A Course Project Report on

Team ID & Title:
CS2021W1042

Ransomware Detection using ML Algorithms

Submitted
as part of **CSE4003-Cyber Security**
by


19BCE2399
Saksham Jain


19BCE0455
Saurav Ranjan


19BCE0466
Saksham Minocha


19BCE0912
Sayan Jana


19BCE0784
Shyam Ranjan Bharti

To


Dr M Rajasekhara Babu

School of Computer Science and Engineering

VIT®
Vellore Institute of Technology
(Deemed to be University under section 3 of UGC Act, 1956)

April 2022

Index

Abstract

Ransomware is a type of sophisticated virus that uses encryption to prohibit users from accessing their data unless a ransom is paid to the attacker. The threat presented by ransomware is quite severe, with new varieties and families being discovered on the internet and dark web on a regular basis. Given the nature of the encryption algorithms utilised by ransomware, recovering from an infection is challenging. The rise in ransomware is also accompanied by an increase in the usage of artificial intelligence. Machine learning and deep learning techniques to detecting ransomware are attracting a lot of attention since machine learning and deep learning can identify zero-day threats. Traditional machine learning algorithms are skewed toward the more commonly occurring class categories, failing to capture the ransomware's overall pattern or structure. They have a tendency to overfit to the data presented, resulting in poor performance on real-world data examples with uncertain class category labels. As a result, we conclude that typical machine learning techniques are unsuitable for such a significant data security challenge. Deep Neural Networks have been shown to function effectively with time series data, as well as multi-class classification and clustering issues, and they capture varying levels of granularity of the underlying structure in the data set at various layers of the model architecture.

***Keywords*:** Ransomware, machine learning, Deep Neural Networks, Deep Learning, Scaling, Receiver Operating Characteristic, Confusion matrix, n-gram, feature extraction, Bitcoin transactions, Encryption, Gray-scale image

Chapter 1
INTRODUCTION

IMPORTANCE OF IDEA:

- Ransomware is a sort of malware that aims to block or limit a user's access to their device, operating system, or files. Locker ransomware and crypto-ransomware are the most common types of ransomware. Locker ransomware creates a lock screen that keeps victims from accessing their computers, typically impersonating law authorities and demanding money in exchange for access. Crypto-ransomware encrypts critical files on a user's system with advanced encryption algorithms and demands decryption payments, generally in the form of bitcoin. Ransomware has grown in popularity, sophistication, and destructiveness over time. Since its inception in 1989, several distinct variables have been blamed for the emergence of ransomware.

- Machine learning has proved successful in identifying malware on both Windows and Android systems. More machine learning research on malware detection as a replacement for signatures, demonstrating the superiority of machine learning-based detection over signature-based detection. Because of their versatility and great capacity to detect unknown strains of ransomware virus, researchers chose to investigate machine learning and deep learning techniques above other non-machine learning-based approaches.

- Deep learning and machine learning have an impact on every part of life. Because of their decision-making capabilities, these technologies have a wide range of applications in every discipline. Because of the employment of these possible technologies, advanced assaults and threat detection got easier in less time. Deep learning is the most effective method for detecting the patterns of a working system. It has a wide range of applications because to its pattern recognition capabilities. Machine learning and deep learning are burgeoning technologies that are widely employed in advanced cybersecurity research. These technologies should also be applied to the detection of ransomware. In terms of pattern recognition, both systems were useful. Machine learning and deep learning are used to detect a variety of viruses and ransomware.

- In this article, we tested various machine learning algorithms for detecting ransomware addresses and discovered that the neural network model outperforms other models in measures such as Receiver Operating Characteristic (ROC), F1 score, and accuracy. The information granularity of a neural network is captured at several layers, allowing it to recognise ransomware addresses in a broader manner.

CHAPTER 2

PROBLEM STATEMENT & OBJECTIVES

2.1. Problem Statement:

Ransomware is a notorious piece of software that has gained notoriety due to its fatal and irreversible consequences on its victims. Crypto ransomware is a sort of ransomware that uses an encryption method to prevent access to its victim's data. Even if the victim is able to delete the ransomware from the infected file, the encrypted file remains permanently barred. The irreversible damage inflicted by ransomware requires early identification of these assaults. Thus we use machine learning algorithms, to detect this ransomware. In terms of parameters like Receiver Operating Characteristic (ROC), F1 score, and accuracy, we suggest a neural network model that performs somewhat better than existing models.

2.2. Objectives:

1 Neural networks capture the information granularity at various layers which helps to detect ransomware addresses in a more generalized way.

2 We propose a neural network model that performs slightly better compared to other models in metrics like Receiver Operating Characteristic (ROC), F1 score and accuracy.

3 This model is a part of Full Antivirus + Ransomware Protection Software. It's a classification problem (Supervised Machine Learning). The data was imbalanced and needed to be transformed (Synthetic Samples: SMOTE-Tomek).

4 We will generalize enough to classify Bitcoin addresses belonging to different Malware families and the test results need to be validated on more recent Bitcoin addresses.

CHAPTER 3

LITERATURE REVIEW

[1] Bae, S. I., Lee, G. B., & Im, E. G. (2020). Ransomware detection using machine learning algorithms. Concurrency and Computation: Practice and Experience, 32(18), e5422.

This paper proposes a ransomware detection method that can distinguish between ransomware and benign files as well as between ransomware and malware. They extracted Windows API (Application Programming Interface) invocation sequences using the Intel PIN tool,8 and n-gram sets were generated from the extracted API sequences. Then they proposed CF-NCF (Class Frequency - Non-Class Frequency), and feature vectors were generated using CF-NCF and six machine learning algorithms were tested to detect ransomware. The experimental results show that their proposed method can detect ransomware among malware and benign files with the ransomware detection accuracy up to 98.65%.

Methodology:

This paper proposes a ransomware detection method based on machine learning algorithms. For each sample, extracted API sequence are processed and n-gram sequences are generated. From the n-gram sequences, input vectors are generated. Each element of the input vector can be represented as 1 if an n-gram appears in the n-gram sequence or as 0 if it does not appear. CF-NCF (Class Frequency Non-Class Frequency) values of each element of the generated vector are calculated, and these values are used as weights of each element. This detection model based on six machine learning algorithms is used to classify unknown binary samples into ransomware, malware, or benign files

Algorithms used:

- Random Forest (RF)

Random Forest is a type of ensemble learning methods for classification and regression analysis and generates output using multiple decision trees that are constructed through training processes.

- Logistic Regression (LR)

Logistic Regression is a statistic method to predict probabilities of events using linear combination of independent variables.

