Phòng thí nghiệm An toàn Thông tin InsecLab
Trường Đại học Công nghệ Thông tin (UIT)

BÁO CÁO THỰC HÀNH

Môn học: Phương pháp học máy trong an toàn thông tin

Tên chủ đề: Advanced Malware Detection

GVHD: Nguyễn Hữu Quyền

Nhóm: NT09

1. THÔNG TIN CHUNG:

(Liệt kê tất cả các thành viên trong nhóm)

Lớp: NT522.O21.ANTT.2

STT	Họ và tên	MSSV	Email
1	Nguyễn Triệu Thiên Bảo	21520155	21520155@gm.uit.edu.vn
2	Trần Lê Minh Ngọc	21521195	21521195@gm.uit.edu.vn

2. NỘI DUNG THỰC HIỆN:[1]

STT	Nội dung	Tình trạng
1	Yêu cầu 1	100%
2	Yêu cầu 2	100%
3	Yêu cầu 3	100%
4	Yêu cầu 4	100%
5	Yêu cầu 5	100%
6	Yêu cầu 6	100%
7	Yêu cầu 7	100%
8	Yêu cầu 8	100%

Phần bên dưới của báo cáo này là tài liệu báo cáo chi tiết của nhóm thực hiện.

[1] Ghi nội dung công việc, các kịch bản trong bài Thực hành

BÁO CÁO CHI TIẾT

Link dataset:

https://drive.google.com/drive/folders/1Vtyq4qmVbGGTy4J07cjYmjKLl-QA4Gwd

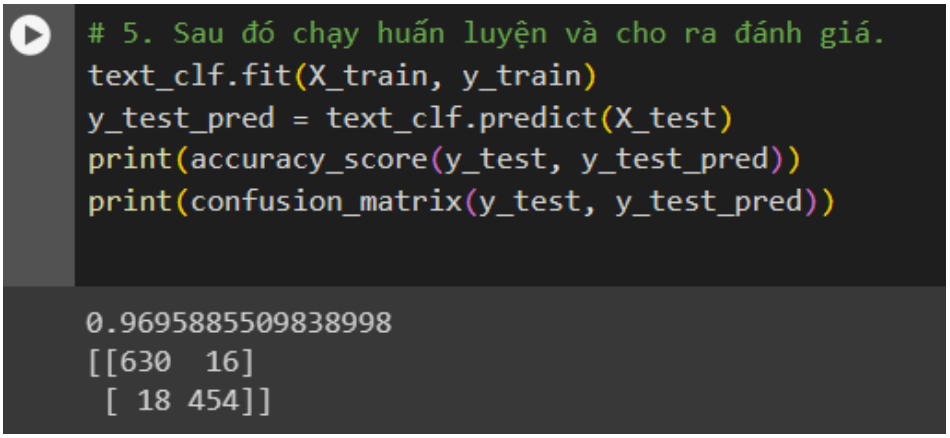
https://drive.google.com/drive/folders/17dTxpWISBSfupkTU2UlFohHs6MwuXxNc?usp=sharing

Link code bài tập:

https://colab.research.google.com/drive/16yZroGXk8rADaGtCPNGN2C4y8PhGxRSH?usp=sharing

1. Phát hiện Javascript bị rối mã

Bài tập 1: Cho biết kết quả accuracy và confusion matrix.



```
# 5. Sau đó chạy huấn luyện và cho ra đánh giá.
text_clf.fit(X_train, y_train)
y_test_pred = text_clf.predict(X_test)
print(accuracy_score(y_test, y_test_pred))
print(confusion_matrix(y_test, y_test_pred))
```

```
0.9695885509838998
[[630  16]
 [ 18 454]]
```

- Kết quả accuracy: 96.96%
- Confusion matrix:

$$\begin{bmatrix} 630 & 16 \\ 18 & 454 \end{bmatrix}$$

Giải thích

- Giá trị 630 đại diện cho số lượng mẫu thực tế thuộc nhóm 0 và được dự đoán đúng là thuộc nhóm 0 (True Negative - TN).
- Giá trị 454 đại diện cho số lượng mẫu thực tế thuộc nhóm 1 và được dự đoán đúng là thuộc nhóm 1 (True Positive - TP).
- Có 16 mẫu bị phân loại nhầm từ nhóm 0 sang nhóm 1 (False Positive - FP).
- Có 18 mẫu bị phân loại nhầm từ nhóm 1 sang nhóm 0 (False Negative - FN).

