Machine Learning Engineer Nanodegree

Capstone Proposal

Menna Essa
February 6th 2019

Proposal

1. Domain Background

Information has always been a challenge in technology as users always download programs from untrusted sources ; PE format (portable executable) is the main file type for executables on windows operating system ; to fight this , different malicious behavior detection technologies have evolved throughout the years , starting from simple hash comparisons all the way to emulation and sand-boxed environments.
legacy techniques like hash comparisons and regular expressions have proven to be inefficient against modern malware which now can evade them using polymorphic code , so attention was shifted towards dynamic techniques , however these techniques are resource exhaustive and can pose a risk as they involve running the suspicious executable.
Finally, with the evolution of machine learning algorithms , information security research began utilizing them to gain a heuristic insight of the behavior of the target , heuristic analysis appeared as middle ground between static and dynamic analysis and helps cut the cost of needlessly running dynamic analysis on non-suspicious samples.

2. Problem Statement

The problem is the PE files can be very big and complex , there are also many anti-analysis techniques that are employed by the threat actor that makes recognizing the executable’s behavior without running it a challenging and sometimes impossible task.
However , Some times just the Headers of the PE file can have indicators that we are looking at a malicious/suspicious file ; In fact , the PE header is usually the first thing an analyst looks at before analyzing the actual code in the binary.
It would be interesting to analyze a mass of header data , discover relevant fields and eventually use them to predict whether a sample is malicious or not.

A paper was published exploring this here : https://arxiv.org/pdf/1709.01471.pdf

3. Datasets and Inputs

I found a PE labeled (malicious/benign) PE header dataset on Kaggle : https://www.kaggle.com/amauricio/pe-files-malwares

The data set covers all the fields in the headers , some fields are repetitive and quite irrelevant so the data will be processed to find fields that seem to capture relevant and irrelevant fields , PCA and further Supervised learning algorithms should help in discovering relevant information that is connected to the maliciousness of a file , there are currently 77 input fields shown in the link above , most of the features are continuous only 4 features are categorical (subsystem , dll_characteristics , LoaderFlags , Machine)
The dataset has a total of 19611 entries , 14599 of them are malware and 5012 of the are not.

The dataset is obviously not balanced , so In training I will recreate dataset with randomly selected malware labeled entries and use 75% of the clean entries and use the remaining entries in validation ; for final testing there is a separate testing dataset provided my the dataset author.

4.Solution Statement

The goal is to find the optimal model for this classification problem , primary candidates are Ensemble decision trees , Naive Bayes and MLP neural network which will be used after data analysis and preprocessing , I also want to explore the effect of PCA (principal component analysis) on the dataset and whether or not it improves performance on this particular problem.

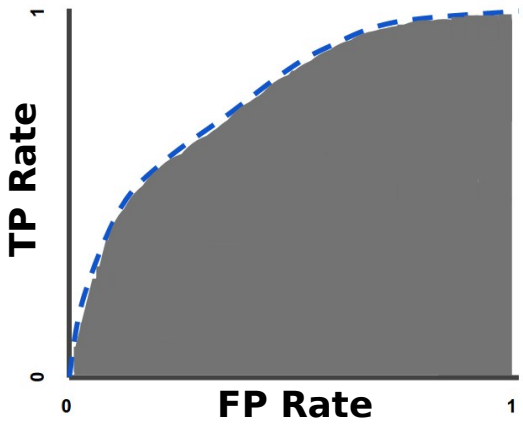
5.Benchmark Model

I will use logistic regression as a benchmark model to use the header fields to predict the label malicious (0 /1) , It is relevant because the problem is a good fit for supervised learning

6.Evaluation Metric

For both benchmark and final model Accuracy and F1-Score will be used to measure the performance of the model

AUC "Area under the ROC Curve." : measures the two-dimensional area underneath the ROC curve (0,0) to (1,1) [resource]



F1-Score is the harmonic mean of both precision and recall

F1 = 2 x (precision x recall) / (precision + recall)

7.Project Design

step 1 : data exploration and preprocessing
In the first part of the project , I will explore the data set , look for irrelevant fields and process sparse and skewed fields , this is important to facilitate the work of the algorithms , also some fields are skewed so I may have to apply a log function to them and normalize other columns so stay with [0,1] range.

Step2 : Train the supervised models [ensemble decision trees , Naive Bayes] and iterate to optimize , after making sure the data is clean , I will start training the models on the dataset and optimize them with Grid search cross validation to make sure I found the optimal hyper parameters.

Step3: Design a simple MLP neural network , and check if it yields a more accurate performance , explore if it is worth optimizing to beat supervised learning models.

Step4 : final accuracy will be calculated on a test set also provided by kaggle.

Step5 : perfom PCA analysis then retrain the optimal model and observe results.