Using ID3, Association, Normalisation, And Entropy For Intrusion Detection System

Akansha Malviya[1], Dr Amit Shrivastava[2]

1Department of computer science, MTech Scholar Computer science branch, Sagar Institute of Research and Technology, Ayodhya Bypass Road Bhopal, India

[2] Department of computer science, HOD Computer science department, Sagar Institute of Research and Technology, Ayodhya Bypass Road Bhopal, India

Abstract— **malware is kind of software or program that can harm the computer or their normal functioning. As the computational needs is changing their face, in the similar ways the security threads also change their face too. Therefore, the traditional approaches are requires some update, use of traditional approach are suspicious and not able to cover the entire definitions of each kind of malicious code patterns. Therefore an improvement on traditional approach is required to incorporate.**

In this presented work, it is intended to find an adoptive approach by which the machine can learn and update self from the last learning. Therefore adoptive Learning based techniques are investigated. Those are frequently used for the developing the malware detection techniques first so study on decision tree is performed. During study that is found that the classifiers having some limitations such as poor accuracy. Thus a hybrid approach is proposed to improve the classification accuracy. Thus a new improved Normalized Associated ID3 algorithm is introduced and provide a new classification approach. Both the data models are promising and provide the accurate classification.

The implementation of the proposed technique is performed using the JAVA technology and their performance in terms of classification accuracy, time complexity and space complexity is evaluated. According to the obtained outcomes the proposed technique is found accurate and efficient as compared to the respective implemented algorithms.

***Keywords*—data mining, machine learning, classification, malware detection, code pattern analysis**

I. Introduction

Malware sometimes also termed as malicious software or program, which is basically a computer program that is designed to destroy the security of any computers or networks normal functioning. The main motive of these programs can be economic targets or for promotional activities. There are a number of approaches recently developed for the malware detection some of them are signature based and some of them are behaviour based techniques. Most of the Commercial malware detection techniques are based on signature based detection approaches. Most commonly the signatures are sequence of bytes that is always present within a malicious program or in the files that are already infected by malware. To find out the infected files it is required to have a large database of signatures to determine a signature present or not.

Therefore the signature-based methods are not much effective for newly introduced malicious programs additionally that is unable to find the infection over the network or computer. Furthermore the computational complexity of comparing the data base sequences to the computer files is much higher. Thus a new technique is required to identify the behaviour of the infected files from the computers or network. Therefore the behaviour analysis based techniques are developed to identify the malware behaviour.

Basically for analysing the behaviour of the malicious programs the machine learning based approaches are utilized. The main advantage of the machine learning based approaches is to efficient classification of infected files and also these methods are less computationally complex as compared to the signature based techniques.

The machine learning having three key phases of the data analysis namely data pre-processing, learning and classification. During the pre-processing of data is processed in order to prepare the format of data which can be acceptable with the algorithms. In next phase the transformed data is utilized with the mathematical models or data models to identify the meaningful patterns from the data, and in final phases the data patterns are used to classify the targeted patterns on which the computer programs make learning.

In this presented work the behaviour analysis based techniques are investigated for finding the efficient and accurate classification of the malwares. Therefore different machine learning algorithms are evaluated and a new Normalized Associated ID3 approach is introduced for further solution development. This new approach provide the promising results as compared to the similar family of classifiers.

This section provides the basics of the proposed investigation the next section provides a brief detail on the study on the different recently developed approaches which are claimed for producing optimum behavioural analysis of the malicious activities of the malware programs.

II. BACKGROUND

This section includes the different efforts and techniques that are recently used for implementation of malware detection.

Malware stands for malicious software. It is designed with a harmful intent. A malware detector is tries to identify malware using Application Programming Interface (API)or other techniques. API based techniques have two main steps, transformation of malware samples using API call graph, and comparison of the constructed graph against existing malware samples. A major issue of this approach is to collect information about malware. Additionally call graph matching is an NP-complete problem and computational complex. ***Ammar Ahmed E. Elhadi et al [1]*** study, a malware detection system based on API call graph. In this approach, each malware sample is represented as a graph using construction algorithm to transform input malware samples. Moreover, the dependence between different types of nodes is identified and represented using graph edges. After that, graph matching algorithm is used to calculate similarity between the input sample and malware API call graph samples that are stored in a database. The matching algorithm is based on enhanced graph edit distance algorithm that simplifies the computational complexity using a greedy approach from the integrated API call graph with high similarity. Experimental results demonstrate that the given system has 98% accuracy and 0 false positive rates.