2. Trích xuất thuộc tính tập tin PDF

Bài tập 2: Cho biết kết quả vector X.

```
X = []
files = listdir(PDFs_path)
for file in files:
    file_path = PDFs_path + file
    X.append(PDF_to_FV(file_path))
print(X)
```

```
[[153, 153, 82, 82, 2, 2, 2, 7, 0, 0, 0, 0, 0, 0, 2, 0, 0, 0, 0, 0, 0], [1096, 1095, 1061, 1061, 0, 0, 2, 32, 0, 43, 0, 0, 1, 0, 1, 0, 0, 0, 0, 0
```

Với mỗi [] là kết quả trả về cho từng file PDF.

3. Trích xuất N-grams bằng cách sử dụng thuật toá hash-gram

Bài tập 3: Cho biết kết quả vector X.

```
# In kết quả
print(X)
```

```
[[3, 18, 0, 21, 1, 1, 47, 2, 4, 3, 2, 11, 0, 8, 6, 2, 1, 0, 4, 1, 15, 236, 23, 19, 1, 1, 0, 73, 13, 14, 2, 2, 8, 4, 40, 0, 4, 8, 1, 0, 7, 6, 1, 4
```

4. Xây dựng bộ phân loại động phần mềm độc hại

Bài tập 4: Cho biết kết quả đánh giá.

```
mi_pipeline.fit(X_train, y_train)
print("Training accuracy:")
print(mi_pipeline.score(X_train, y_train))
print("Testing accuracy:")
print(mi_pipeline.score(X_test, y_test))
```

```
Training accuracy:
0.8923631990378833
Testing accuracy:
0.8123872519542995
```

- Kết quả accuracy training: 89.24%
- Kết quả accuracy testing: 81.24%

5. MalConv – Quy trình áp dụng sâu cho phát hiện phần mềm độc hại PE

Bài tập 5: Cho biết kết quả đánh giá mô hình qua tập test

```
print(model.evaluate(X, Y))
```

```
3/3 [==============================] - 1s 183ms/step - loss: 0.1069 - acc: 0.9885
[0.10685260593891144, 0.9885057210922241]
```

- Giá trị đầu tiên 10.69% là giá trị mất mát (loss) trên dữ liệu kiểm tra.
- Giá trị thứ hai 98.85% là độ chính xác (accuracy) của mô hình trên dữ liệu kiểm tra.

6. Xử lý phần mềm độc hại packer

Bài tập 6: Cài đặt. UPX từ https://github.com/1.upx/upx/releases, và tiến hành đóng gói các tập tin pe tại Benign PE Samples UPX

