Industrial Engineering Journal
ISSN: 0970-2555
Volume : 53, Issue 4, April : 2024

DETECTION OF MALICIOUS CODE VARIANTS BASED ON DEEP LEARNING

P.S.S.Geethika[1],A.Y.S.S.Prasanth[2],V.Srujana[3],CH.Sai Santosh[4],

[1]**Assistant Professor,Department of Computer Science & Engineering, Raghu Engineering College, Vishakhapatnam, Andhra Pradesh**

[2,3,4]**Department of Computer Science & Engineering, Raghu Engineering College , Vishakhapatnam, Andhra Pradesh**

E-mail address: saigeethika.p@raghuenggcollege.in, 203j1a4603@raghuinstech.com, 203j1a4661@raghuinstech.com, 203j1a4612@raghuinstech.com

ABSTRACT

Cyber danger exists for every business. The news reports validate the facts: cyberattacks are increasing in unprecedented ways in terms of frequency, sophistication, and scope. Furthermore, they are widespread outside national and industry borders. Regulations pertaining to data security and privacy are multiplying as grave cyberthreats are frequently featured in the media. Companies worldwide are placing a high premium on addressing and reducing cyber risk, as seen by the surge in data security breaches, denial of service attacks, and other attacks as well as data loss. It is obvious that network security by itself is unable to fully solve the problem of cyber risk; no security system is impenetrable, and no firewall is impenetrable. In order to address, minimize, and optimize protection against cyber risk, insurance can be a critical component of a business's entire strategy. The Securities and Exchange Commission is aware of this fact. Following "more frequent and severe cyber incidents," the Division of Corporation Finance of the SEC has released guidelines for disclosures related to cyber security under the federal securities laws. The guidance states that businesses "should review, on an ongoing basis, the adequacy of their disclosure relating to cyber security risks and cyber incidents" as well as that "appropriate disclosures may include" a "[d] description of relevant insurance coverage." One current worldwide concern pertaining to human rights is gun violence. The right to life, which is our most fundamental human right, is threatened by gun-related violence. Every day tragedies involving gun violence impact people's lives all throughout the world. Every day, gun violence claims the lives of more than 500 people. The goal of developing OpenCV, an open source, cross-platform, cross-language computer vision library, was to increase the use of computer vision in commercial goods by providing a common foundation for these applications. The library supports hardware accelerators like CUDA and SSE and has over 2500 optimized algorithms.

KEYWORDS:- CUDA,PSO,SVM, optimized algorithms, cyber incidents

1. INTRODUCTION

The word "CMalicious" refers to malicious software, which includes viruses, Trojan horses, worms, and other similar programs. It is an acronym for "malicious software." Numerous functionalities are included in these programs, including the ability to steal, encrypt, or remove private information, alter or take control of fundamental computer operations, and keep an eye on user activities. Display user consent.

1.1 PC viruses are among the malicious kind.

It's typically an application that is installed on a computer without the user's consent and has the potential to harm the hardware (physical components of the computer) as well as the operating system.

The virus's effects include:

● File destruction.

Sizing up or down a file.

● Remove everything from the CD, including the formatting.

The information on the disk cannot be read due to the destruction of the file allocation table.

A variety of benign but unsettling visual and acoustic effects

● Decrease the computer's operation speed till it crashes.

Worms: The Worm Computer worms are harmful programs that propagate by communicating between computers. Worms and

viruses have the same characteristics:

Worms can replicate similarly to viruses, but on other computers rather than locally. To expand to other systems, I use computer networks.

Computer worm types include:

TM Worms for Instant Messaging

● Email Worms

worms on the Internet

● IRC worms

1.2 Trojan horses are as follows: Trojan horses are "disguised" programs that attempt to get into the operating system and grant users access. Unlike computer viruses, Trojans lack the ability to replicate themselves.Trojan horses fall into a few

1.3 **different groups:**

Backdoors: these programs enable an attacker to gain access to the victim's computer through the Internet; Password stealers: these programs read passwords from the keyboard and store them in files that the attacker can access at a later time or send directly to an email account. mail of the attacker);

● **Logical bombs:** these Trojans can carry out actions that jeopardize system security under specific circumstances;

● **Denial of Service tools:** these are programs that send specific data sequences to the intended audience (typically a website) with the goal of disrupting that target's Internet services.

Ransomware: Ransomware is a kind of malware that prevents its target from using their computer and requests payment of a ransom.

2. LITERATURE SURVEY AND RELATED WORK

You Only Look Once: Unified, Real-Time Object Detection, by Joseph Redmon. Their prior work is on detecting objects using a regression algorithm. To get high accuracy and good predictions they have proposed YOLO algorithm in this paper [1]. Understanding of Object Detection Based on CNN Family and YOLO, by Juan Du. In this paper, they generally explained about the object detection families like CNN, R-CNN and compared their efficiency and introduced YOLO algorithm to increase the efficiency. Learning to Localize Objects with Structured Output Regression, by Matthew B. Blaschko. This paper is about Object Localization. In this, they used the Bounding box method for localization of the objects to overcome the drawbacks of the sliding window method.

2.1. Viruses, Trojans, And Spyware, Oh My! The yellow brick road to coverage in the land of internet OZ

Author: Roberta D. Anderson

Tort Trial & Insurance Practice Law Journal

3. Implementation Study

Network security is the key challenges for much enterprise level application which are mainly used identify the security threats associated across different types of attacks by hackers as well as intruders who always involve in spoofing the data during the data transmission.