The volume of malware is growing every year and creates a serious security issues. Therefore effective malware detection becomes an essential domain in digital security. The signature-based method fails to detect newly introduced malware. ***Igor Santos et al [2]*** propose a method to detect unknown malwares. This model is based on the frequency of the appearance of op-code sequences. Furthermore, they describe a technique for mining the relevance op-code and assess the frequency of each op-code sequence. In addition, the validation of the approach shows that new method is capable to detect unknown malware.

Detection of malicious software or malware using machine learning methods enables fast detection and adaptation of new malwares. The performance of these approaches are depends on the induction algorithms. To benefit from multiple different classifiers, and exploit their strengths ***EitanMenahem et al [3]*** suggest use an ensemble method that will combine the results of the individual classifiers into one final result to achieve overall detection accuracy. Author evaluates several methods using five different base learners (C4.5 Decision Tree, Naïve Bayes, KNN, VFI and OneR) on five different malware datasets. The main goal is to find the best combining method for the task of detecting malicious files in terms of accuracy, AUC and Execution time.

The recently developed systems are works on Detection, Alert and Response (eDare). The key aim is filtering Web traffic of Network Service Providers from malicious code. To obtain the patterns system applies powerful network scanners.These scanners are capable to cleaning traffic from known source. The remaining traffic is monitored through Machine Learning (ML) algorithms to find out unknown malicious patterns.***YuvalElovici et al [4]*** utilize Decision trees, Neural Networks and Bayesian Networks for static code analysis to analyse the malicious code patterns. These algorithms evaluated and results are found promising.

Mobile malware has growing much rapidly. This becomes more effective on Android due to its open platform. Recently, a new generation of Android malware families are introduced with advanced evasion capabilities. These are more difficult to detect using traditional methods. ***Suleiman Y. Yerima et al [5]*** proposes and investigates a parallel machine learning classification approach for detection of such malwares. Using real malware samples, a composite data model is developed from parallel combination of heterogeneous classifiers. The evaluation of data model under different combinations demonstrates its efficacy and detection accuracy. More obviously, by hybrid classifiers with different characteristics, the strengths can be improved.

The increasing complexity of malicious programs needs new techniques that are able to detect the infection, and also able to protect users against new threats. ***AcarTamersoy et al [6]*** present Aesop, a scalable algorithm that identifies malicious executable files by applying Aesop's moral that "a man is known by the company he keeps." they use a large dataset contributed by members of Norton Community Watch. The data set consist of lists of the files that exist on their machines, to obtain relationships between files that frequently appear together. Aesop influences locality-sensitive hashing to measure the asset of inter-file relationships to construct a graph. Aesop attained early labelling of 99% of benign files and 79% of malicious files, over a week before they are labelled by the state-of-the-art techniques, with a 0.9961 true positive rate at flagging malware, at 0.0001 false positive rates.

One of the serious issues on the security on the Internet is malicious programs. The malwares are designed by attackers are polymorphic and metamorphic. Additionally diversity and significant amount of variants some of them are undermining by traditional approaches those uses signature based methods. The behavioural patterns either statically or dynamically obtained can be unable to detect and classify unknown patterns using machine learning techniques. ***EktaGandotra et al [7]*** provide an overview of techniques for analysing and classifying the malwares effectively and accurately.

JinrongBai et al [7] proposed a malware detection approach by mining format information of portable executable files. On keen analysis of static format information of the PE files, extracted 197 features and applied feature selection methods to reduce dimensionality of the features and to achieve high performance outcomes. The selected features are trained with classification algorithms. The results of experiments show accuracy of classification algorithm is 99.1%. The performance of detection scheme and ability of detection of unknown and new malwares make it more promising. Experimental results of identifying new malware are not much accuratebut still able to identify 97.6% of new malwareswith 1.3% false positive rates.