Kết quả đóng gói thành công

```
(b'                    Ultimate Packer for eXecutables\n                              Copyright (C) 1996 - 2024\nUPX 4.2.3       Markus Oberhum
/content/drive/MyDrive/PPHM/Lab3/Benign PE Samples UPX/appidpolicyconverter.exe
(b'                    Ultimate Packer for eXecutables\n                              Copyright (C) 1996 - 2024\nUPX 4.2.3       Markus Oberhum
/content/drive/MyDrive/PPHM/Lab3/Benign PE Samples UPX/CloudStorageWizard.exe
(b'                    Ultimate Packer for eXecutables\n                              Copyright (C) 1996 - 2024\nUPX 4.2.3       Markus Oberhum
/content/drive/MyDrive/PPHM/Lab3/Benign PE Samples UPX/clip.exe
(b'                    Ultimate Packer for eXecutables\n                              Copyright (C) 1996 - 2024\nUPX 4.2.3       Markus Oberhum
/content/drive/MyDrive/PPHM/Lab3/Benign PE Samples UPX/CloudNotifications.exe
(b'                    Ultimate Packer for eXecutables\n                              Copyright (C) 1996 - 2024\nUPX 4.2.3       Markus Oberhum
/content/drive/MyDrive/PPHM/Lab3/Benign PE Samples UPX/cliconfg.exe
(b'                    Ultimate Packer for eXecutables\n                              Copyright (C) 1996 - 2024\nUPX 4.2.3       Markus Oberhum
/content/drive/MyDrive/PPHM/Lab3/Benign PE Samples UPX/CloudExperienceHostBroker.exe
(b'                    Ultimate Packer for eXecutables\n                              Copyright (C) 1996 - 2024\nUPX 4.2.3       Markus Oberhum
/content/drive/MyDrive/PPHM/Lab3/Benign PE Samples UPX/CIDiag.exe
(b'                    Ultimate Packer for eXecutables\n                              Copyright (C) 1996 - 2024\nUPX 4.2.3       Markus Oberhum
/content/drive/MyDrive/PPHM/Lab3/Benign PE Samples UPX/cipher.exe
(b'                    Ultimate Packer for eXecutables\n                              Copyright (C) 1996 - 2024\nUPX 4.2.3       Markus Oberhum
/content/drive/MyDrive/PPHM/Lab3/Benign PE Samples UPX/choice.exe
(b'                    Ultimate Packer for eXecutables\n                              Copyright (C) 1996 - 2024\nUPX 4.2.3       Markus Oberhum
```

7. Xây dựng bộ phân loại packer

Bài tập 7: Cho biết kết quả đánh giá.

```
# 9. Sử dụng bộ phân loại được đào tạo để dự đoán trên bộ test và đánh đánh giá hiệu suất bằng confusion matrix
y_pred = clf.predict(X_test)
from sklearn.metrics import confusion_matrix
confusion_matrix(y_test, y_pred)
```

```
array([[16, 45,  0],
       [53,  7,  0],
       [ 2,  0, 21]])
```

8. MalGAN – Tạo phần mềm độc hại

Bài tập 8: Cho biết kết quả đánh giá mẫu mới trong việc đánh lừa bộ nhận diện.

Kết quả nhận được (Lưu vào MalGan_output/adversarial_log.csv)

	A	B	C	D	E	F
1	filename	original score	file length	pad length	loss	predict score
2	c2wtshost.exe	0.5894867	2626	262	1	0.9999995828
3	ctfmon.exe	0.99071383	9728	972		
4	dcomcnfg.exe	0.9977375	10240	1024		
5	dsamain.exe	0.99997675	5048	504		
6	iisreset.exe	0.658685	8232	823		
7	inetinfo.exe	0.9937145	7238	723		
8	InetMgr6.exe	0.80603945	10202	1020		
9	instnm.exe	0.99357754	8704	870		
10	lpq.exe	0.9996595	4708	470		
11	lpr.exe	0.07787239	5668	566	1	0.8691348433
12	06a428dd5a543e67	0.9301745	64360	6436		
13	.exe	0.1692581	56224	5622	0.1692581028	0.1692581028
14	3_4.exe	0.7424586	60928	6092		
15	BOTBINARY.EXE	0.041258547	77824	2176	0.04125854746	0.04125854746
16	counter.exe	0.91256404	32256	3225		
17	MiniConfigBuilder.ex	0.49144137	13312	1331	0.4914413691	0.4914413691
18	UpdateCheck.exe	0.32340267	8192	819	0.9552101493	1
19	win32.exe	0.15605617	24960	2496	0.1560561657	0.1560561657
20	win33.exe	0.47826192	68096	6809	0.4782619178	0.4782619178
21	yfoye_dump.exe	0.43931308	36864	3686	0.4393130839	0.4393130839

Nhận xét: Kết quả đánh lừa bộ nhận diện với mẫu mới khá thấp.