3.1 Proposed Methodology

The techniques outlined in the article apply to both computer vision (CV) libraries and frameworks. OpenCV (Open Source Computer Vision Library) stands out as one of the most widely used libraries in the field. Initially designed to enhance the integration of computers and artificial intelligence, OpenCV facilitates the rapid deployment of machines as commercial products. The library encompasses numerous steps for image processing and data acquisition, aimed at streamlining machine operations. The methodological approach in the article is divided into three distinct sections for clarity: Data Collection Image Processing Object Recognition The objective of this project is to investigate the implications of deploying weapons and identifying objects captured in camera images. The article discusses utilizing the Haar cascade technique, a machine learning-based method involving the classification of images into positive (good) and negative (bad) categories during training. Positive images

typically contain instances of the classes one wishes to identify, while negative images encompass everything else. Consequently, the primary aim is to evaluate the real-world outcomes of the data available online and to ascertain them through OpenCV. Moreover, incorporating YOLOv8, a state-of-the-art object detection model, into the project can significantly enhance accuracy and speed compared to traditional OpenCV methods. By training the YOLOv8 model with a customized dataset, the system can achieve superior performance in object detection tasks, offering both accuracy and efficiency benefits. This integration broadens the capabilities of the system, enabling it to handle a wider range of object recognition tasks with improved precision and speed.

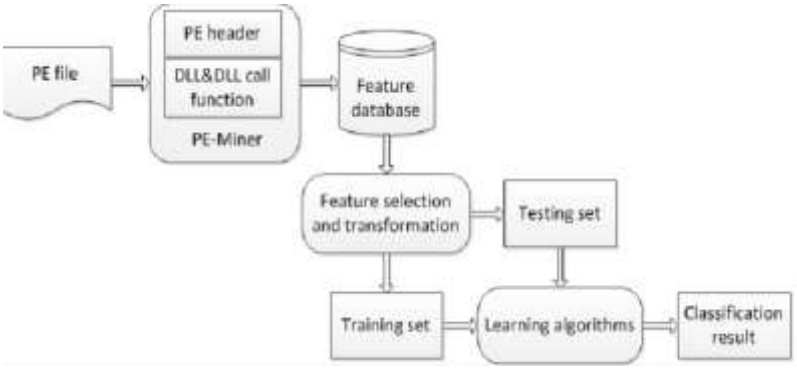

Fig 1:- Proposed SYSTEM ARCHITECTURE

4. METHODOLOGY & ALOGRITHAM

4.1 DATA PREPROCESSING

Data preprocessing is a process of preparing the raw data and making it suitable for a machine learning model. It is the first and crucial step while creating a machine learning model.

When creating a machine learning project, it is not always a case that we come across the clean and formatted data. And while doing any operation with data, it is mandatory to clean it and put in a formatted way. So, for this, we use data preprocessing task.

4.2 DATA EXPLORATION

Data exploration refers to the initial step in data analysis in which data analysts use data visualization and statistical techniques to describe dataset characterizations, such as size, quantity, and accuracy, in order to better understand the nature of the data.

4.3 MODEL CREATION

A deep learning model is a file that has been trained to recognize certain types of patterns. You train a model over a set of data, providing it an algorithm that it can use to reason over and learn from those data.

4.4 TRAINING AND TESTING

This is an equation that involves calculating the inner products of a new input vector (x) with all support vectors in training data. The coefficients B0 and ai (for each input) must be estimated from the training data by the learning algorithm.

4.5 PREDICTION

Training data is the initial dataset you use to teach a machine learning application to recognize patterns or perform to your criteria, while testing or validation data is used to evaluate your model's accuracy. You'll need a new dataset to validate the model because it already “knows” the training data

.

4.6 ALOGRITHAM

4.6.1 Random Forest:

Random Forest is a powerful ensemble learning method that employs multiple decision trees to enhance prediction accuracy and mitigate overfitting. During training, each tree is constructed using a random subset of the training data and features, ensuring diversity among the trees. When making predictions, the algorithm aggregates the results of individual trees through a majority voting mechanism, producing a robust final prediction. Moreover, it can provide insights into feature importance, aiding

Industrial Engineering Journal
ISSN: 0970-2555
Volume : 53, Issue 4, April : 2024

in the identification of relevant features for the classification task.

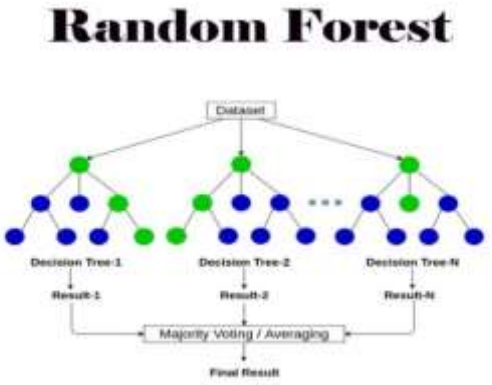

Figure 2: - Random Forest

4.6.2 K-Nearest Neighbors (KNN):

K-Nearest Neighbors (KNN) is a simple yet effective algorithm used for classification and regression tasks in machine learning. It operates on the principle of similarity, where an unlabeled instance is classified based on the class labels of its nearest neighbors in the feature space. Here's how KNN works and its application in detecting phishing websites:

In the context of detecting phishing websites, KNN can be applied by first defining relevant features extracted from websites, such as URL characteristics, domain attributes, and content-based features. These features serve as dimensions in the feature space, and each website is represented as a point in this space.

During the classification phase, when a new website is encountered, KNN identifies its k nearest neighbors based on the similarity of their features to the features of the new website. The class labels of these neighbors are then used to predict the class of the new website. For example, if the majority of the k nearest neighbors are labeled as phishing websites, the new website is classified as phishing as well. By considering the characteristics and attributes of known phishing websites, KNN can effectively identify similarities between new and existing instances, enabling accurate classification decisions

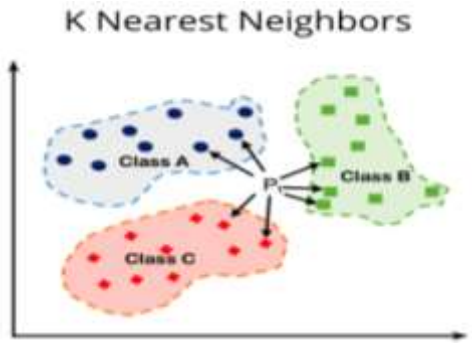

Figure 3: - K Nearest Neighbor

4.6.3 Naïve Bayes:

Naive Bayes is a probabilistic classifier based on Bayes' theorem with the assumption of feature independence. This means that it calculates the probability of a hypothesis (e.g., a website being phishing) given the observed evidence (e.g., features extracted from the website) using conditional probability. Despite its simplifying assumption, Naive Bayes is widely used in machine learning

for classification tasks due to its simplicity and efficiency. By modeling the relationship between features and classes using probabilities, Naive Bayes can effectively classify instances based on their observed attributes.