This section provides the study on the recently developed approaches of malware detection system in further the different machine learning algorithms are discussed which are frequently used for behaviour based malware analysis.

III. ALGORITHM STUDY

This section different algorithms are involved which are used for the proposed system development, additionally both the algorithms are also used for the individual classification of malware patterns.

A. Normalized Associated ID3

An Improved ID3 Decision Tree is an algorithm used to generate a decision tree, which is based on Quinlan's earlier ID3 algorithm. The resulting tree is constructed top-down from a fixed set of examples. The leaf node contain the class name whereas a nonleaf node is a decision node. At each decision node, each attribute is tested to decide how good it classifies the examples. The appropriate attribute is then chosen, while the remaining examples are partitioned by it[14] . The attribute selection method is first applied to obtain the importance of each attribute. Then the retrieved Info Gain is combined with attribute importance and thus the highest result obtained for attribute is selected to construct decision tree. One of the method for computing attribute importance is Correlation Function Method(CF) not only overcome the ID3's deficiency which tends towards the value with more number of attributes, but also can represent the relations between attributes and their class attributes. The decision tree generated by this algorithm is used for classification.

CF Algorithm: Suppose D is data set and A is an attribute of D, and C is the class attribute of D. The CF between Attribute and Class can be expressed as follows:

$$CF(A) = \frac{\sum_{i=0}^{n} |\mathrm{xi1} - \mathrm{xi2}|}{n}$$

Where x_{i1} and x_{i2} are the number of attributes of two different classes and n is the total number of unique attributes that A contains. After calculating CF, the normalization of correlation of each attribute is calculated as follows:

CF(1), CF(2), CF(3)....... CF(m), respectively. Thus

$$\mathrm{N(k)} = \frac{CF(k)}{CF(1)+CF(2)+\ldots\ldots\ldots+CF(m)}$$

Where $0 < k \leq m$. Then we calculate Gain for each Attribute as follows:

$$Gain'(A) = E(D) - E(A)*N(A)$$

where *E(D)* is information retrieved by Dataset and *E(A)* is the information retrieved by Attribute *A*.

Gain'(A) can be used as a new attribute selection standard for constructing decision tree according to the procedure applied for ID3 algorithm developed by Ross Quinlan. The attribute with the largest Gain value can be selected as a test value.

A. data pre-processing

Traditionally in this phase the data is processed for cleaning, transforming and enhancing the quality of data for make it more effective to use with the classification algorithms. Therefore a malware data set which contains the malicious and normal op-code patterns are collected from the internet sources. That is a kind of labelled data which can be used with the normal text classification approaches.

But in order to make it more effective the data is pre-processed by the term frequency and a relational attribute based dataset is prepared by using the given samples in dataset. For obtaining the term frequency the following formula is used for effective processing.

$$term\ frequency = \frac{total\ times\ a\ code\ token\ appeared}{total\ amount\ of\ tokens\ avialble}$$

In order to understand the pre-processing we can consider it by an example.

Mov ax, 000h

Add [0ba1fh], c1

Push cs

Add [si+0CD09h], dh

After pre-processing of the data that can be converted in the following set of attributes

Mov	Push	Add
1	1	2

Table 2 example of pre-processing

In further during the computation of term frequency and the above listed attributes are converted as given in table 3.

Mov	Push	Add	Class
0.25	0.25	0.5	Normal

Table 3 example of finalized dataset

In the similar way, the malicious code as given below can also be added in the above pre-processed data table as;

```
Movax, mx
Add [0ba1fh], c1
Add c1, [00hx1b]
Push cs
Add dh, HX
```

Can be a malicious code block thus after pre-processing of data the given table 2 can be viewed as given table 4

Mov	Push	Add
1	1	3

Table 4 example of pre-processing

And in further processing the table 4 data can be demonstrated as given in table 5.

Mov	Push	Add	Class
0.2	0.2	0.6	Malicious

Table 5 example of finalized dataset

B. proposed classification

In order to enhance the accuracy of the traditional data model a probabilistic approach which is frequently used for accurate modelling of data namely proposed classifier is employed. The entire learning process of the proposed data model can be described using figure 1.

