Detectando Malwares com Técnicas Baseadas em Machine Learning

Gian Giovanni Rodrigues da Silva

Jessyca Jordanna Barroso de Moraes

Tammy Hikari Yanai Gusmão

Thalita Naiara Andre Alves

O problema

INTRODUÇÃO

- Malwares: ameaças mais eficazes na área de cibersegurança;
- Modificar e aumentar a complexidade de códigos maliciosos objetivando explorar falhas na segurança de sistemas.

CONTEXTO

- Malwares são comumente utilizados para desencadear um amplo escopo de ataques à segurança;
- Exemplos de malwares:
  - Vírus, Trojan Horse, Spyware, Worm.

O problema

DESCRIÇÃO

- Urgente necessidade de métodos de detecção de malware;
- Aplicar modelos de Machine Learning (ML), para podem ser de grande valor para a segurança de sistemas.

OBJETIVOS

- Geral:
  - Comparar performances de diferentes técnicas de ML, para detectar malwares;
- Específicos:
  - Selecionar técnicas para serem utilizadas no contexto do trabalho.
  - Comparar e analisar os resultados obtidos das técnicas utilizadas.

Metodologia

DATASET

- 100.000 dados de observação;
- 35 atributos;
- 50.000 classificados como malware e o restante como benign (benigno).

FEATURES

- Calcular a correlação entre todas as variáveis do dataset;
- Mapa de calor de correlação de Pearson;
- Correlação fraca ou forte e positiva ou negativa.

TRATAMENTO

- Discretizar a variável target em 0 (*benign*) e 1 (*malware*);
- Dividir o dataset em 70% de observações (70.000) para aprendizado e 30% para teste (30.000);
- Aplicando os modelos KNN, Decision Tree e Random Forest.

Técnicas de Machine Learning

KNN

- Algoritmo de aprendizagem de máquina supervisionado;
- Definição de *k* para aplicação da métrica de similaridade;
  - Distância Euclidiana
- A capacidade de predição não vem do aprendizado.

Decision Tree

- Algoritmo de aprendizagem de máquina supervisionado;
- Baseado na divisão dos dados em grupos similares;
- Problemas de classificação quanto de regressão;
- Tende a sofrer sobreajuste.

Random Forest

- Forma de resolver o problema de sobreajuste das árvores de decisão;
- Define essencialmente uma coleção de árvores de decisão;
- Cada árvore pode fazer um bom trabalho de previsão;
- Cálculo da média dos seus resultados.

Métricas de classificação

Acurácia

- Cálculo da divisão do total de acertos (positivos e negativos);
- Datasets balanceados.

Precisão

- Divisão entre o número de exemplos classificados corretamente e a soma desse número com o total de exemplos classificados erroneamente;
- Dentre todas as classificações da classe Positivo que o modelo fez, quantas estão corretas.

Sensibilidade

- Proporção ou porcentagem de classificações positivas que foram identificadas corretamente;
- É mais indicada para uma situação em que os falsos negativos são considerados mais prejudiciais que os falsos positivos.

Métricas de Classificação

F1-Score

- Média harmônica entre precisão e sensibilidade;
- F1-Score baixo é um indicativo de que ou a precisão ou a sensibilidade está baixa;
- Mais indicado para *datasets* com classes proporcionais e para um modelo que não emite probabilidades.

Observação: Foram aplicados diferentes valores nos parâmetros *macro* e *weighted average* das métricas de precisão, sensibilidade e f1-score.

Resultados

Método KNN

KNN:

	precision	recall	f1-score	support
benign	0.99180	0.98870	0.99025	15050
malware	0.98866	0.99177	0.99022	14950
accuracy			0.99023	30000
macro avg	0.99023	0.99024	0.99023	30000
weighted avg	0.99024	0.99023	0.99023	30000

Método Decision Tree

Decision Tree:

	precision	recall	f1-score	support
benign	0.99615	0.99674	0.99645	15050
malware	0.99672	0.99612	0.99642	14950
accuracy			0.99643	30000
macro avg	0.99643	0.99643	0.99643	30000
weighted avg	0.99643	0.99643	0.99643	30000

Método Random Forest

Random Forest:

	precision	recall	f1-score	support
benign	0.99728	0.99701	0.99714	15050
malware	0.99699	0.99726	0.99712	14950
accuracy			0.99713	30000
macro avg	0.99713	0.99713	0.99713	30000
weighted avg	0.99713	0.99713	0.99713	30000

Conclusão

Foi observado que:

- Similaridade de comportamentos entre algumas métricas dos modelos e em seus parâmetros macro e weighted average.
- Para a detecção de malwares no dataset escolhido, o Random Forest teve o melhor desempenho em todos métricas utilizadas.

Obrigado!