

**ĐẠI HỌC QUỐC GIA TP.HCM
TRƯỜNG ĐẠI HỌC BÁCH KHOA
KHOA KHOA HỌC VÀ KỸ THUẬT MÁY TÍNH**



**BÁO CÁO
ĐỒ ÁN TỐT NGHIỆP
PHÁT HIỆN LÙA ĐẢO
BẰNG CÁC PHƯƠNG PHÁP HỌC MÁY**

Ngành: Khoa học Máy tính

HỘI ĐỒNG: Hội đồng 18 Khoa học Máy tính

GVHD: TS. Nguyễn Lê Duy Lai

GVPB: TS. Nguyễn Đức Thái

—o0o—

SVTH1: Phạm Đại Hoàng An – 1912539

SVTH2: Nguyễn Đăng Hải – 1913254

TP. HỒ CHÍ MINH, THÁNG 06/2023

TRƯỜNG ĐẠI HỌC BÁCH KHOA

KHOA: Khoa học và Kỹ thuật Máy tính
BỘ MÔN: Hệ thống và Mạng

NHIỆM VỤ LUẬN ÁN TỐT NGHIỆP

Chú ý: Sinh viên phải dán tờ này vào trang nhất của bản thuyết trình

HỌ VÀ TÊN:

Phạm Đại Hoàng An

MSSV: 1912539

HỌ VÀ TÊN:

Nguyễn Đăng Hải

MSSV: 1913254

NGÀNH:

Khoa học Máy tính

LỚP:

1. Đầu đề luận án:

(Tiếng Việt): Phát hiện lừa đảo dùng các phương pháp máy học.

(English): Phishing Detection using Machine Learning Methods.

2. Nhiệm vụ (yêu cầu về nội dung và số liệu ban đầu):

Mục tiêu của đề tài: phát triển phần mềm phát hiện các trang web được tạo để lừa đảo bằng cách sử dụng các kỹ thuật học máy.

- Nghiên cứu về phát hiện lừa đảo: các đặc điểm chính của các trang web không chính thức (với mục đích đánh cắp thông tin người dùng); tại sao người dùng trở thành nạn nhân của các cuộc tấn công lừa đảo (xác định các nguyên nhân chính); xác định những thách thức hiện tại của các cuộc tấn công lừa đảo.
- Nghiên cứu về các kỹ thuật Trí tuệ nhân tạo (AI): Học máy, Học sâu, Học kết hợp và các kỹ thuật dựa trên kịch bản để phát hiện các cuộc tấn công lừa đảo. So sánh các nghiên cứu khác nhau nhằm phát hiện các cuộc tấn công lừa đảo cho từng kỹ thuật AI và xem xét các tính năng và thiếu sót nổi bật của các phương pháp này.
- Trích xuất và xử lý các đặc điểm từ các trang web (cả chính thức và có ý định xấu) để mô hình hóa và huấn luyện máy học.
- Thu thập và phân loại các bộ dữ liệu được sử dụng để đào tạo và thử nghiệm.
- Thử nghiệm nhiều mô hình máy học và đánh giá kết quả để xem xét mô hình tối ưu.
- Thiết kế hệ thống ứng dụng theo quy trình phát triển phần mềm.
- Xây dựng ứng dụng và chạy bản demo.

3. Ngày giao nhiệm vụ luận án: ___ / ___ / ___.

4. Ngày hoàn thành nhiệm vụ: ___ / ___ / ___.

5. Họ tên giảng viên hướng dẫn:

Phàn hướng dẫn:

1) T.S Nguyễn Lê Duy Lai

100%

Nội dung và yêu cầu LVTN đã được thông qua Bộ môn.

Ngày tháng năm

CHỦ NHIỆM BỘ MÔN

(Ký và ghi rõ họ tên)

GIẢNG VIÊN HƯỚNG DẪN CHÍNH

(Ký và ghi rõ họ tên)

PHẦN DÀNH CHO KHOA, BỘ MÔN:

Người duyệt (chấm sơ bộ): _____

Đơn vị: _____

Ngày bảo vệ: _____

Điểm tổng kết: _____

Nơi lưu trữ luận án: _____

Nguyễn Lê Duy Lai

Ngày tháng năm 2023.