The proposed model is described using the figure 1 in this diagram the different components of the proposed data model is defined. The detailed description of each process can be discussed as:

1. Input data: input data is a kind of training samples by which the data modelling is performed. This data can be found in structured or in unstructured format. In this experiment the training data is used in structured format therefore provision for accepting the set of ARFF data or using the data base is supplied for training or learning process. In the same way during the test set generation for cross validation process the data is again selected randomly from the similar formats.

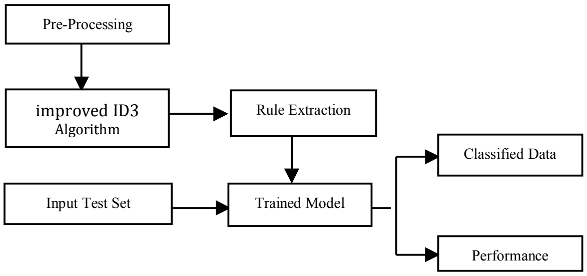

Figure 1 Proposed Data Model

2. Pre-processing: pre-processing in data mining is used when the data available for modelling is not in appropriate or in unstructured format. Using this options the unwanted symbols, undefined instances, noisy contents or incomplete set of data is eliminated from the datasets and the refined and cleaned data can be used to for data modelling or data mining algorithm based learning.

3. Proposed Algorithm: Improved ID3 is based on ID3 algorithm. The pre-processed input dataset is produced in Improved ID3 algorithm in the form of instance data. Instances are the objects of dataset row which are processed using Improved ID3 algorithm and converted to the decision tree as given in figure 2.

4. Rule extraction: Improved ID3 decision tree generates classification rules from the input dataset in the decision tree. This tree mounts entire attributes as nodes and the decisional values as the edges between these nodes. When the data arrives for classification,using the concerned attribute sets comparison or by traversing the nodes through the edges, the decisions are obtained. The traversing process is used as a rule to classify the input samples. For example, the given tree in diagram 2 can be converted to if then else rules.

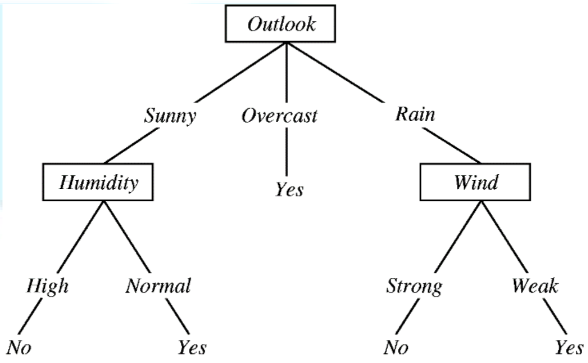

Figure 2 decision tree

This tree rules can be written in the form of the following:

This tree rule is written in the form of the following:
If Outlook = “sunny” and humidity = “high” then
Play tennis = “No”

Else if Outlook = “sunny” and humidity = “normal” then
Play tennis = “Yes”
End if

5. Trained Model: Trained model demonstrates a mathematical relation among the attributes of dataset and difference in their class distribution. Therefore, the trained model has entire information of the training samples in mathematical term. These terms are used to recognise a pattern, how it falls on a specified category. If the computations are not accurate during data modeling, the performance of model becomes poor.

6. Test set input: The test set is used for evaluating the developed data model from the input training sets. In N-foldcross-validation process, the trained data is sampled randomly for constructing the random instances of input dataset. These randomly selected data patterns are used with the trained classifiers to compute their target class labels. During prediction, if the predicted outcome is equal to the defined class label, then accuracy is increased otherwise the error rate is increased.

8. Classified data: The test set is applied on the trained classifier and evaluated using the prepared probabilistic model. The predicted class labels are those labels which are nearest to the input pattern. Therefore, the whole input instances available in test set are recognized through the trained classifier.

9. Performance: that is final stage of the system data processing where the N-cross validation technique is applied on data to find the accuracy of the prepared model. In addition of the accuracy the other performance parameters such as error rate, memory consumption, training time and search time is also reported during model cross validation.