- Näive Bayes (NB)

Näive Bayes is a simple classification algorithm based on Bayesian theorem. An element can be classified into a class among multiple classes.

- Stochastic Gradient Descent (SGD)

SGD repeatedly updates parameters for each training data to reach global optimization so that losses between real values and predicted values can be minimized.

- K-Nearest Neighbors (KNN)

KNN is a machine learning algorithm used for classification and regression in pattern recognition. KNN determines which class an element belongs to by calculating K nearest elements in a trained data set.

- Support Vector Machine (SVM)

SVM is a supervised learning model widely used for pattern recognition or data analysis. SVM generates non probabilistic binary linear classification model that determines which category new data belongs to.

[2] Fernando, D. W., Komninos, N., & Chen, T. (2020). A study on the evolution of ransomware detection using machine learning and deep learning techniques. IoT, 1(2), 551-604.

This paper investigates the contributions of research into the detection of ransomware malware using machine learning and deep learning algorithms. The main motivations for this study are the destructive nature of ransomware, the difficulty of reversing a ransomware infection, and how important it is to detect it before infecting a system.

They reviewed research which uses machine learning and deep learning for the detection of ransomware. In general, the approaches reviewed boast high detection rates in the mid to high 90s. These models are all trained on a mix of network, behavioural, or static features. While most are conceptual systems, like RansomWall and the EldeRan system, some have been tested via deployment. The results achieved give us confidence that machine and deep learning models can be deployed to detect ransomware

Methodology:

In this paper, they did the following:

• Review of Research: Review available machine learning and deep learning approaches to detecting ransomware. They evaluate the weaknesses of each approach and how improvements can be made in the future.

• Evaluate Research: They evaluate each study’s strengths, weaknesses, and how they can be improved.

• Longevity Evaluation Experiments: They evaluate the longevity of these approaches by running our experiments on current generation and older generation ransomware. Their experiments introduce concept drift to the approaches we reviewed and observe their accuracy under concept drift.

Algorithm Used:

- Random Forest
- Logistic Regression
- SVM
- J48 Decision Tree
- GTB

[3] Poudyal, S., Subedi, K. P., & Dasgupta, D. (2018, November). A framework for analyzing ransomware using machine learning. In 2018 IEEE symposium series on computational intelligence (SSCI) (pp. 1692-1699). IEEE.

In this paper, they proposed a ransomware detection framework leveraging techniques of reverse engineering, static analysis, and machine learning.They built the feature database from analysis at assembly and dll level of binaries and trained and tested with our machine learning training model. Their experimental results show that among eight supervised machine learning classifiers our framework could achieve an accuracy of more than 90% (96.5% on average) for seven of them for combined feature dataset and assembly level instruction dataset. They claim that static analysis of binaries at assembly and dll level is crucial to build distinguishing characteristics for ransomware detection using machine learning.

Methodology:

In this technique, a malware or an untrusted program is executed in a safe artificial environment so that there is no real harm to the system

• They designed an automatic static analysis framework that leveraged the assembly instructions and dlls to improve the detection accuracy of ransomware.

• They performed the reverse engineering of ransomware and normal binaries to get the code for different levels. These codes were analyzed using static analysis and its output is used to build the model using machine learning.

• The average ransomware detection accuracy was 92.11% while considering both individual and combined feature dataset for all given supervised machine learning classifiers.

Algorithm Used:

- Logistic Regression
- Random Forest

- SMO with LK
- SMO with PK
- AdaboostM1 with J48

[4] J. A. H. Silva, L. I. B. López, Á. L. V. Caraguay, and M. Hernández-Álvarez, ‘‘A survey on situational awareness of ransomware attacks—Detection and prevention parameters,’’ Remote Sens., vol. 11, no. 10, p. 1168, May 2019.

This paper proposes a ransomware analysis and identification framework based on the runtime behavior of ransomware and deep learning based semi-supervised technique. Deep learning is a robust unsupervised approach which can extract the hidden intrinsic patterns from unsupervised feature space through a non-linear transformation and layered structure in which upper layers compute more abstract forms of features presenting the latent sources of variabilities in the feature space. Their proposed approach is that deep learning based semi-supervised technique can extract dynamics of behavioral patterns from the new variants of ransomware obtained from the wild and can integrate the latent sources to the supervised classifier, making the detection engine independent of manual signature generation and robust to the changes.

The contributions of this paper are listed below:

1) To develop an anti-obfuscation model of feature extraction using dynamic analysis.

2) To develop an adaptive detection engine using a deep learning based semi-supervised model.

Methodology:

This paper proposes that the model can extract the attack patterns of the ransomware through the deep learning-based semi-supervised method, ransomware from 14 families with a large number of features have been considered and a novel feature extraction procedure has been developed. Since the model can learn the frequently changing behavior of the ransomware and apply this knowledge to detect them, it ensures zero-day detection. The extracted features are integrated into the supervised detection engine to build an adaptive model. Their experimental results demonstrate that the proposed model achieves significant performance improvement over supervised detection approaches and achieves more than 95% accuracy.

Algorithm used:

- Global Feature Set Generation
- Feature Vector Generation
- Varying the Node Arrangements Using Deep Learning Based Model

[5] Kok, S. H., Azween, A., & Jhanjhi, N. Z. (2020). Evaluation metric for crypto-ransomware detection using machine learning. Journal of Information Security and Applications, 55, 102646.

This study initially contributed to the use of application programme interface (API) as the data for analysis before encryption. The creation of a pre-encryption detection algorithm (PEDA) with two layers of detection to increase overall detection performance and accuracy is the second contribution. The final contribution is to suggest six additional indicators that may be used to gain a better understanding of the LA's capabilities.

Methodology:

The authors of this study gathered crypto-ransomware samples from three different sources: Old, VirusTotal, and theZoo. VirusTotal and theZoo are two online-based libraries of malware samples open to the public, and the Old source contains samples utilised by them. The Cuckoo Sandbox analysis system was used to dynamically analyse the acquired samples, capturing all API queries in each sample. This data was retrieved from the PEDA and transformed to a dataset format for machine learning training and testing of the LA. The proposed measures were then used to analyse the test outcomes.