PHIẾU CHẤM BẢO VỆ LVTN

(Dành cho người hướng dẫn/phản biện)

1. Họ và tên SV: **Phạm Đại Hoàng An** (1912539), **Nguyễn Đăng Hải** (1913254)
MSSV: _____
Ngành (chuyên ngành): Khoa học máy tính

2. Đề tài: **Phát hiện lừa đảo bằng phương pháp máy học**
3. Họ tên người hướng dẫn/phản biện: **TS. Nguyễn Lê Duy Lai**
4. Tổng quát về bản thuyết minh:

Số trang: _____ Số chương: _____
Số bảng số liệu: _____ Số hình vẽ: _____
Số tài liệu tham khảo: _____ Phần mềm tính toán:
Hiện vật (sản phẩm): _____

5. Tổng quát về các bản vẽ:
- Số bản vẽ: _____ Bản A1: _____ Khô khắc:
- Số bản vẽ vẽ tay: _____ Số bản vẽ trên máy tính: _____

6. Những ưu điểm chính của LVTN:
Có ba đóng góp chính của luận văn như sau:

- Luận văn thể hiện ý tưởng giúp người dùng cuối phát hiện Website lừa đảo bằng cách tạo tiện ích mở rộng tích hợp vào trình duyệt Web, phân tích các sự kiện/dữ liệu đáng ngờ bằng các mô hình dựa trên quy tắc phù hợp, sau đó chuyển thông tin Website đáng ngờ về máy chủ chuyên dụng để phân tích chuyên sâu và gán nhãn.
- Luận văn khảo sát các phương pháp học máy cỏ điên khác nhau (Logistic Regression, Random Forest, SVM, v.v.) và cả các phương pháp lai, đồng thời đánh giá hiệu suất của chúng để đưa ra các phương pháp phù hợp nhất cho mô hình tập hợp (Ensemble Learning). Các mô hình được cải tiến và tối ưu hóa để đạt được kết quả tốt hơn.
- Luận văn này cũng đã tiến hành đánh giá chi tiết hơn để cho thấy hiệu suất của phương pháp đề xuất và hiệu quả của Hệ thống phát hiện lừa đảo.

7. Những thiếu sót chính của LVTN:

- Bộ dữ liệu rất hạn chế nhưng hệ thống có thể tự mở rộng bằng cách bổ sung thêm mẫu có gán nhãn.
- So sánh kết quả với các dự án khác làm việc trên cùng một bộ dữ liệu.

8. Đề nghị: Được bảo vệ Bổ sung thêm để bảo vệ Không được bảo vệ
9. 3 câu hỏi SV phải trả lời trước Hội đồng:

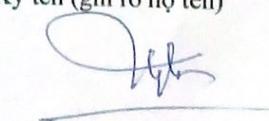
a.

b.

c.

10. Đánh giá chung (bằng chữ: giỏi, khá, TB): Điểm : **9.0** /10

Ký tên (ghi rõ họ tên)



Nguyễn Lê Duy Lai

Ngày 02 tháng 06 năm 2023

PHIẾU CHẤM BẢO VỆ LVTN

(Dành cho người phản biện)

1. Họ và tên SV: Phạm Đại Hoàng An – Student ID: 1912539
Nguyễn Đăng Hải – Student ID: 1913254

Ngành (chuyên ngành): Khoa học Máy tính

2. Đề tài: Phát hiện lừa đảo bằng các phương pháp học máy.