This section provides the entire solution development and the next section provides the detailed study about the algorithm designed to represent working steps.

IV. ALGORITHM SIMULATION

The given section provides the functional steps of the designed data model as:
Input: Dataset D, Test Dataset T
Output: Decision Tree
ALGORITHM STEPS:

```
1. $A = [\text{column, rows}]$
2. $A = \text{ReadDataset}(D)$
3. $\text{Pa} = \text{PreProcess}(A)$
4. $[\text{model, validation}] = \text{Train\_Improved\_ID3}(\text{Pa, num\_folds})$
5. $T_{\text{rule}} = \text{Extract}_{\text{Rules(model)}}$
6. $[R_{\text{rule}}, \text{Rindex}] = \text{ValidateRule}(T_{\text{rule}}, \text{model})$
7. $\text{Ref}_{\text{rule}} = \text{PrunRules}(R_{\text{rule}}, \text{Rindex})$
8. $[\text{Accuracy, ErrorRate, Decision}] = R_{\text{rule}}\text{Classify}(T)$
```

The entire process of the proposed data model is summerised in the above steps. In first step, the data set D is given as input to the system. The next step includes a variable named A which is declared with the similar size of rows and columns as the input dataset. Function $ReadDataset(D)$ is used to read the data row and column wise and the extracted contents of data is stored into the previously defined matrix A. Further the data is pre-processed using function $PreProcess(A)$. The pre-processing function accepts the matrix data A and pre-process the data as defined in section 4. After pre-processing of data, it is stored in a variablePa. Next the decision tree algorithm is employed over the pre-processed data for making training rules from the input pre-processed data. Function $Train\_Improved\_ID3(Pa, num\_folds)$ is implemented that accepts the pre-processed data and the number of folds as arguments to validate the training process of the system. After successful training, this function returns the model (decision tree data model) that incorporates the learned patterns over the tree. $Extract_{Rules(model)}$Function is used to extract the rules form the decision tree. In the next step, the rule validation process and pruning process are taken place to optimize the performance of classifier by reducing the unwanted rules from the developed decision tree. $ValidateRule(T_{rule}, B_{model})$Function is used to validate rules using the trained Bayes model,$B_{model}$and the rules extracted from the decision tree, T_{rule} . After that the $PrunRules(R_{rule}, Rindex)$ function is used with poor or ambiguous rules having arguments index,$Rindex$ and the list of appropriate rules, R_{rule}. Finally $R_{rule}Classify$ function is called to classify the new input testing set that returnsAccuracy, Error rate and the Decisions of the input data.

V. RESULTS ANALYSIS

This section provides the performance analysis and results of the proposed malware detection system. Additionally in order to justify the proposed approach the comparative analysis among two additionally implemented algorithms is also reported in this section.

A. Accuracy

In data mining the classification accuracy is the amount of data instances that accurately identified using the trained algorithm or data model. That can be calculated by the given formula

$$Accuracy = \frac{Total\ correctly\ classified\ samples}{Total\ input\ samples\ to\ classify}$$

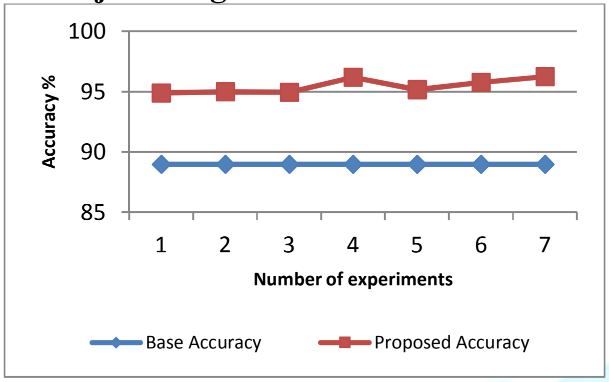

Figure 3**accuracy**

The figure 3 shows the comparative performance of Base Paper algorithm and proposed ID3 algorithm. In order to represent their performance the X axis contains the different experiments performed with the system and the Y axis denotes the percentage accuracy achieved during experiments. According to the obtained results, Bayesian classifier and Base algorithm provides less accurate results as compared to the proposed ID3 algorithm, thus after new approach implementation the performance of decision tree is improved significantly.