Algorithms:

- Pre-encryption detection algorithm (PEDA)
- Support Vector Machine (SVM)
- Random Forest (RF)

3.1. Gaps identified in existing literature

Paper 1: It doesn't contain, compile IoT ransomware to test the applicability of the detection methods which they explored in this paper. It acknowledges the scarcity of ransomware in IoT currently; therefore, it is impossible to evaluate the current ransomware detection approaches in an IoT context. The viewpoint that ransomware will not target IoT devices and instead attack their back-end servers could well come to fruition, so we must take into account aspects of IoT back-end architecture that might be vulnerable to ransomware. If the IoT ransomware threat does not emerge immediately, we need to create or simulate IoT ransomware and their operation to analyse and prepare for what we believe is an eventuality. They had not analysed the viability of ransomware targeting IoT devices and the data centres and servers which keep them running, as well as using this analysis to determine which of the two is the more likely target.

Paper 2: In the research paper they had not taken a greater number of ransomware samples and performed the experiment with other machine learning algorithms including deep learning.

Paper 3: The matrix they have used for accuracy, uses all values in the confusion matrix. This means that they may not fully represent the results in the confusion matrix on which their formulas are based. The second gap is the inability of the individual metrics to provide an indication of how well the predictive model can differentiate between positive and negative results.

Paper 4: They had not worked on a hybrid feature selection and adaptive model. It needs a big data sample

Paper 5: The LA has a very poor chance of prediction if the J index value is larger than 0.99. Although the Zoo's ransomware had the smallest net benefit of 0.0500, the lower sample size did not have a substantial impact on the entire dataset created.

CHAPTER 4

REQUIREMENTS

4.1. Software Requirements

S.No.	Item	Version	Spec	Vendor	Price	Description
01	Python Notebook	4.8	5.0	Jupyter	Open Source	Jupyter Notebook (formerly IPython Notebooks) is a web-based interactive computational environment for creating notebook documents.
02	Operating System	10	Windows 10	Microsoft	Rs. 10,000	Windows 10 is a major release of Microsoft's Windows NT operating system. It is the direct successor to Windows 8.1, which was released nearly two years earlier.

Table 1: Summary of Software Components

Justification for the software usage

Windows is a graphical operating system developed by Microsoft. It allows users to view and store files, run the software, play games, watch videos, and provides a way to connect to the internet. It was released for both home computing and professional works.

The Jupyter Notebook is an open-source web application that allows data scientists to create and share documents that integrate live code, equations, computational output, visualizations, and other multimedia resources, along with explanatory text in a single document. One can use Jupyter Notebooks for all sorts of data science tasks including data cleaning and transformation, numerical simulation, exploratory data analysis, data visualization, statistical modeling, machine learning, deep learning, and much more.
Jupyter is a way of working with Python inside a virtual “notebook” and is growing in popularity with data scientists in large part due to its flexibility. It gives you a way to combine code, images, plots, comments, etc., in alignment with the step of the “data science process.

CHAPTER 5

Design

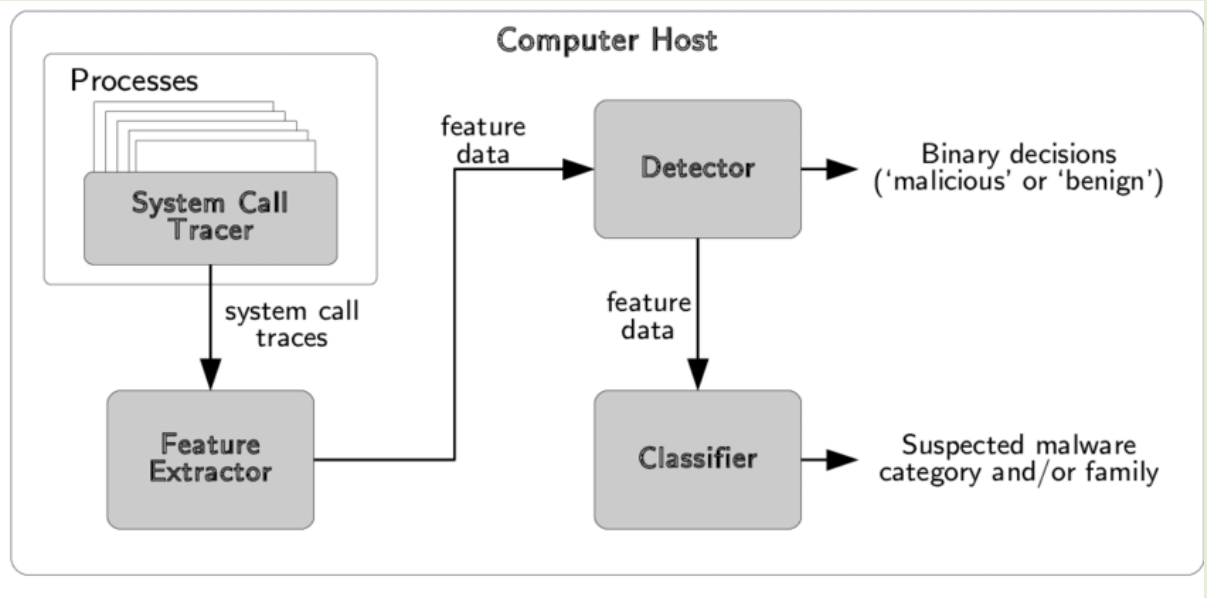

Fig. 1, Block Diagram of the proposed architecture

There are many different machine learning mechanisms that are used today for both detecting and protecting your data from a ransomware infection:

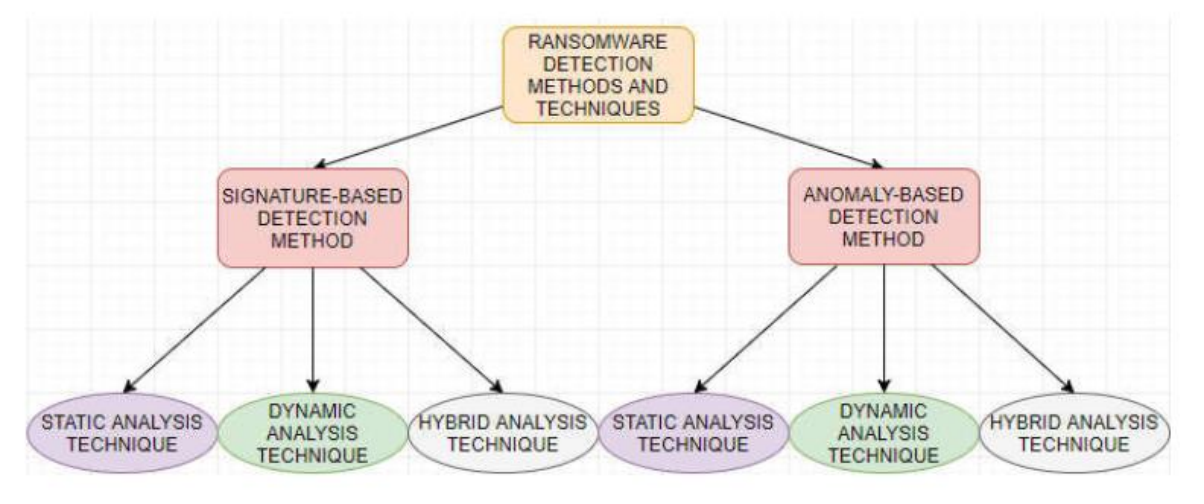

Fig. 2, Classification of ransomware detection techniques and methods

The following diagram includes the sections explaining each processing step involved in the proposed method:

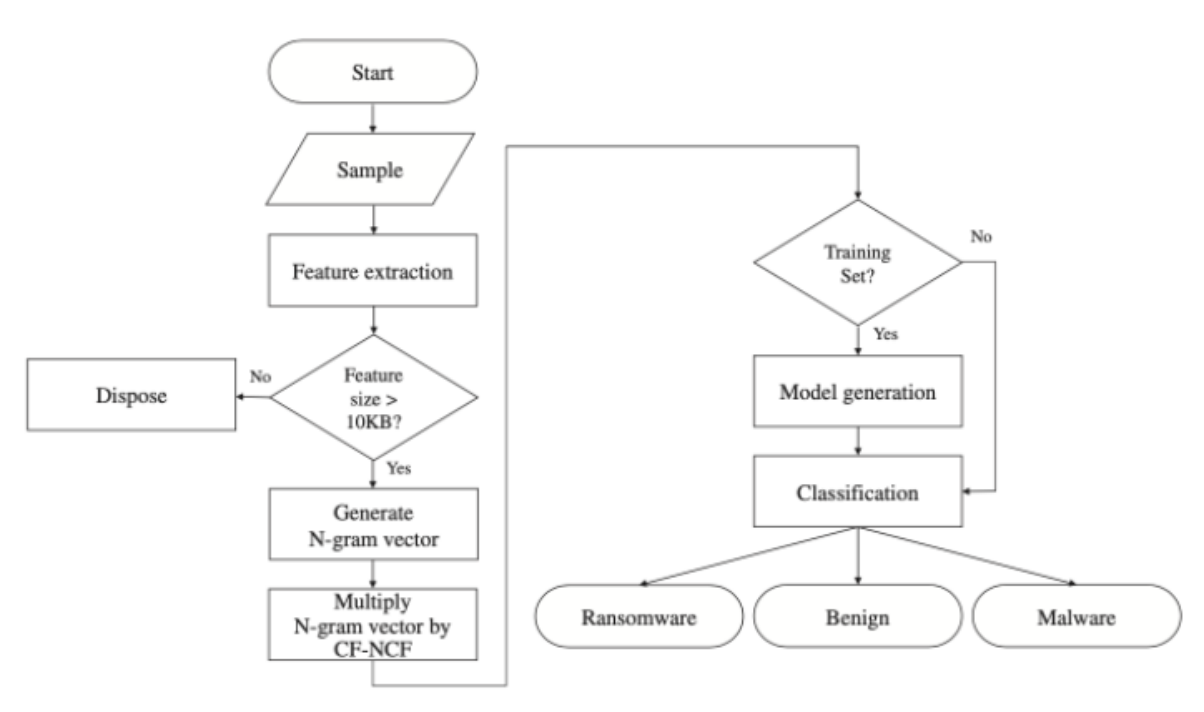

Fig. 3, Step-wise processing of the architecture used

The overview of the experiments performed for Ransomware Detection is explained through the following diagram:

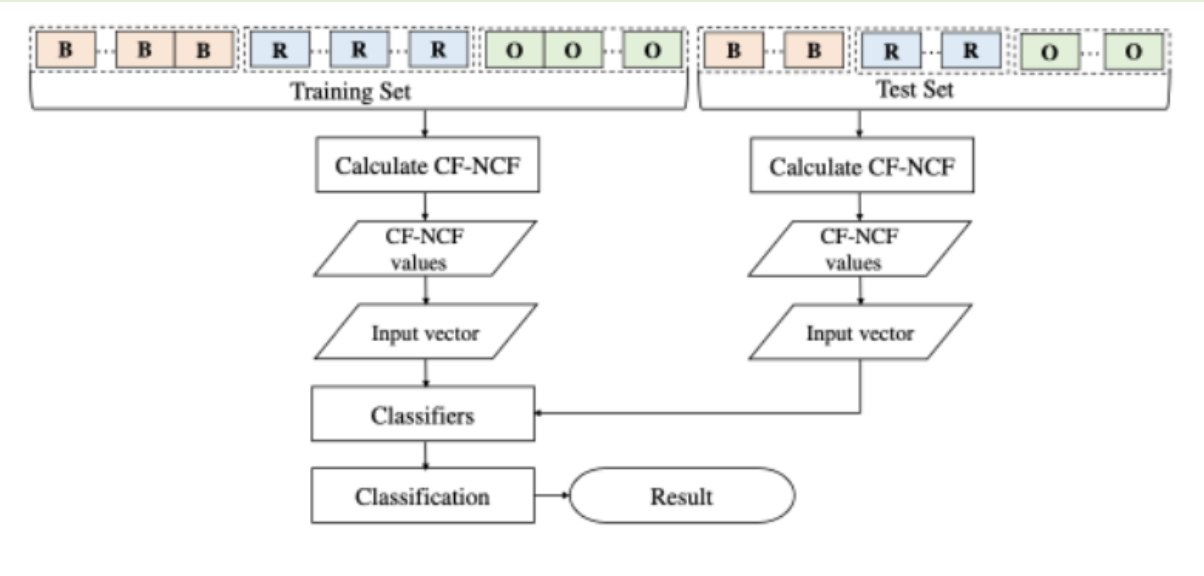

Fig. 4, Training and testing the dataset gathered for detection of ransomware

CHAPTER 6

IMPLEMENTATION

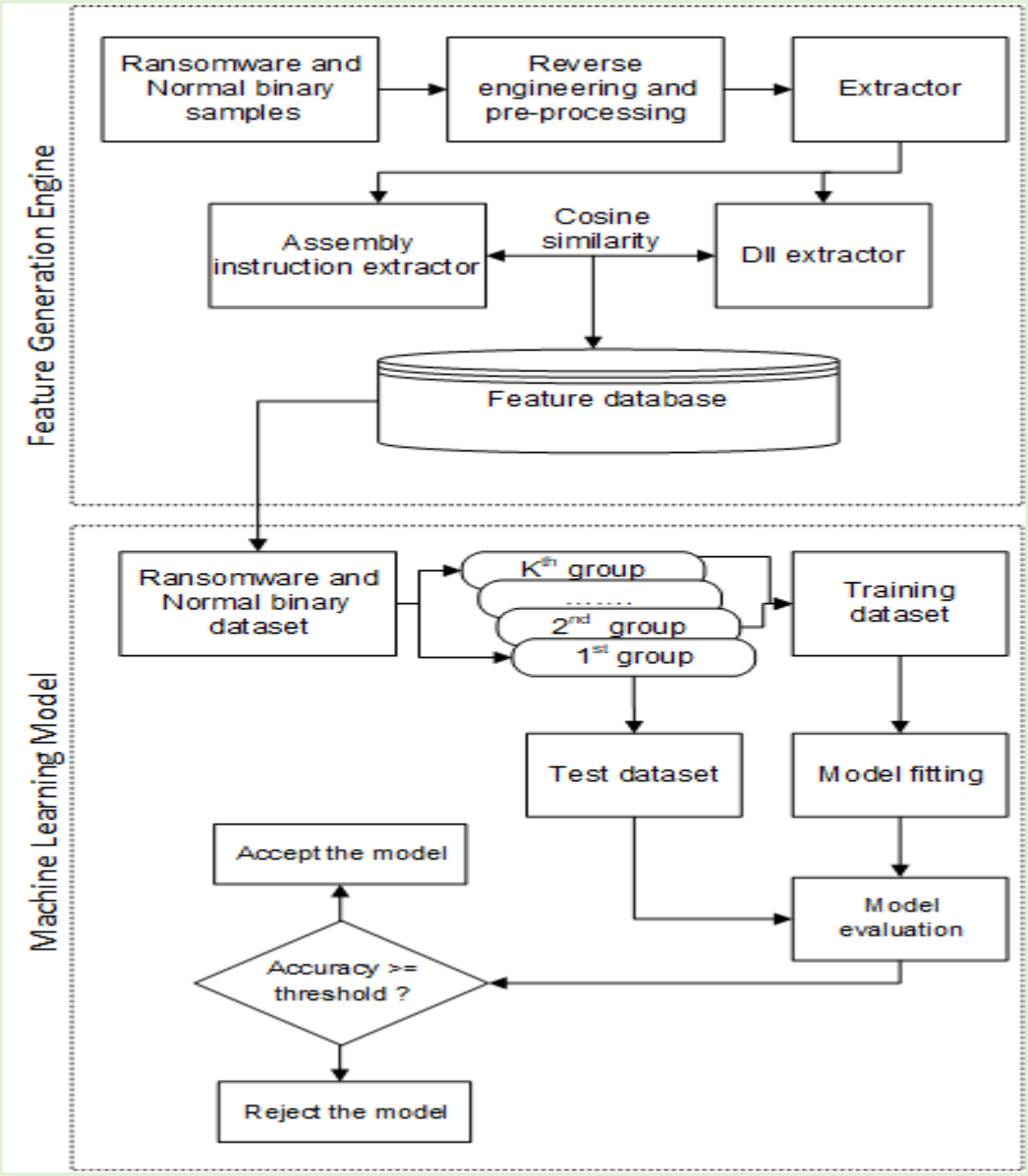

Fig. 5, Flow chart depicting the working of the parts: feature generation engine and machine learning model of our project

Algorithm

1. The implementation starts with the **gathering of data** from Bitcoin Heist Ransomware Address dataset. The dataset consists of 10 attributes in which the last attribute is the target label of the bitcoin addresses.

2. The, comes the **data pre-processing**. Irrelevant features like year, day, Bitcoin address were dropped since year and day does not add any value to the classification and each of the Bitcoin addresses are unique so it's of no use in the task of detecting bitcoin ransomware addresses.

3. **Binary label encoding** was performed on target label, neighbours, length, count, looped columns. In target label column white label was encoded as 1 and rest of the labels grouped as black label was encoded as 0. In neighbour column those values which was greater than 2 was encoded 0 else 1, in length column those values which was greater than 8 was encoded as 0 else 1, in count as well as looped columns those values which was greater than 1 was encoded as 0 else 1. Encoding as 1 indicates it's a non-ransomware address whereas encoding 0 indicates it's a ransomware address.

4. **Model Building:** The dataset was divided into 80% train and 20% test splits. Various scaling techniques like StandardScaler, MinMaxSccaler, RobustScaler was applied to both the train and test data to scale them down appropriately.

5. After that, various **supervised classification machine learning models** was applied like logistic regression, KNN, SVM, Decision Tree, Random Forest, AdaBoost, XGBoost, Neural Networks. To evaluate our models, we have used metrics like accuracy, precision, recall, F1 score and ROC

    - True positive rate (T P R) = T P/ (T P + F N ……... (1)
    - False positive rate (F P R) = F P/ (F P + T N) ……. (2)
    - Precision = T P/ (T P + F P) ……. (3)
    - Recall = T P/ (T P + F N) ……. (4)
    - F-measure = 2∗Precision∗Recall/ (Precision + Recall) …… (5)
    - Accuracy = T P + T N/ (T P + T N + F P + F N) ……. (6)

In the above equations, TP is a true positive which represents the number of ransomware samples correctly classified. TN is a true negative which represents the number of normal samples correctly classified. FP is a false positive which represents normal binaries incorrectly classified as ransomware. FN is a false negative which represents ransomware incorrectly classified as normal binary.

TPR gives the value of predicted ransomware classified correctly as ransomware, whereas FPR gives the value of normal binaries incorrectly classified as ransomware. Precision defines the accuracy of the machine learning model in terms of classifying the relevant instances. Recall defines the ability to find the relevant instances in the dataset. F-measure is the harmonic mean of precision and recall and estimates the performance of the given machine learning model. The accuracy is defined by the ratio of correctly predicted instances to the total testing instances expressed in percentage. ROC curves are frequently used to show in a graphical way the connection/trade-off between clinical sensitivity and specificity for every possible cut-off for a test or a combination of tests. In addition, the area under the ROC curve gives an idea about the benefit of using the test(s) in question.