In the context of detecting phishing websites, Naive Bayes proves to be useful for several reasons. Firstly, it can analyze various features extracted from websites, including URL attributes (e.g., length, presence of certain keywords), domain characteristics (e.g., age, registration information), and content (e.g., presence of suspicious links or language). This multifaceted analysis allows Naive Bayes to capture diverse patterns associated with phishing activity, contributing to its effectiveness in classification. Additionally, Naive Bayes requires minimal training data compared to more complex algorithms, making it particularly useful in scenarios where labeled datasets are limited or costly to obtain.

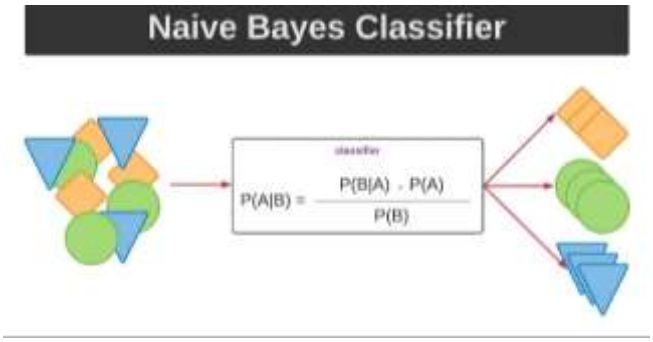

Figure 4: - Naïve Bayes

4.6.4 CNN Algorithm

1. **Data Collection**: Obtain a dataset of benign and malicious code samples. These could include executables, scripts, or other forms of code. Ensure the dataset is labeled with information on whether each sample is benign or malicious.
2. **Feature Extraction**: Convert the code samples into a format suitable for deep learning models. This might involve techniques like converting binary code into numerical representations or extracting features such as opcode sequences, n-grams, or control flow graphs.
3. **Preprocessing**: Prepare the data for training by performing preprocessing steps such as normalization, balancing the dataset, and splitting it into training, validation, and test sets.
4. **Model Selection**: Choose an appropriate CNN architecture for your task. You may experiment with different architectures, including standard architectures like VGG, ResNet, or custom architectures tailored to your specific problem.
5. **Training**: Train the CNN using the labeled dataset. During training, the model learns to distinguish between benign and malicious code based on the features extracted from the samples.
6. **Evaluation**: Evaluate the trained model on a separate test dataset to assess its performance. Metrics such as accuracy, precision, recall, and F1-score can be used to measure the effectiveness of the model.
7. **Fine-tuning and Optimization**: Fine-tune the model and experiment with hyperparameters to improve performance. Techniques such as data augmentation, regularization, and transfer learning can also be applied to enhance the model's generalization ability.
8. **Deployment**: Once satisfied with the model's performance, deploy it for real-world use. This may involve integrating the model into existing security systems or developing standalone malware detection solutions.

It's important to note that the effectiveness of CNNs for malware detection depends on various factors such as the quality of the dataset, feature representation, model architecture, and training strategy. Additionally, keep in mind the ethical considerations and legal implications associated with malware research and detection.

Industrial Engineering Journal
ISSN: 0970-2555
Volume : 53, Issue 4, April : 2024