3. Họ tên người phản biện: Nguyễn Đức Thái

4. Tổng quát về bản thuyết minh:

Số trang: 71

Số chương: 5

Số bảng số liệu: 9

Số hình vẽ: 51

Số tài liệu tham khảo: 16

Phần mềm tính toán:

Hiện vật (sản phẩm)

5. Tổng quát về các bản vẽ:

- Số bản vẽ:

Bản A1:

Bản A2:

Khô khác:

- Số bản vẽ vẽ tay

Số bản vẽ trên máy tính:

6. Những ưu điểm chính của LVTN:

- Sinh viên hiểu và áp dụng được các phương pháp học máy vào việc phát hiện lừa đảo.
- Sinh viên đã xây dựng được mô hình phân loại các website giả mạo, so sánh và đánh giá các mô hình này để tìm ra mô hình phù hợp nhất.
- Xây dựng được extension trên trình duyệt Chrome giúp cảnh báo người dùng phát hiện website giả mạo.

7. Những thiếu sót chính của LVTN:

- Sinh viên chưa lý giải được giá trị độ chính xác trong việc phát hiện lừa đảo, chưa đánh giá và thảo luận độ chính xác này.
- Sinh viên chưa lý giải được làm cách nào để đạt được kết quả (độ chính xác) trong việc phát hiện lừa đảo.

8. Đề nghị: Được bảo vệ

- Bổ sung thêm để bảo vệ

- Không được bảo vệ

9. 3 câu hỏi SV phải trả lời trước Hội đồng:

- a. Làm sao tính được độ chính xác trong việc phát hiện phishing?

- b. Độ chính xác được trình bày trong báo cáo luận văn là khá cao so với bình thường, đề nghị sinh viên giải thích.

- c. Ngoài ra, đề nghị SV giải thích các trường hợp phát hiện sai, tỷ lệ phát hiện sai là bao nhiêu?

10. Dánh giá chung (bằng chữ: giỏi, khá, TB): Giỏi

Điểm : 9/10

Ký tên (ghi rõ họ tên)

Nguyễn Đức Thái



Lời cam đoan

Chúng tôi xin cam đoan bài báo cáo Đồ án tốt nghiệp này là công trình nghiên cứu của chúng tôi dưới sự hướng dẫn và giám sát của TS. Nguyễn Lê Duy Lai. Mọi điều được ghi trong báo cáo và mã nguồn là do chúng tôi thực hiện - ngoại trừ những kiến thức tham khảo và mã nguồn mẫu được trích dẫn được cung cấp, hoàn toàn không sao chép từ bất cứ nguồn khác. Các số liệu và kiến thức tham khảo dùng để nhận xét, phân tích được nhóm thu thập từ nhiều nguồn và sẽ ghi rõ trong phần Tài liệu tham khảo. Nhóm xin chịu trách nhiệm trước Khoa và Nhà trường nếu lời cam đoan trái với sự thật.

Nhóm sinh viên thực hiện đề tài



Lời cảm ơn

Đầu tiên, chúng tôi xin gửi lời cảm ơn sâu sắc tới thầy giáo TS. Nguyễn Lê Duy Lai đã tận tình hướng dẫn và giúp đỡ trong quá trình thực hiện Đồ án tốt nghiệp. Với sự hướng dẫn tận tình, chu đáo và những kiến thức mà thầy truyền đạt đã giúp chúng tôi hoàn thiện Đồ án tốt nghiệp lần này.

Ngoài ra, nhóm xin cảm ơn các thầy cô trường Đại học Bách Khoa - Đại học Quốc gia Thành phố Hồ Chí Minh, đặc biệt là các thầy cô của khoa Khoa học và Kỹ thuật Máy tính đã truyền đạt những kiến thức quý báu, đã giúp chúng tôi có những kiến thức nền tảng để thực hiện Đồ án tốt nghiệp này.

Nhóm cũng xin cảm ơn các anh, chị, bạn bè, những người đã giúp đỡ, góp ý, chia sẻ kiến thức và hỗ trợ chúng tôi trong suốt thời gian thực hiện Đồ án tốt nghiệp.