UML Diagrams

- **Activity Diagram**
- **Use-Case Diagram**
- **Sequence Diagram**

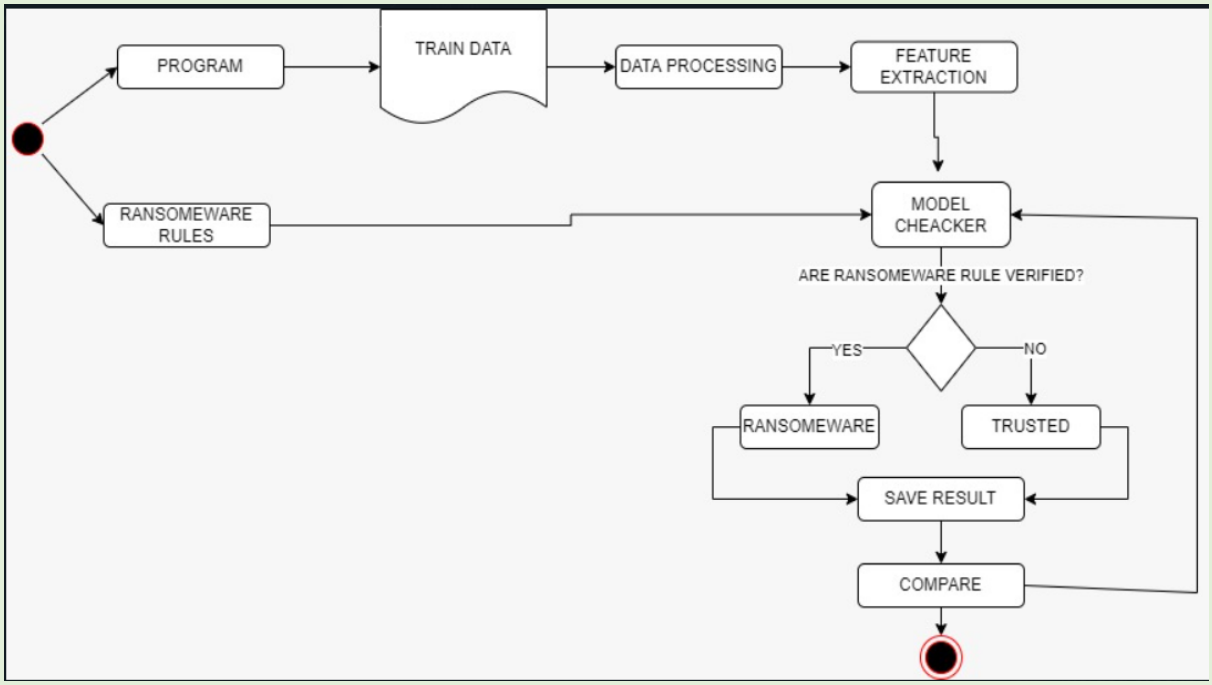

Fig. 6, Activity diagram for the ransomware detection

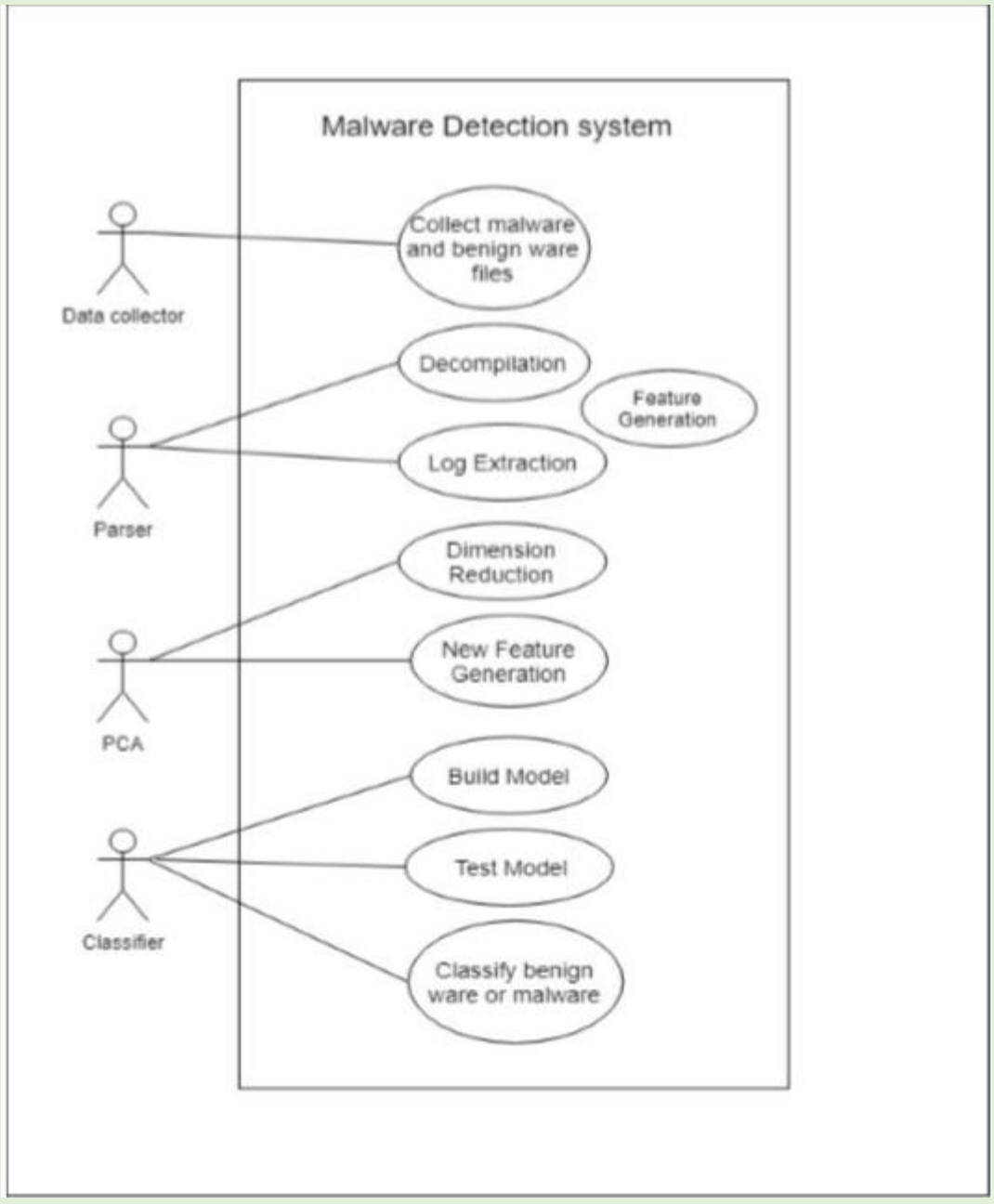

Fig. 7, Use-case diagram for the ransomware detection

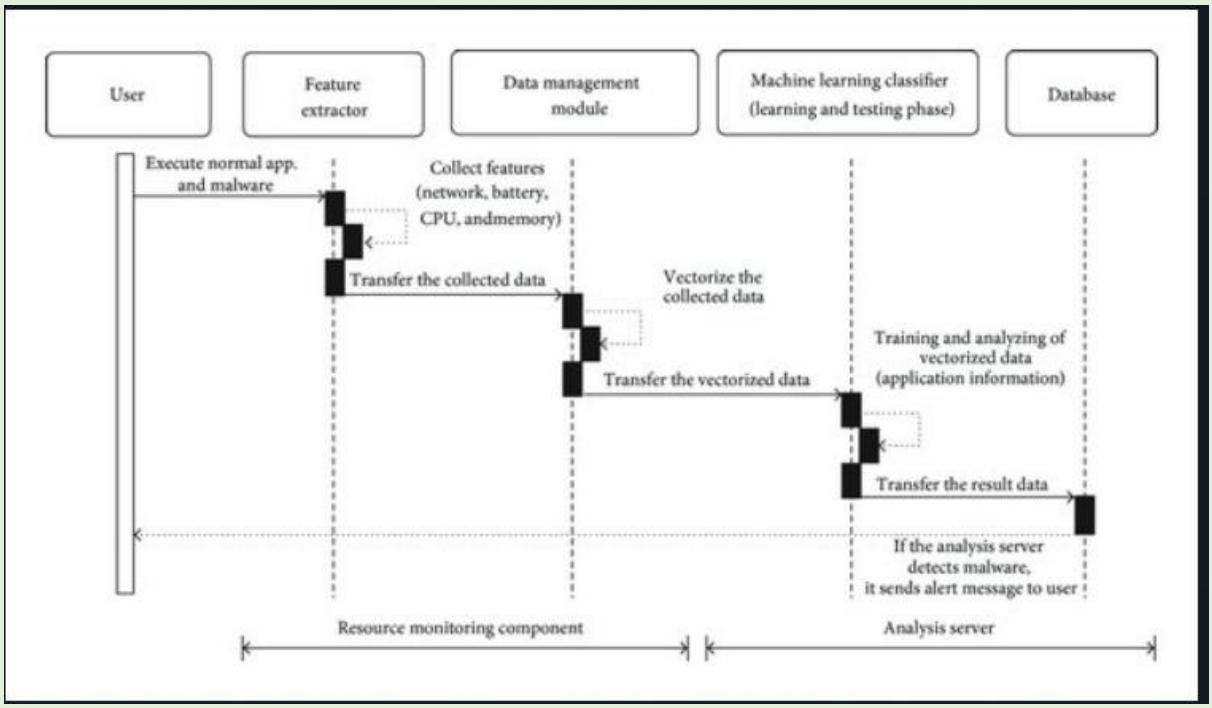

Fig. 8, Sequence Diagram for the ransomware detection

CHAPTER 7

RESULTS ANALYSIS

	length	weight	count	looped	neighbors	income
length	1.000000	0.132993	0.519681	0.406445	-0.094199	-0.002613
weight	0.132993	1.000000	-0.043395	0.055573	-0.362437	0.259408
count	0.519681	-0.043395	1.000000	0.630045	-0.010823	-0.015450
looped	0.406445	0.055573	0.630045	1.000000	-0.073472	-0.007087
neighbors	-0.094199	-0.362437	-0.010823	-0.073472	1.000000	-0.403073