A. *Error rate*

The error rate demonstrates the amount of data which is not accurately classified using the trained data model. The comparative error rate of the implemented simulation is given using figure 4.

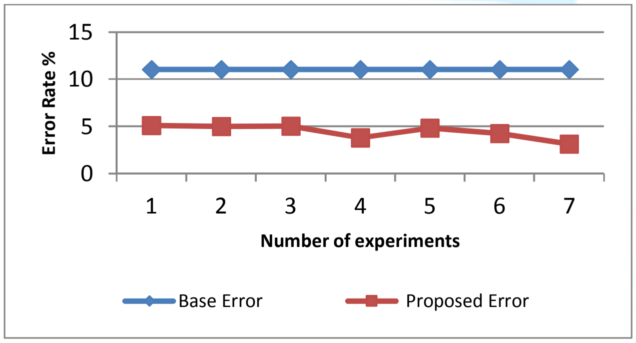

Figure 4 Error Rate

The given figure 4 contains the error rate curve of the proposed learning algorithm in order to compare performance of classifiers the X axis contains the number of experiments performed and the Y axis shows the Error rate percentage. According to obtained results the proposed classifier's error rate is fewer as compared to traditionally implemented classifiers. Thus the proposed method is much adoptable as compared to traditional classification techniques.

B. *Memory Used*

The amount of main memory required to evaluate the input data using the selected algorithm is known as the memory utilizationor space complexity. The comparative memory consumption of the proposed ID3, and base paper algorithms are given using figure 5.

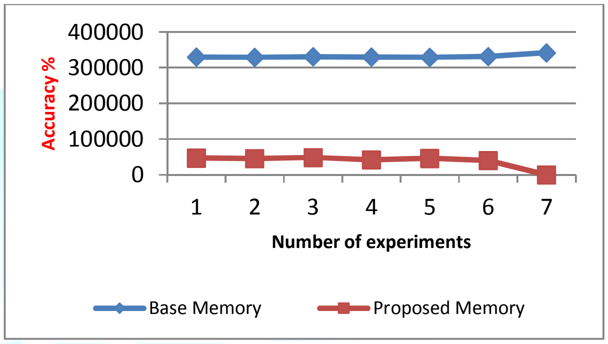

Figure 5 memory consumption

The memory consumption of implemented algorithm is given using figure 5. According to the obtained results if the amount of data for training is increases then the memory consumption of the systems are also increases therefore that is depends upon the amount of data produced for learning. As given in above diagram the amount of base paper classification algorithm's memory consumption is more than the proposed algorithm. The key reason behind this is it stores more data in the main memory as compared to the new algorithms.

C. *Time*

The amount of time required to evaluate the input training samples for learning with data models is known as the training time. The comparative training time of the proposed and traditional algorithms is given in figure 6.

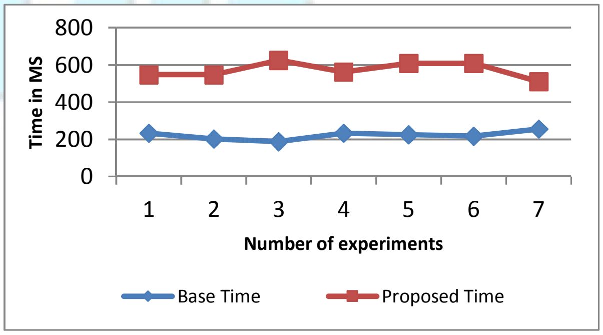

Figure 6 Time

According to the obtained results, the time of the proposed system is higher as compared to the base paper approaches.

Thus, the performance of the algorithm is not good in terms of time.

According to the evaluated outcomes the performance of the proposed classification algorithm is much efficient as compared to the traditionally available algorithm. Because the less amount of tree branches are generated after the pruning process thus the evaluation of data model becomes efficient as compared to the traditional approaches.

This section provides the results analysis of the proposed algorithm with respect to the other similar machine learning algorithms. According to the obtained performance the proposed technique is found the optimum results.