Đồ án tốt nghiệp của chúng tôi vẫn còn những hạn chế và thiếu sót trong khả năng nghiên cứu và triển khai. Chúng tôi trân trọng mọi ý kiến đóng góp từ phía hội đồng giảng viên và mọi người để chúng tôi có thể bổ sung kiến thức, hoàn thiện và cải thiện đề tài chất lượng hơn.

Nhóm sinh viên thực hiện đề tài

Tóm tắt

Trong những năm gần đây, sự phát triển của Internet và các thiết bị di động thông minh đã góp phần thúc đẩy sự bùng nổ của các trang mạng xã hội, trong đó Facebook là nền tảng được sử dụng rộng rãi nhất với hơn 1,4 tỷ người dùng trên toàn cầu và trên 30 triệu người dùng tại Việt Nam. Tuy nhiên, sự phát triển nhanh chóng này cũng đã tạo điều kiện thuận lợi cho các hoạt động lừa đảo và việc đánh cắp thông tin cá nhân người dùng đã trở thành vấn đề đáng quan ngại. Vì vậy, nhu cầu tăng cường bảo mật an ninh mạng ngày càng được coi trọng, đặc biệt là trong bối cảnh của sự phát triển mạnh mẽ của trí tuệ nhân tạo và công nghệ 4.0. Để giúp người dùng tránh khỏi những rủi ro này, một công cụ hỗ trợ có thể phát hiện các trang web giả mạo hay những tên lừa đảo sẽ giúp cho người dùng an tâm hơn trong việc truy cập Internet.

Học máy là một nhánh của trí tuệ nhân tạo, nói đến việc máy học tự động từ dữ liệu để giải quyết các vấn đề cụ thể. Với sự phát triển của kỹ thuật và nhiều kết quả đáng kỳ vọng trong việc xử lý dữ liệu, việc xây dựng một hệ thống phát hiện giả mạo thông qua việc trích xuất các đặc điểm chung của các trang web giả mạo là hoàn toàn khả thi. Các nguồn dữ liệu lớn về các trang web giả mạo và các email giả mạo đã được xử lý và có sẵn trên các nền tảng như Kaggle và PhishTank.

Trong Đồ án tốt nghiệp này, nhóm sẽ tiến hành xây dựng các mô hình học máy phát hiện các trang Web lừa đảo với thông qua nguồn dữ liệu lớn đã được xử lý. Từ đó, nhóm áp dụng các mô hình đã xây dựng vào việc phát triển tiện ích mở rộng trên Chrome nhằm hỗ trợ người dùng khi lướt Web hay truy cập vào một trang Web bất kỳ.

Ngoài việc cung cấp cho người dùng một công cụ hỗ trợ trong việc phát hiện các trang Web giả mạo, đề tài này còn đóng góp vào việc phát triển công nghệ an ninh mạng. Việc áp dụng học máy vào giải quyết vấn đề an ninh mạng sẽ là một xu hướng phát triển của công nghệ trong tương lai.

Tổng kết lại, đề tài này sẽ tiến hành nghiên cứu và xây dựng một hệ thống phát hiện các trang Web giả mạo thông qua việc sử dụng các kỹ thuật xử lý dữ liệu và mô hình học máy. Ngoài việc cung cấp cho người dùng một công cụ hỗ trợ trong việc phát hiện các trang Web giả mạo, đề tài này còn đóng góp vào việc phát triển công nghệ an ninh mạng, góp phần làm giảm thiểu các hoạt động lừa đảo trên mạng.