income	-0.002613	0.259408	-0.015450	-0.007087	-0.403073	1.000000
label_encoded	0.024864	-0.327689	0.024238	-0.112542	0.269160	-0.140211

	label_encoded
length	0.024864
weight	-0.327689
count	0.024238
looped	-0.112542
neighbors	0.269160
income	-0.140211
label_encoded	1.000000

Fig. 9, Correlation matrix of the features used by us in ransomware detection

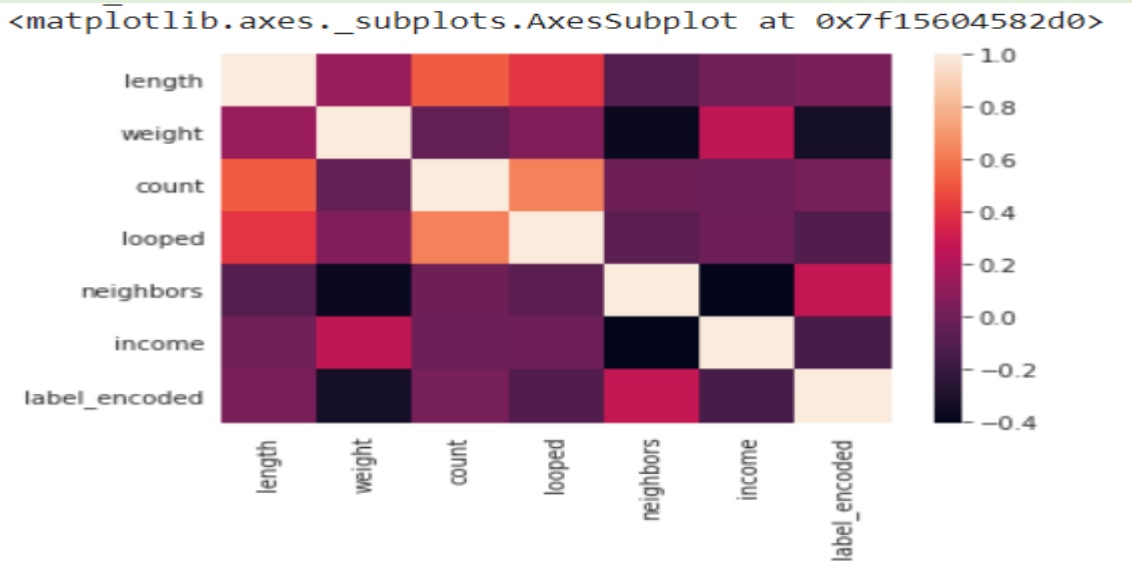

Fig. 10, Heat map of the features used by us in ransomware detection

	precision	recall	f1-score	support
0	0.68	0.76	0.72	893
1	0.73	0.64	0.68	896
accuracy			0.70	1789
macro avg	0.70	0.70	0.70	1789
weighted avg	0.70	0.70	0.70	1789

Logistic Regression

	precision	recall	f1-score	support
0	0.70	0.77	0.73	893
1	0.74	0.67	0.71	896
accuracy			0.72	1789
macro avg	0.72	0.72	0.72	1789
weighted avg	0.72	0.72	0.72	1789