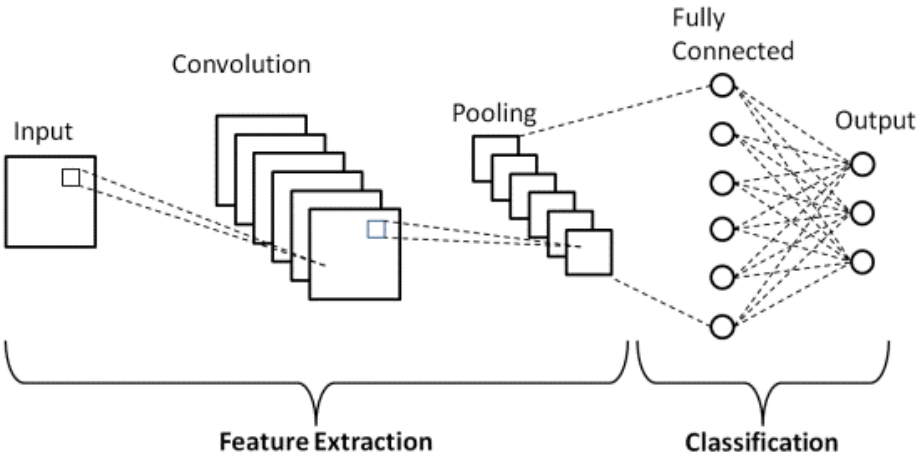

Fig 5:- CNN Model proposed System

6. RESULTS AND DISCUSSION SCREEN SHOTS

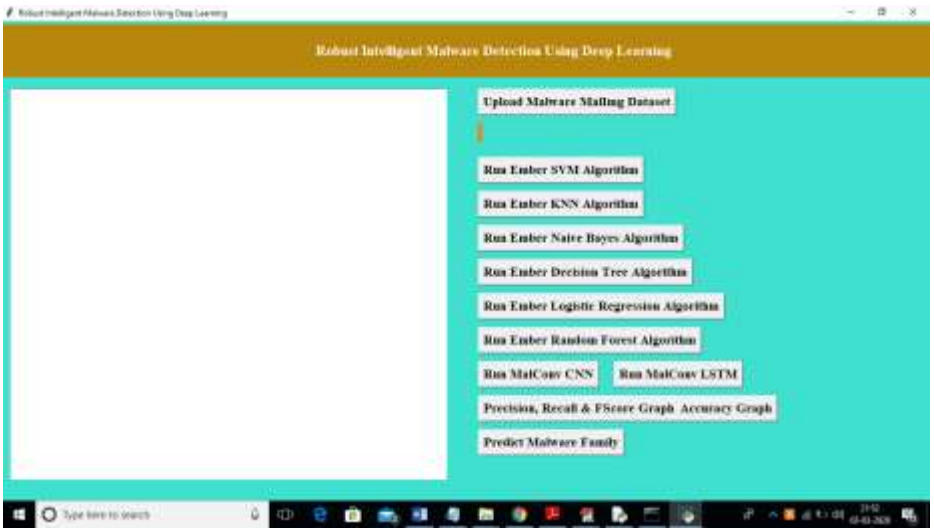

Fig 6 :-In above screen click on 'Upload Malicious MalImg dataset' button to upload dataset

Fig 6.1 Login Page

Industrial Engineering Journal
ISSN: 0970-2555
Volume : 53, Issue 4, April : 2024

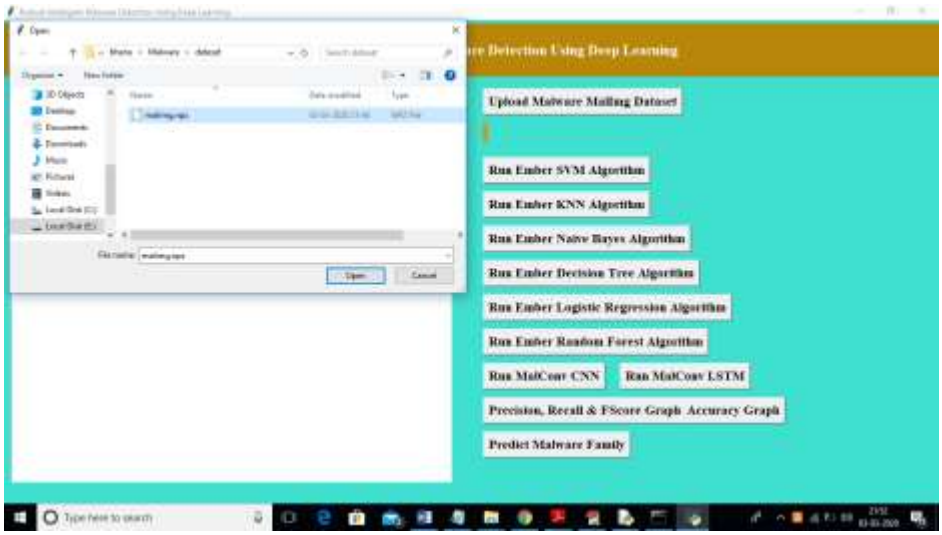

Fig 7 :-In above screen I am uploading 'malimg.npz' binary Malicious dataset and after uploading dataset will get below screen For Upload File

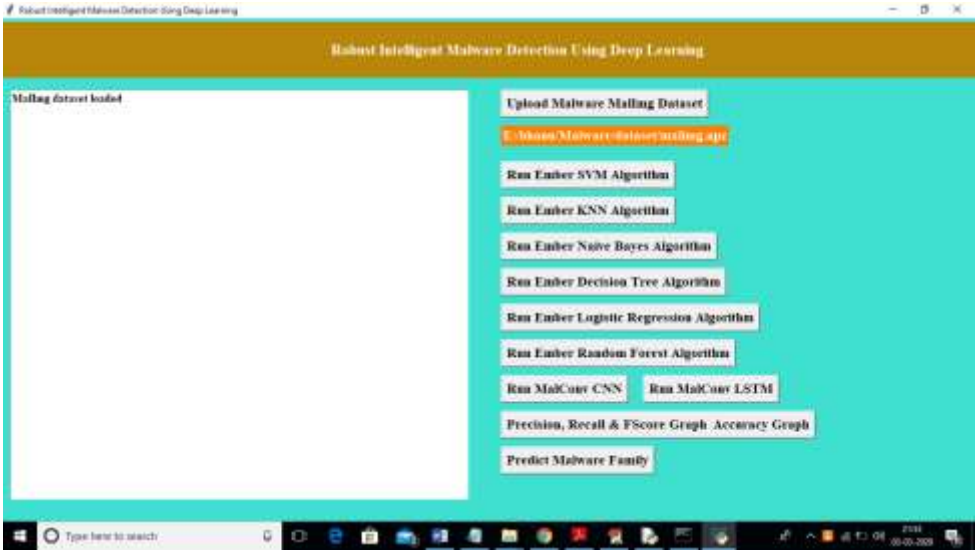

Fig 8 View Uploaded Dataset Now click on 'Run Ember SVM algorithm' button to read Malicious dataset and generate train and test model and then apply SVM algorithm to calculate its prediction accuracy, FSCORE, Precision and Recall. If algorithm performance is good then its accuracy, precision or recall values will be closer to 100.