Mục lục

I	Tổng quan về đề tài	10
1	Giới thiệu về tấn công mạng	10
1.1	Tấn công lừa đảo qua mạng là gì?	10
1.2	Những thủ đoạn tấn công mạng	10
1.3	Cách thức tấn công của tội phạm mạng	11
2	Thực trạng và tính cấp thiết của đề tài	12
2.1	Thực trạng chung hiện nay	12
2.2	Thách thức hiện tại trong việc phòng chống tội phạm mạng	14
2.3	Tính cấp thiết của đề tài	15
3	Mục tiêu và giá trị khoa học mà đề tài này mang lại	17
3.1	Mục tiêu mà đề tài này hướng đến	17
3.2	Giá trị khoa học mà đề tài này mang lại	17
4	Kế hoạch thực hiện đồ án	18
II	Tổng quan về kiến thức sử dụng	20
1	Giới thiệu về bài toán phát hiện bất thường bằng phương pháp học máy .	20
2	Cách tiếp cận phát hiện sự bất thường bằng học máy	21
3	Các giải thuật sử dụng phát hiện bất thường	22
3.1	Giải thuật SVM	22
3.2	Giải thuật Logistic Regression	23
3.3	Giải thuật Decision Tree	25
3.4	Giải thuật KNN	26
3.5	Giải thuật Random Forest	27
4	Ensemble Learning	28
4.1	Max Voting	29
4.2	Stacked Generalization	29
4.3	Blending	30
4.4	Bagging Classifier	31
4.5	Gradient Boosting Classifier	32
4.6	Extreme Gradient Boosting	33
4.7	Light Gradient Boosting	33
5	Thông số đánh giá	33
5.1	Confusion Matrix	33
5.2	Accuracy	34

5.3	Precision	34
5.4	Recall	35
5.5	F1-score	35
6	Learning curve	35
7	Hệ số tương quan (Correlation coefficient)	37
8	Ngôn ngữ lập trình sử dụng	38
8.1	Python	38
8.2	HTML	38
8.3	CSS	38
8.4	Javascript	39
9	Các frameworks và thư viện sử dụng	39
9.1	Flask	39
9.2	Numpy	40
9.3	Pandas	40
9.4	Scikit-learn	40
9.5	Whois	41
9.6	Công cụ Google Colab	41
10	Azure App Service	41
III	Thiết kế xây dựng mô hình học máy	42
1	Mô hình Machine Learning phát hiện giả mạo	42
1.1	Đào tạo mô hình phân loại	42
1.2	Phát hiện giả mạo dựa trên mô hình phân loại	44
2	Mô tả dữ liệu	45
2.1	Tổng quan về tập dữ liệu	45
2.2	Phân loại các đặc trưng của trang web	46
2.3	Tầm quan trọng việc chọn lọc những đặc trưng	47
2.4	Những đặc tính quan trọng giúp phát hiện trang web giả mạo	47
3	Huấn luyện mô hình học máy	51
3.1	Hiệu năng của từng mô hình học máy	51
3.2	Thời gian xử lý của các mô hình học máy	53
3.3	Đồ thị học tập	54
IV	Thiết kế kiến trúc hệ thống phần mềm ứng dụng	55
1	Giới thiệu tổng quan về phần mềm	55
2	Lược đồ mô tả hệ thống	56
2.1	Lược đồ Use-case	56

2.2	Lược đồ Sequence	57
3	Kiến trúc hệ thống	59
3.1	Tổng quan về kiến trúc hệ thống	59
3.2	Tiện ích mở rộng (Extension)	60
3.3	Hệ thống máy chủ (Server)	62
3.3.1	Hệ thống trung gian (Intermediary Server)	62
3.3.2	Hệ thống dự đoán bằng học máy (Model Server)	62
4	Thiết kế UI hệ thống	63
V	Hiện thực và đánh giá hệ thống	65
1	Demo ứng dụng	65
2	Kiểm thử và đánh giá hệ thống	70
2.1	Kiểm thử hệ thống	70
2.2	Đánh giá hiệu năng hệ thống	71
2.3	Đánh giá và nhận xét từ người dùng tiện ích	73
VI	Tổng kết	78
1	Kết quả đạt được	78
2	Những hạn chế tồn tại	79
3	Hướng phát triển trong tương lai	79
TÀI LIỆU THAM KHẢO		81

