International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 07 Issue: 07 | July - 2023 SJIF Rating: 8.176 ISSN: 2582-3930

SEDM Droid An Enhanced Stacking Ensemble Framework for Android Malware Detection

P Hareesh[1], Prof Seema Nagaraj[2]

Dept of MCA, Bangalore Institute of Technology, Bangalore, India[1]

Dept of MCA, Asst Professor, Bangalore Institute of Technology, Bangalore, India[2]

ABSTRACT Suggest a architecture that enables the use of various machine learning algorithms to effectively differentiate between malware files and clean files while attempting to reduce the amount of false positives. In this research, we first work with cascade one-sided perceptrons and then with cascade kernelized one-sided perceptrons to demonstrate the concepts underlying our system. The concepts underpinning this framework were tested successfully on medium-sized datasets of malicious and clean files before being put to a scaling-up Process that enables us to work with very big datasets of malicious and clean files.

INTRODUCTION

In is defined as programme intended to access or harm A set for computers with owner's consent in- formed consent. Malware is a generic definition for all kind of computer threats. A simple classification of malware consists of file infectors and stand-alone malware. Another way of classifying malware is based on their particular action: Trojans, worms, backdoors, rootkits, spyware, adware etc. Malware detection through standard, signature-based methods [1] is getting more and more difficult since all current malware applications tend to have multiple polymorphic layers to evade discovery or to use side mechanisms to automatically update themselves to a newer version at short periods of time to avoid being noticed by any antivirus software. For an example of dynamical file examination for malware detection, via emulation in a virtual environment, the interested reader can see [2]. Classical methods for the detection of metamorphic viruses are described in [3]. An overview on many machine learning techniques that were [4] provides a recommended method for malware detection. Here, we provide a few examples of these techniques in use. - In [5], it is discovered that boosted decision trees using n-gram data outperform both the Naive Bayes classifier and Support Vector Machines in terms of output. To be able to differentiate between malicious and legitimate programme files, [6] employs automated extraction of association rules on Windows API execution sequences. [7] uses association rules as well, but only with honeytokens with well-known properties. - To determine if a specific programme file is (or is not) a variation of a prior programme file, In [8], Hidden Markov Models are used. In the past, profile hidden Markov models have utilised quite successfully for sequence analysis in bioinformatics, are employed by [9] to accomplish a similar task. - The amount using neural networks to find polymorphic malware is explored in [10].

II. LITERATURE SURVEY:

Abdul Raoof[1] The study is implemented a system for file encryption that is based on AES and has an effective communication security mechanism. This security architecture is simple to implement on PaaS, IaaS, and SaaS, and one-time passwords add an extra layer of protection when users are authenticated.

Advantages:

- Performance time and accuracy

Disadvantages:

- Training model prediction on Time is High
- It is based on Low Accuracy

Nitin Pandey[2]

Numerous organisations have chosen the cloud computing environment, therefore the quick move to the mist has fueled worries from a security standpoint. The adoption of cloud computing has led to a diversity of concerns and difficulties. In instruction to help cloud service providers and customers address security concerns in utilizing cloud computing, this article highlight those concerns. In order to address these possible dangers, This essay will initially pinpoint the security needs for the cloud and then try to propose a workable solution.

Faisal Hussain [3]

In this study, we offered a tough problem that is tracking traffic. which requires efficient ways to detect every deviation from the normal behaviour on computer networks. In this study, we offer two models based on Firefly Algorithm and Genetic Algorithm to identify network abnormality utilising flow data, such as bits and packets per second. Both outcomes were assessed to determine their ability identify networks anomalies, and results were then compared. We experienced good results using data collected at the backbone of a university.

- Suresh M[5] The proposed solutions are this ensures fine-grained detection of various attacks. Three deep learning replicas have been used to compare the proposed framework to the current models real datasets (a new dataset NBC, a combination of UNSW-NB15 and CICIDS2017 consisting of 101 classes).

Advantages:

It performs accurate classification of health state in comparison with other methods

Disadvantages:

It is low in efficiency.

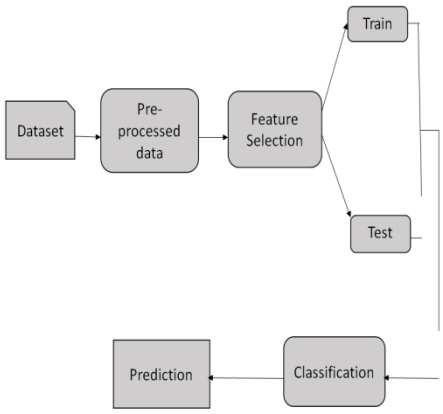

Fig. 1. Proposed Architecture

III. EXISTING MODEL

In existing system, The malware files in The viruses from Heaven repository provided the training information. The test set of data contains malware files from the WildList collection and clean files from different operating systems (other files that the ones usedin the first database). The malware collection in the training and test datasets consists of trojans, backdoors, hacktools, rootkits, worms and other types of malware. The first and third columns in Table II represent the percentage of those malware types from the total number of files of the training and respectively test datasets. The second column in Table II represents the corresponding percentage of malware unique combinations from the total number of unique combinations value features for the practice data

Disadvantages

- Doesn't Efficient for handling large volume of data.
- Theoretical Limits
- Incorrect Classification Results.
- Less Prediction Accuracy.

