The Efficiency of machine learning Techniques in minimizing Dwell Time of Cyberattacks

Abstract

The goal of this project is to minimize the dwell time of cyberattacks. The training/testing dataset are crafted in certain way to suit the project problem. The best models with highest precision score are identified by tuning the hyper parameters of three algorithms. Eventually, the predictions of the best models of unseen test datasets are evaluated using recall and precision metrics . The conclusion is all the built best models are not performing well in detecting the cyberattack early.

Design:

Cyberattacks Dwell Time is the time between when a compromise first occurs and when it is detected. CrowdStrike Services Report Shows that Average dwell time grew 10 days to 95 in 2019, up from 85 in 2018. According to attivonetworks.com in 2020 the problem became more worse with notable jump of Cyberattacks Dwell Time to 100 days. Advanced adversaries and state-sponsored threat actors (APT) are constantly modify their tactics and techniques to evade the detection techniques that allow them to remain undetected for a long time. This project tries to answer the below question:

Can the machine learning Techniques efficiently detect the emergent APT techniques early to minimize the Dwell Time of cyberattacks?

Data:

Raw Botnet dataset that created by Canadian Institute for Cybersecurity is chosen. The raw dataset is in PCAP format and divided into training and test datasets that included 7 and 16 types of botnets respectively.

The extracted training and testing datasets are crafted to suit the project problem through the following steps:

- Bidirectional 82 Flow features are extracted for training and testing datasets.
- The extracted training dataset consists of 356156 data instances, the target feature has two labels: Botnet which consists of 131946 data instances (representing the behavior of seven botnet types). Background (Normal) which consists of 224210 data instances.
- The initial extracted test dataset consists of 16 botnet types , all those types are dropped except one (Weasel Botnet) which represents the emergent cyberattack (not seen by model during training time, I assume that known cyberattack and emergent one share some pattern that can be detected by machine learning model).

- The goal of this project is to detect the cyberattack as early as possible in attack period which 10 days. The test dataset is spliced into 10 sub-datasets representing the accumulated cyberattack behavior on that day as illustrated in the below table.

Test dataset	Total Instances	Botnet Instances	Normal Instances
Day 0	113545	0	113545
Day 1	122095	8550	113545
Day 2	130645	17100	113545
Day 3	139195	25650	113545
Day 4	147745	34200	113545
Day 5	156295	42750	113545
Day 6	164845	51300	113545
Day 7	173395	59850	113545
Day 8	181945	68400	113545
Day 9	190495	76950	113545
Day 10	199045	85500	113545

Algorithms:

Actually, machine leaning is a workflow consisting of problem formulation, data preparation & cleaning, model selection & validation, Model evaluation using unseen data, and at the end model deployment. The problem formulation is covered in the design section. This project does not contain model deployment. In the

following subsections I will explain how data preparation & cleaning , model selection & validation, and Model evaluation using unseen data is implemented in this project.

Data Preparation and Cleaning:

The Exploratory Data Analysis (EDA) shows that some features are highly correlated, and some features are constant. The highly correlated features one of them is dropped. All constant features are dropped. After dropping features, the features number is reduced 30 features.

All the features are order in descending order based on feature-target mutual information because that will help algorithm (like decision tree) in identifying relevant features easily.

Model Selection & Validation:

The hyperparameters of Decision Tree (max_depth, min_samples_leaf , criterion), Balanced Random Forest (max_depth, criterion), and Logistic regression (C) are tuned by applying cross validation (K=3) randomized search on training dataset to find the

best model that score high precision. At the end of hyperparameter tuning, the best models are fitted with whole training dataset.

Among the three best models the highest precision score (91.79) is achieved by balanced random forest, while the lowest precision score (75.86) is achieved by logistic regression model. The decision tree model precision score is (84.64).

Model evaluation:

The predictions of each best model for each unseen test sub-dataset (day 1 - 10) are evaluated using recall ,precision metrics.

The recall and precision metrics (more details in presentation documents) shows that the three model are not performing well for early detection of cyberattacks.

Tool:

The below python packages will be used:

Pandas for data analysis and preparation

Matplotlib, Seaborn and Excel for data visualization

scikits.learn and imblearn for model fitting and evaluation

Communication:

The project details including the visualization are communicated through presentation document