Danh sách hình vẽ

1	Quy trình lừa đảo thông qua email và trang web giả mạo	11
2	Số lượng tấn công lừa đảo từ tháng 4/2021 đến tháng 3/2022	13
3	Các lĩnh vực bị tấn công trong quý 1 năm 2022	13
4	Tấn công bằng hình thức mã độc tống tiền vào các lĩnh vực trong quý 1 năm 2022	14
5	Biểu đồ nạn nhân là các công ty dựa trên doanh thu	15
6	Minh họa mục tiêu sản phẩm	17
7	Giản đồ Gantt thực hiện đề tài	18
8	Giản đồ Gantt thực hiện giai đoạn 1	19
9	Giản đồ Gantt thực hiện giai đoạn 2	19
10	Giản đồ Gantt thực hiện giai đoạn 3	20
11	Giản đồ Gantt thực hiện giai đoạn 4	20
12	Hyper-plane phân chia 2 lớp riêng biệt	22
13	Linear Regression VS Logistic Regression Graph	24
14	Giải thuật Stacking	30
15	Giải thuật Blending	31
16	Kỹ thuật Bagging	32
17	Kỹ thuật Gradient Boost	32
18	Confusion matrix cho phân loại nhị phân	34
19	Đồ thị học tập Naive Bayes và SVM	36
20	Ví dụ về hệ số tương quan	37
21	Quy trình huấn luyện mô hình, tham khảo [13]	43
22	Quy trình phát hiện giả mạo, tham khảo [13]	45
23	Quy trình xây dựng bộ dữ liệu	45
24	Sơ đồ phân loại các đặc tính	46
25	Ma trận tương quan của đặc trưng	48
26	So sánh đồ thị học tập của từng mô hình	55
27	Lược đồ Use-case của hệ thống	56
28	Lược đồ Sequence cho usecase cảnh báo người dùng	57
29	Lược đồ Sequence cho usecase kiểm tra website có phải giả mạo hay không	58
30	Lược đồ Sequence cho usecase báo cáo website giả mạo	59
31	Kiến trúc tổng quan của hệ thống phát hiện lừa đảo	59
32	Cấu trúc thư mục Extension	61
33	Giao diện app hiển thị kết quả	64

34	Giao diện app	64
35	Giao diện app khi xử lý xong	65
36	Giao diện tiện ích mở rộng khi đang đánh giá	66
37	Giao diện tiện ích mở rộng khi đánh giá là uy tín	66
38	Giao diện tiện ích mở rộng khi đánh giá là lừa đảo	67
39	Biểu tượng của extension được cập nhật để hiển thị là uy tín	67
40	Biểu tượng của extension được cập nhật để hiển thị là lừa đảo	68
41	Pop-up hiện lên để cảnh báo người dùng	68
42	Giao diện của extension khi kiểm tra website	69
43	Giao diện của extension báo cáo	69
44	Form đánh giá và nhận xét	73
45	Kết quả đánh giá về UI của ứng dụng	73
46	Kết quả đánh giá của người dùng về thời gian phản hồi	74
47	Kết quả đánh giá tính năng cảnh báo người dùng	75
48	Kết quả đánh giá tính năng kiểm tra trang web giả mạo	76
49	Kết quả đánh giá tính năng báo cáo website giả mạo	76
50	So sánh các tính năng với nhau	77
51	Kết quả người dùng đánh giá tính hữu ích của ứng dụng	78

Danh sách bảng

1	Cấu hình phần cứng Google Colab cung cấp	41
2	Các đặc trưng URL-based cho việc phát hiện giả mạo	49
3	Các đặc trưng Content-based cho việc phát hiện giả mạo	50
4	Các đặc trưng External-based cho việc phát hiện giả mạo	51
8	Các API của hệ thống	63
9	Kiểm thử tính năng cảnh báo người dùng	70
10	Kiểm thử tính năng kiểm tra website giả mạo hay không	71
11	Kiểm thử tính năng báo cáo website giả mạo	71
12	Kiểm tra tốc độ xử lý của hệ thống	72