Industrial Engineering Journal
ISSN: 0970-2555
Volume : 53, Issue 4, April : 2024

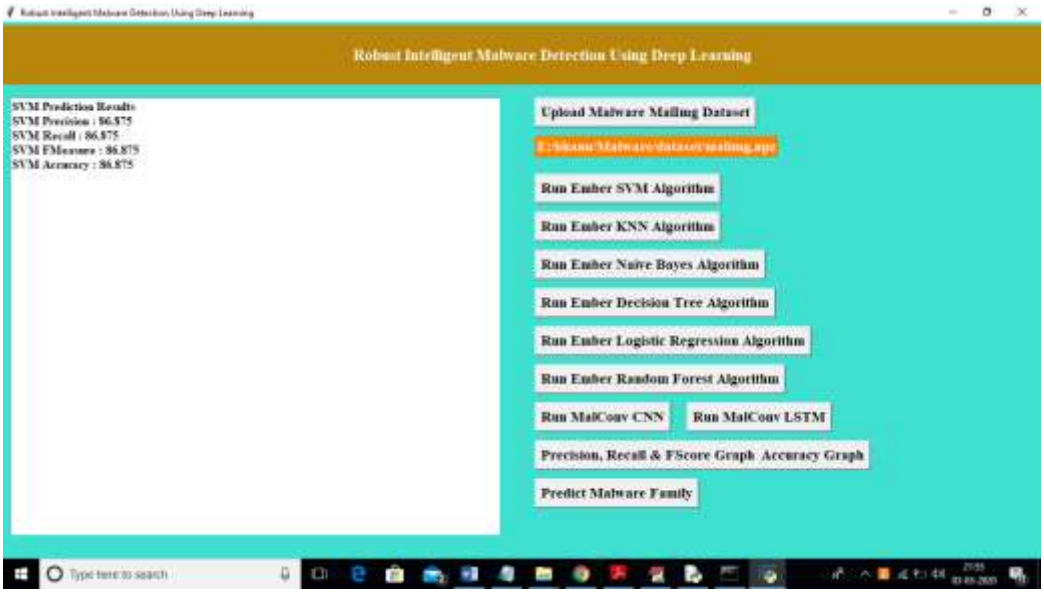

Fig 9 SVM Pediction In above screen we got SVM precision, recall and fSCORE. Now click on 'Run Ember KNN Algorithm' button to get its performance

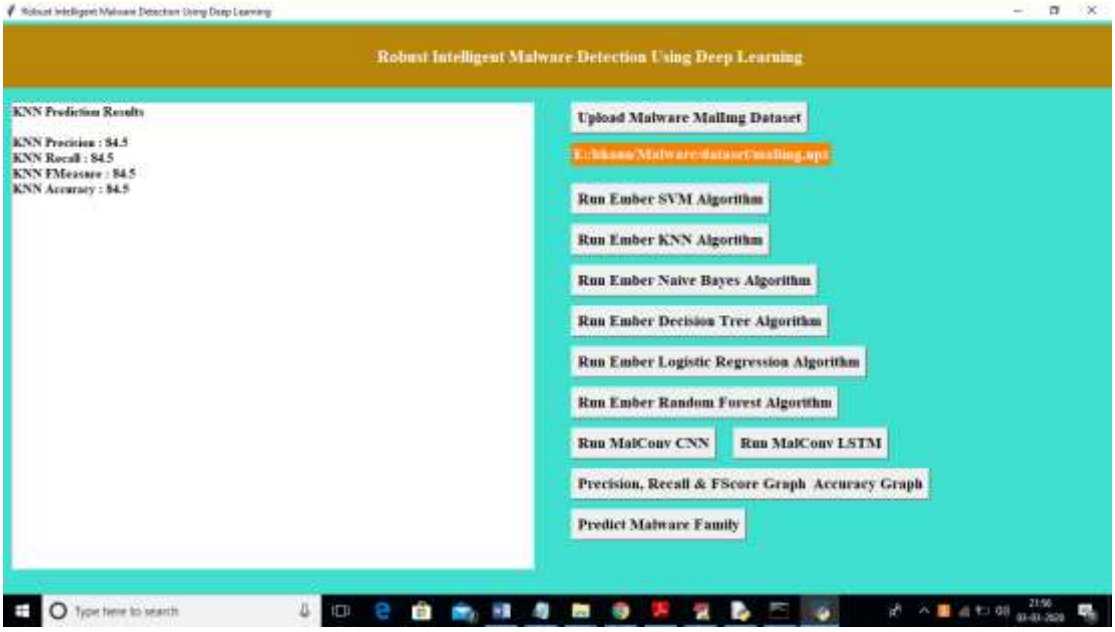

Fig 10 KNN Prediction In above screen we got KNN details and now click on 'Naïve Bayes' button to get its performance details

Industrial Engineering Journal
ISSN: 0970-2555
Volume : 53, Issue 4, April : 2024