VI. Conclusion

The proposed work is intended to investigate the effective and accurate technique for malware detection. There are two key techniques are available for malware detection namely signature based or behavior based. During investigation that it is found that the signature based techniques are accurately detect the malicious patterns. But these techniques are having their own complexities such as slow processing or high time complexity and high space complexity due to storage of the malicious patterns on the database. In addition of that new patterns of the malicious patterns are not recognized. Therefore the behavior based malicious patterns detection techniques are provide the effective and efficient solution.

Therefore in this work the behavior based technique is adopted for further investigation and the machine learning techniques are used for solution development. Thus the key aim is to enhance the performance of the traditional system in terms of the space and time complexity during the detection process and also can make adoptive system which can adopt new arrived patterns of the malicious codes.

Therefore the proposed algorithm based on traditional ID3 decision tree algorithm is designed and implemented using the JAVA technology and the performance is tested over the 783 malicious and normal op-code samples. According to the obtained performance the proposed algorithm is accurate and efficient for classification but that is lacked on the time consumption as compared to the base paper algorithm. According to the different experimental outcome that reflects the accuracy of classification approximately 97-100%. Also compare and write error%, Memory used and time and compare it base paper

Future work:

The proposed classification technique is an effective and promising approach for malicious pattern detection. In near future the proposed concept can be extended with the a storage implementation that can preserve the classifier tree for further use and adopting new rules from the big data environment using block decomposition concept.

References

[1] Ammar Ahmed E. Elhadi, MohdAizainiMaarof, Bazara I.A. Barry, HentabliHamza, “Enhancing the detection of metamorphic malware using call graphs”, © 2014 Elsevier Ltd. All rights reserved.

[2] Igor Santos, Felix Brezo, XabierUgarte-Pedrero, Pablo G. Bringas, “Opcode sequences as representation of executables for data-mining-based unknown malware detection”, 2011 Elsevier Inc. All rights reserved.

[3] EitanMenahem, AsafShabtai, LiorRokach, and Yuval Elovici, “Improving Malware Detection by Applying Multi-Inducer Ensemble”, Copyright © 2008 Elsevier B.V. All rights reserved.

[4] Yuval Elovici, AsafShabtai, Robert Moskovitch, Gil Tahan, and ChananGlezer, “Applying Machine Learning Techniques for Detection of Malicious Code in Network Traffic”, © Springer-Verlag Berlin Heidelberg 2007

[5] Suleiman Y. Yerima, SakirSezer, Igor Muttik, “Android Malware Detection Using Parallel Machine Learning Classifiers”, 8 th International Conference on Next Generation Mobile Applications, Services and Technologies, (NGMAST 2014), 10-14 Sept., 2014.

[6] AcarTamersoy, Kevin Roundy, DuenHorngChau, “Guilt by Association: Large Scale Malware Detection by Mining File-relation Graphs”, KDD’14, August 24–27, 2014, New York, NY, USA. Copyright 2014 ACM 978-1-4503-2956-9/14/08

[7] EktaGandotra, DivyaBansal, SanjeevSofat, “Malware Analysis and Classification: A Survey”, journal of Information Security, Vol.5 No.2(2014), Article ID:44440,9 pages

[8] JinrongBai, Junfeng Wang, and GuozhongZou, “A Malware Detection Scheme Based on Mining Format Information”, Hindawi Publishing Corporation Scientific World Journal, Volume 2014, Article ID 260905, 11 pages

[9] AashooBais, KavitaDeshmukh, Manish Shrivastava, “Implementation of Decision Tree”, International Journal of Engineering and Advanced Technology (IJEAT) ISSN: 2249 – 8958, Volume-2, Issue-2, December 2012

[10] RoshaniChoudhary, JagdishRaikwal, “An Ensemble Approach to Enhance Performance of Webpage Classification”, International Journal of Computer Science and Information Technologies, Vol. 5 (4) , 2014, 5614-5619.

[11] Adel Sabry EESA, Zeynep ORMAN, Adnan Mohsin Abdulazeez BRIFCANI, "A new feature selection model based on ID3 and bees alorithm for intrusion detection system" Turk J Elec Eng & Comp Sci(2015) 23: 615-622