I. Tổng quan về đề tài

1. Giới thiệu về tấn công mạng

1.1 Tấn công lừa đảo qua mạng là gì?

Tấn công lừa đảo qua mạng là hình thức tấn công mạng bằng việc xây dựng những hệ thống lừa đảo nhằm đánh cắp các thông tin nhạy cảm, như tên đăng nhập, mật khẩu hay thông tin về các loại thẻ tín dụng của người dùng. Nó xảy ra khi kẻ giả mạo xuất hiện như một thực thể đáng tin cậy, một trang thông tin điện tử, eBay, Paypal, Gmail, hay các ngân hàng trực tuyến là những mục tiêu hướng đến của hình thức tấn công này. Việc giả mạo thường diễn ra qua email và tin nhắn nhanh, nhằm đánh lừa người dùng để nhập thông tin vào một biểu mẫu hoặc nhấp vào một liên kết của một trang web lừa đảo.

Việc lừa đảo và đánh cắp thông tin như vậy thường xảy ra trong các công ty lớn hoặc các tổ chức chính phủ, nhằm làm suy yếu các tổ chức này hoặc chuẩn bị cho cuộc tấn công lớn hơn, được gọi là Tấn công có chủ đích (APT). Kẻ giả mạo cũng có thể phát tán phần mềm độc hại vào môi trường làm việc của công ty hoặc truy cập dữ liệu tối mật để bán cho các công ty đối thủ.

Một tổ chức bị tấn công như vậy thường phải chịu tổn thất tài chính nghiêm trọng, cùng với việc giảm thị phần, danh tiếng và sự tin tưởng của khách hàng. Phạm vi và mức độ của cuộc tấn công có thể leo thang thành một sự cố bảo mật, gây khó khăn cho doanh nghiệp trong việc khôi phục hoặc thậm chí dẫn đến phá sản hoàn toàn.

1.2 Những thủ đoạn tấn công mạng

Hình thức lừa đảo qua mạng Internet, mạng viễn thông và các ứng dụng trên mạng đã có dấu hiệu gia tăng trong thời gian gần đây, với mục đích chiếm đoạt tiền của người dân. Thông thường, kẻ xấu sẽ lợi dụng hai hình thức chính là email giả mạo (Phishing Email) và website giả mạo (Phishing Website) để đánh lừa người dùng.

Phishing Email

Trong hình thức Phishing Email, các hacker thường giả mạo một doanh nghiệp uy tín để thực hiện các kế hoạch lừa đảo trong email của người dùng. Họ tạo ra nội dung email nhằm lừa đảo người nhận, sử dụng các giao diện tương tự với các bên ngân hàng hoặc tổ chức tài chính khác. Mục tiêu của họ là làm cho người dùng tin tưởng và cung cấp thông tin tài khoản cá nhân hoặc truy cập vào các liên kết có chứa mã độc, nhằm xâm nhập vào hệ thống mạng doanh nghiệp.

Các hacker có thể tạo ra nội dung email gần giống với giao diện email thực tế của các ngân hàng, để lừa đảo người dùng và làm cho họ tin rằng đó là email chính thức từ các ngân hàng mà họ đang giao dịch. Khi người dùng tin tưởng, họ dễ dàng chia sẻ các thông

tin quan trọng như mật khẩu đăng nhập hệ thống, mật khẩu giao dịch, thông tin thẻ tín dụng và các thông tin tuyệt mật khác.