KNN

	precision	recall	f1-score	support
0	0.69	0.84	0.75	893
1	0.79	0.62	0.69	896
accuracy			0.73	1789
macro avg	0.74	0.73	0.72	1789
weighted avg	0.74	0.73	0.72	1789

Support Vector

	precision	recall	f1-score	support
0	0.67	0.67	0.67	893
1	0.67	0.67	0.67	896
accuracy			0.67	1789
macro avg	0.67	0.67	0.67	1789
weighted avg	0.67	0.67	0.67	1789

Decision Tree

	precision	recall	f1-score	support
0	0.71	0.76	0.73	893
1	0.74	0.68	0.71	896
accuracy			0.72	1789
macro avg	0.72	0.72	0.72	1789
weighted avg	0.72	0.72	0.72	1789

Random Forest

	precision	recall	f1-score	support
0	0.69	0.81	0.74	893
1	0.77	0.64	0.70	896
accuracy			0.72	1789
macro avg	0.73	0.72	0.72	1789
weighted avg	0.73	0.72	0.72	1789

Adaboost

	precision	recall	f1-score	support
0	0.71	0.82	0.76	893
1	0.79	0.66	0.72	896
accuracy			0.74	1789
macro avg	0.75	0.74	0.74	1789
weighted avg	0.75	0.74	0.74	1789

Gradient Boosting

	precision	recall	f1-score	support
0	0.71	0.81	0.76	893
1	0.78	0.67	0.72	896
accuracy			0.74	1789
macro avg	0.74	0.74	0.74	1789
weighted avg	0.74	0.74	0.74	1789

Artifical Nueral Networks

	precision	recall	f1-score	support
0	0.71	0.83	0.76	893
1	0.79	0.66	0.72	896
accuracy			0.74	1789
macro avg	0.75	0.74	0.74	1789
weighted avg	0.75	0.74	0.74	1789

ANN with multilayer perceptron

Fig. 11, Accuracy and Precision of every ML model used in our project

Fig. 12, ROC curve of every ML model used in ransomware detection by us

CHAPTER 8

Applicability category

Our project has a wide variety of applicability in many real-life categories and scenarios. It is relevant in the following areas:

1. Social Relevance: The number of ransomware variants has increased rapidly every year, and ransomware needs to be distinguished from the other types of malwares to protect users' machines from ransomware-based attacks. Ransomware detection, hence, shall act as a safeguard for the society to protect people's sensitive information.

2. Healthcare: Ransomware is a type of malware that infects systems and files, rendering them inaccessible until a ransom is paid. When this occurs in the healthcare industry, critical processes are slowed or become completely inoperable. Hospitals are then forced to slow down the medical process and ultimately soaking up funds. Thus, the project shall prove beneficial there.

3. Social Networking: Online social networks (OSNs) have become the new vector for cybercrime, and hackers are finding new ways to propagate spam and malware on these platforms. Hence, Ransomware detection shall help in preventing this social malware.

Chapter 9

CONCLUSION

With the increased number of ransomware-based cyber-attacks, ransomware detection methods are needed. Even though ransomware share many aspects of features with other types of malware, some features are specific to ransomware. Because neural networks have the capacity to learn on their own and produce output which is not limited by inputs, they can learn from past events and apply what they have learned whenever a similar circumstance arises, enabling them to cope with real-time problems, therefore it performs better than conventional machine learning algorithms. It would have performed much more significantly if the proportion of white to black labels was more, if the data had not been skewed and even with log transformations the data is still not normal enough to show good metric results. Ransomware poses a huge threat to people's daily lives and social networks. In these attacks, the attacker appears as a reputable company in order to obtain sensitive and crucial information. The methodology identified in this paper, is a robust strategy for detecting ransomware and can give more effective defenses against future ransomware assaults.

Chapter 10

REFERENCES

1. Bae, S. I., Lee, G. B., & Im, E. G. (2020). Ransomware detection using machine learning algorithms. Concurrency and Computation: Practice and Experience, 32(18), e5422.

2. Fernando, D. W., Komninos, N., & Chen, T. (2020). A study on the evolution of ransomware detection using machine learning and deep learning techniques. IoT, 1(2), 551-604.

3. Poudyal, S., Subedi, K. P., & Dasgupta, D. (2018, November). A framework for analyzing ransomware using machine learning. In 2018 IEEE symposium series on computational intelligence (SSCI) (pp. 1692-1699). IEEE.

4. J. A. H. Silva, L. I. B. López, Á. L. V. Caraguay, and M. Hernández-Álvarez, ‘‘A survey on situational awareness of ransomware attacks—Detection and prevention parameters,’’ Remote Sens., vol. 11, no. 10, p. 1168, May 2019.

5. Kok, S. H., Azween, A., & Jhanjhi, N. Z. (2020). Evaluation metric for crypto-ransomware detection using machine learning. Journal of Information Security and Applications, 55, 102646.

6. M.M. Hassan Avoiding future digital extortion through robust protection against ransomware threats using deep learning based adaptive approaches IEEE Access, 8 (2020), pp. 24522-24534 doi:10.1109/ACCESS.2020.2970466

7. Khalaf, B.A.; Mostafa, S.A.; Mustapha, A.; Mohammed, M.A.; Mahmoud, M.A.; Al-Rimy, B.A.S.; Abd Razak, S.; Elhoseny, M.; Marks, A. An Adaptive Protection of Flooding Attacks Model for Complex Network Environments. Secur. Commun. Netw. 2021, 2021, 5542919.

8. Fernandez Maimo, L.; Huertas Celdran, A.; Perales Gomez, A.L.; Garcia Clemente, F.J.; Weimer, J.; Lee, I. Intelligent and dynamic ransomware spread detection and mitigation in integrated clinical environments. Sensors 2019, 19, 1114.

9. Sharmeen, S.; Ahmed, Y.A.; Huda, S.; Koçer, B. ¸S.; Hassan, M.M. Avoiding future digital extortion through robust protection against ransomware threats using deep learning based adaptive approaches. IEEE Access 2020, 8, 24522–24534.

10. Al-Rimy, B.A.S.; Maarof, M.A.; Alazab, M.; Alsolami, F.; Shaid, S.Z.M.; Ghaleb, F.A.; Al-Hadhrami, T.; Ali, A.M. A pseudo feedback-based annotated TF-IDF technique for dynamic crypto-ransomware pre-encryption boundary delineation and features extraction. IEEE Access 2020, 8, 140586–140598.