IV. PROPOSED METHODOLOGY

The Suggested model is presented to eliminate all the drawbacks of the present system. By categorizing the data grounded on the software quality prediction dataset and others using SVM, Gradient Boosting, Navie Bayes Random Forest, and decision Tree techniques, this system will advance the precision of the classification findings. The performance of the overall classification results is improved.

ADVANTAGES

- High performance.
- Provide accurate prediction results.
- It avoid sparsity problems.
- Reduces the information Loss and the bias of the inference due to the multiple estimates.

V. IMPLIMENTATION

DATA SELECTION AND LOADING

- Data selection is the process of determining the appropriate data type and source, in addition suitable instruments to collect data.
- Data selection precedes the actual practice of data collection and it is the procedure where data relevant to the analysis is decided and retrieved from the data collection.

• The Malware dataset is utilised in this research for the detection and prediction of Malware kind.

DATA PREPROCESSING

• There may be a lot of missing data and worthless information. To manage this part, data purification is performed. It involves coping with inaccurate data, noisy data, etc.

• Missing Data: This problematic occurs when the data has gaps in it. There are several ways to handle it.

- Ignore the tuples: This approach only the whole thing when our dataset is sizable and a tuple has numerous missing items.
- There are several methods to complete this work, so fill in the missing data. You can opt to manually fill in the missing values, use the attribute mean, or use the value that is most likely.
- Encoding Categorical Data: Variables having a limited number of label values are referred to as categorical data. that numerous machine learning techniques want numerical input and output variables. categorical data is converted to integer data using one integer and one hot encoding.
- Scikit-learn's Count VectorizerA gathering of text documents may be converted into a vector of term/token counts using the vectorizer programme. Additionally, it makes it possible to pre-process text data before creating the vector illustration. It is a very flexible feature representation module for text because of this functionality.

SPLITTING DATASET INTO TRAIN AND TEST DATA

- Data splitting, which is frequently done for cross-validator reasons, is the process of dividing accessible data into two pieces.
- A predictive model is created using a portion of the data, and its effectiveness is assessed using a different portion of the data.
- A crucial step in analysing data mining algorithms is dividing the data into training and testing sets.

VI. CONCLUSIONS

We examined several significant machine learning-based malware prediction systems. Designing IDS with high prediction rates and low false positive rates while the system swiftly adapts itself is made possible by ML approach characteristics. Three different ML-based classifiers were created using these algorithms: Random Forest (RF), Support Vector Machine (SVM), and Decision Tree (DT). Despite the similarities between these two methods, they are different in a number of ways that meet the criteria for creating effective software quality prediction, including adaptability, high computing speed, and error resilience in the face of noisy information.

VII. REFERENCES

- A. A. Khan, M. H. Rehmani, and M. Reisslein, ‘‘Cognitive radio for smart grids: Survey of architectures, spectrum sensing mechanisms, and networking protocols,’’ IEEE Commun. Surveys Tuts., vol. 18, no. 1, pp. 860–898, 1st Quart., 2016.
- Y. Lin, X. Zhu, Z. Zheng, Z. Dou, and R. Zhou, ‘‘The individual identification method of wireless device based on dimensionality reduction,’’ J. Supercomput., vol. 75, no. 6, pp. 3010–3027, Jun. 2019.
- T. Liu, Y. Guan, and Y. Lin, ‘‘Research on modulation recognition with ensemble learning,’’ EURASIP J. Wireless Commun. Netw., vol. 2017, no. 1, p. 179, 2017.
- Y. Tu, Y. Lin, J. Wang, and J.-U. Kim, ‘‘Semi-supervised learning with generative adversarial networks on digital signal modulation classification,’’ Comput. Mater. Continua, vol. 55, no. 2, pp. 243–254, 2018.
- C. Shi, Z. Dou, Y. Lin, and W. Li, ‘‘Dynamic threshold-setting for RFpowered cognitive radio networks in non-Gaussian noise,’’ EURASIP J. Wireless Commun. Netw., vol. 2017, no. 1, p. 192, Nov. 2017.
- Z. Zhang, X. Guo, and Y. Lin, ‘‘Trust management method of D2D communiqué based on RF fingerprint identification,’’ IEEE Access, vol. 6, pp. 66082–66087, 2018.
- H. Wang, J. Li, L. Guo, Z. Dou, Y. Lin, and R. Zhou, ‘‘cFractal complexitybased feature extraction algorithm of communication signals,’’ Fractals, vol. 25, no. 4, pp. 1740008-1–1740008-3, Jun. 2017.
- J. Zhang, S. Chen, X. Mu, and L. Hanzo, ‘‘Evolutionary-algorithm-assisted joint channel estimation and turbo multiuser detection/decoding for OFDM/SDMA,’’ IEEE Trans. Veh. Technol., vol. 63, no. 3, pp. 1204–1222, Mar. 2014.