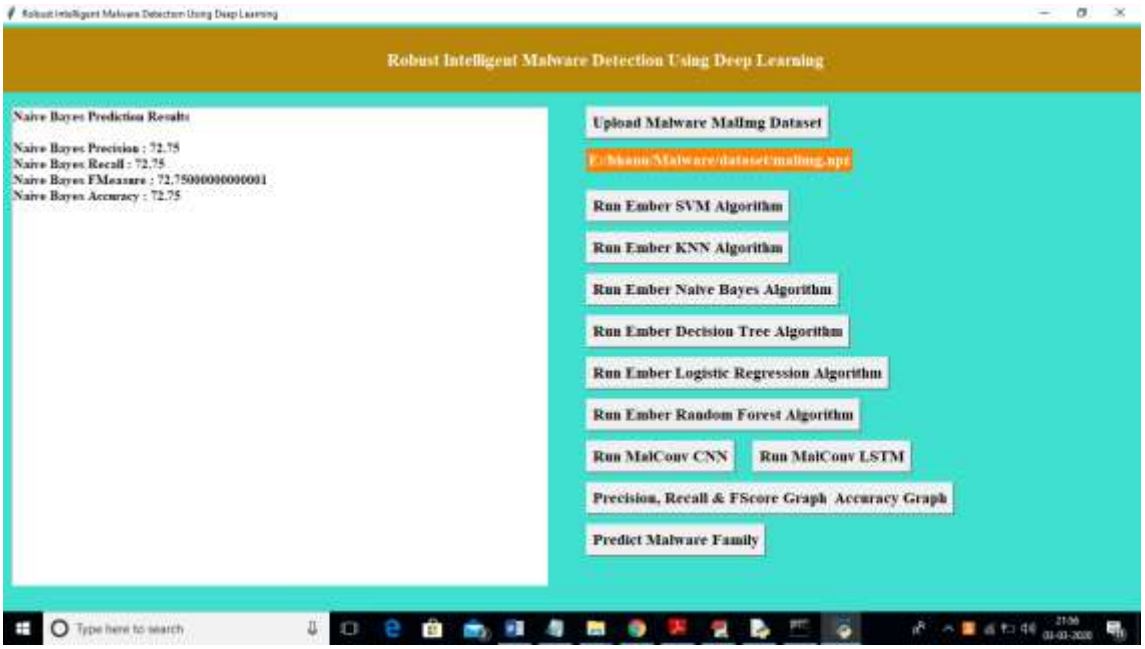

Fig 11 Naïve Bayes Prediction In above screen we got naïve bayes details and now click on 'Decision Tree' button to get its performance details

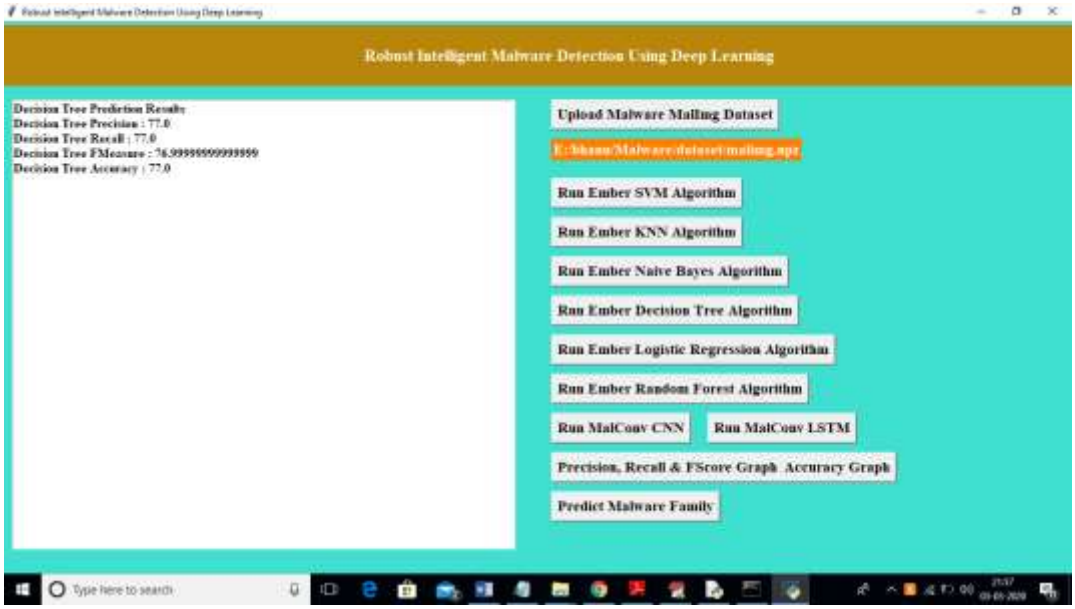

Fig 12 Decision Tree Prediction In above screen we got decision tree details and now click on 'Logistic Regression' button to get its details

Industrial Engineering Journal
ISSN: 0970-2555
Volume : 53, Issue 4, April : 2024

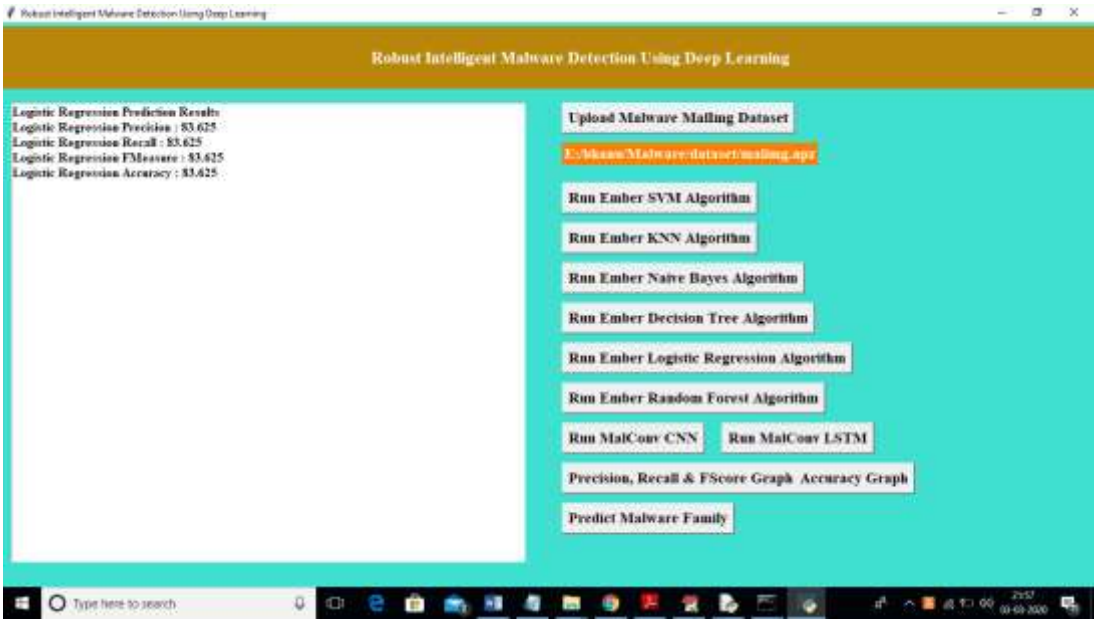

Fig 13 Logistic Regression Prediction In above screen we got logistic regression details and now click on 'Run Random Forest' button to get its performance

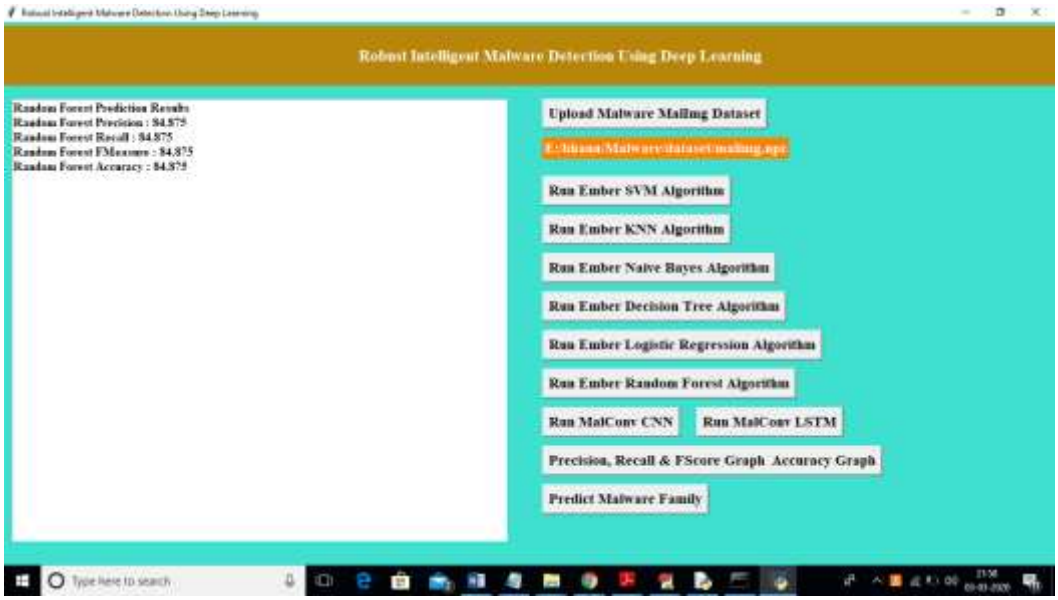

Fig 14 Random Forest Prediction In above screen we got random forest details and now click on 'Run MalConv CNN' button to get its performance details. CNN may take 10 minutes to complete execution and u can check its ongoing processing in black console

Industrial Engineering Journal
ISSN: 0970-2555
Volume : 53, Issue 4, April : 2024

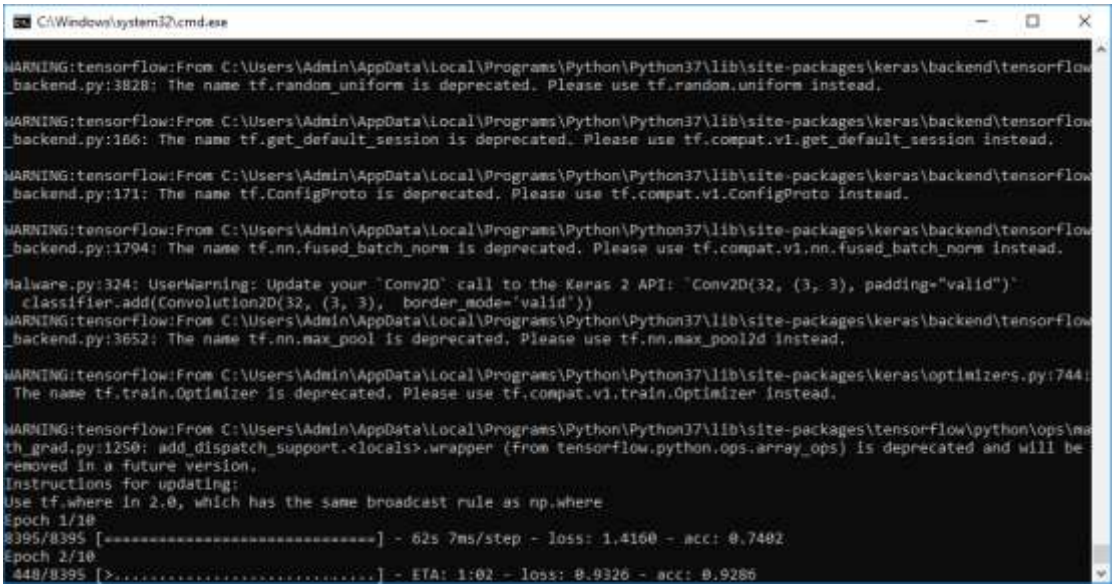

Fig 15 :- In above console it will take 10 epochs iteration and for each iteration it calculate accuracy for 8395 Malicious data. So u need to wait till all 10 epochs completed then u will get its performance details

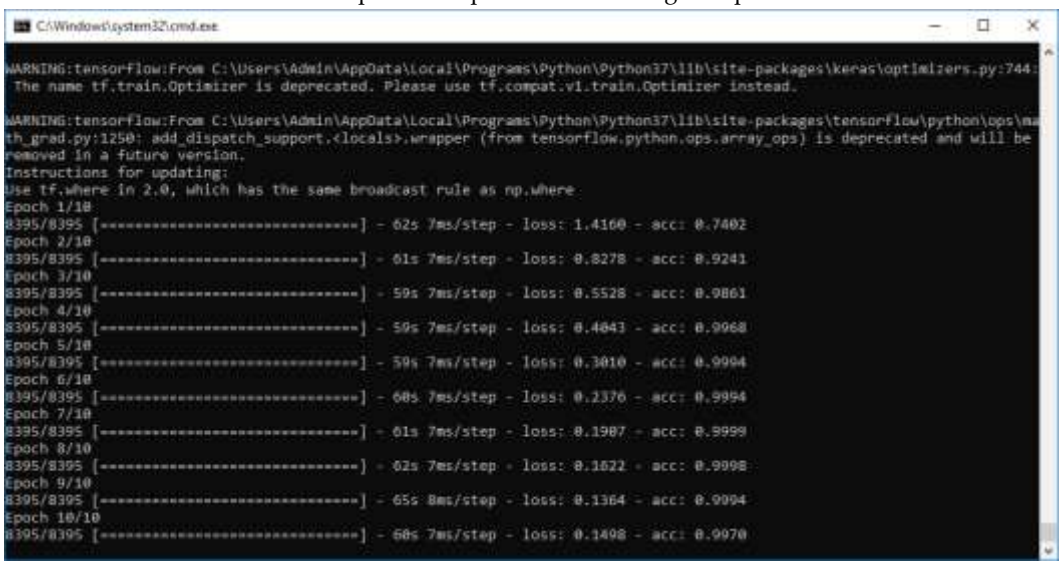

Fig 16 :- In above screen we can see CNN complete all 10 epochs and after that we will get accuracy details in main screen

Industrial Engineering Journal
ISSN: 0970-2555
Volume : 53, Issue 4, April : 2024

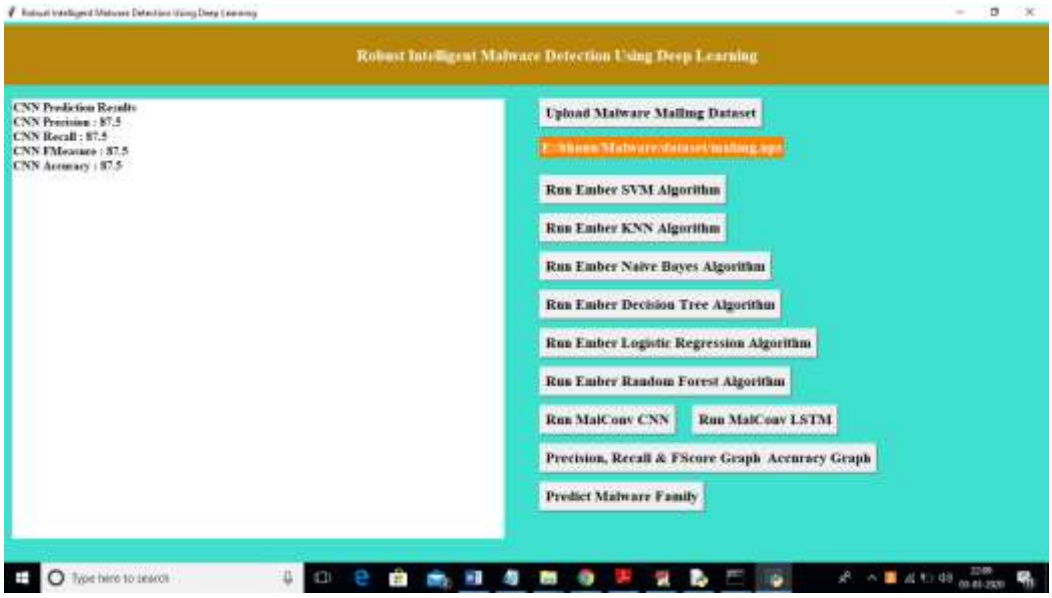

Fig 17:- CNN Prediction

Industrial Engineering Journal
ISSN: 0970-2555
Volume : 53, Issue 4, April : 2024

Graph Representation 1

Graph Representation 2

In above screen we can see precision graph for all algorithms and CNN get better performance. In above graph x-axis represents algorithm name and y-axis represents precision value

6. CONCLUSION AND FUTURE SCOPE

6.1 CONCLUSION

The aim of this paper is to present a machine learning approach to the Malicious problem. Due to the sudden growth of Malicious, we need automatic methods to detect infested files. In the first phase of the work, the data set is created using infested and clean executables, in order to extract the data necessary for the creation of the data set, we used a script created in Python. After creating the data set, it must be ready to train machine learning algorithms. The algorithms used are: decision trees, Random Forest, Naïve Bayes, GradientBoost and ADABoost presented comparatively. After applying the best accuracy algorithms, it had an Random Forest algorithm with an accuracy of 99.406012 %. This work demonstrates that Random Forest is the best algorithm for detecting malicious programs. In the future, this accuracy can be improved, if we add a much larger number of files in the data set to drive the algorithms. Each algorithm has several parameters that can be tested with different values to increase their accuracy. This project can reach the application level with the help of a library called pickle, to save what the algorithm has learned and then we can test a new file to see if it is clean or infected. Static analysis has also proven to be safer and free from the overhead of execution time.

6.2 FUTURE SCOPE

- This can be made more accurate with adding more data set

- More algorithms with better performance can add on to accuracy
- It can be hosted on web for real time analysis of exe files on the cloud

7 REFERENCES

1. Malicious Types and Classifications, Bert Rankin, 28.03.2018, published in LastLine, last accessed 12.09.2018.

2. A Brief History of Malicious - Its Evolution and Impact, Bert Rankin, 05.04.2018, published in LastLine, last accessed 12.09.2018.

3. Detecting Malicious through static and dynamic techniques, Jeremy Scott, 14.09.2017, published in NTT Security, last accessed 12.09.2018.

4. Hybrid Analysis and Control of Malicious, Kevin A. Roundy and Barton P. Miller, International Workshop on Recent Advances in Intrusion Detection, pp. 317-338, 2010, Springer.

5. Advanced Malicious Detection - Signatures Vs. Behavior Analysis John Cloonan Director of Products, Lastline, 11.04.2017, published in Infosecurity Magazine, last accessed 12.09.2018.

6. What is Machine Learning? Daniel Faggella, 12.08.2017, published in techemergence, last accessed 12.09.2018.

7. Data mining, Margaret Rouse, Search SQL Server last accessed 12.09.2018, the article can be found here. https://searchsqlserver.techtarget.com/definition/data- mining [8]

8. Supervised and Unsupervised Machine Learning Algorithms, Jason Brownlee, 16.03.2016, published in Machine Learning Algorithms, last accessed 12.09.2018.

9. Decision trees, scikit-learn.org last accessed 12.09.2018.