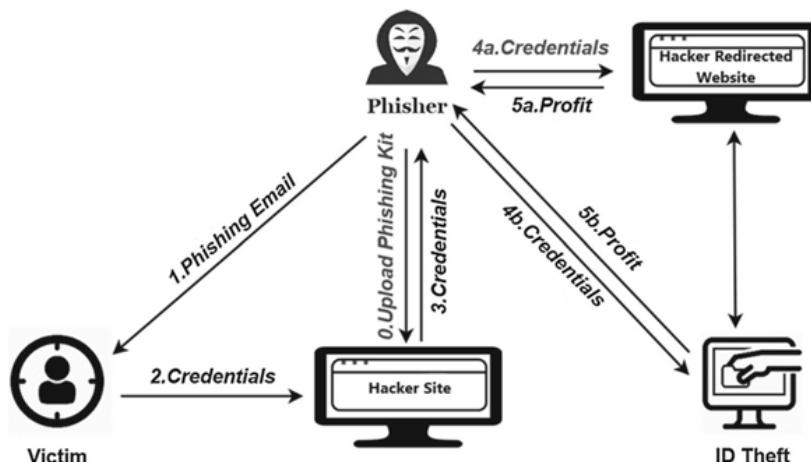
Phishing Website

Website là một tập hợp các trang thông tin chứa nội dung văn bản, số liệu, âm thanh, hình ảnh, video,... được lưu trữ trên máy chủ (web server) và truy cập từ xa qua Internet. Giả mạo website là hành vi tạo ra một trang web sử dụng tên, hình ảnh và địa chỉ của cá nhân, tổ chức để mạo danh và làm người dùng Internet nhầm lẫn trong việc nhận thông tin và tiến hành giao dịch.

Phishing Website là phương pháp thường được sử dụng để tấn công trên các trang mạng xã hội trực tuyến như Facebook, Twitter, TikTok,... Trong hình thức tấn công này, kẻ tấn công tạo ra các trang web bằng cách sao chép các trang web gốc và gửi URL đáng ngờ cho các nạn nhân thông qua tin nhắn rác, văn bản hoặc mạng xã hội trực tuyến. Kẻ tấn công sẽ tạo ra một phiên bản giả mạo của trang web gốc, hy vọng rằng nạn nhân sẽ tin tưởng và tuân theo các bước hướng dẫn trên trang web giả mạo để đánh cắp thông tin có tính bảo mật cao của người dùng.

1.3 Cách thức tấn công của tội phạm mạng

Có rất nhiều cách thức lừa đảo của tội phạm mạng trên Internet, nhưng nhìn chung, chúng đều có chung một quy trình lừa đảo [1]. Quy trình lừa đảo điển hình diễn ra được biểu diễn như hình 1.



Hình 1: Quy trình lừa đảo thông qua email và trang web giả mạo

1. Các kẻ lừa đảo thường sử dụng phương pháp Phishing Email để thực hiện các cuộc tấn công. Họ gửi email cho nạn nhân với hình thức và nội dung tương tự như các email thương mại hay thông báo giảm giá mà người dùng thường nhận được. Tuy

nhiên, trong những email này, kẻ lừa đảo sẽ chèn URL dẫn đến các website giả mạo mà họ đã tạo sẵn.

2. Những nạn nhân mất cảnh giác có thể cung cấp các thông tin nhạy cảm như tài khoản, mật khẩu, thông tin tài khoản ngân hàng,... thông qua những trang web mà kẻ lừa đảo đã cung cấp trước đó. Những thông tin này sau đó sẽ bị kẻ lừa đảo thu thập thông qua các trang web giả mạo hoặc các công cụ lừa đảo mà họ cung cấp cho nạn nhân.
3. Sau khi có được thông tin của nạn nhân, kẻ lừa đảo sẽ sử dụng nó để kiểm soát tài khoản cá nhân hoặc tài khoản ngân hàng của nạn nhân. Họ có thể trực lợi từ những thông tin này mà nạn nhân không hề hay biết, hoặc nếu nạn nhân nhận ra, có thể không kịp can thiệp để ngăn chặn hoặc khôi phục lại tài khoản của mình.
4. Sau khi có được thông tin của nạn nhân, những kẻ lừa đảo sẽ dùng chúng nhằm để kiểm soát tài khoản cá nhân hay tài khoản ngân hàng nhằm trực lợi mà các nạn nhân không hề hay biết hoặc biết nhưng có thể không can thiệp những kẻ lừa đảo kịp lúc.

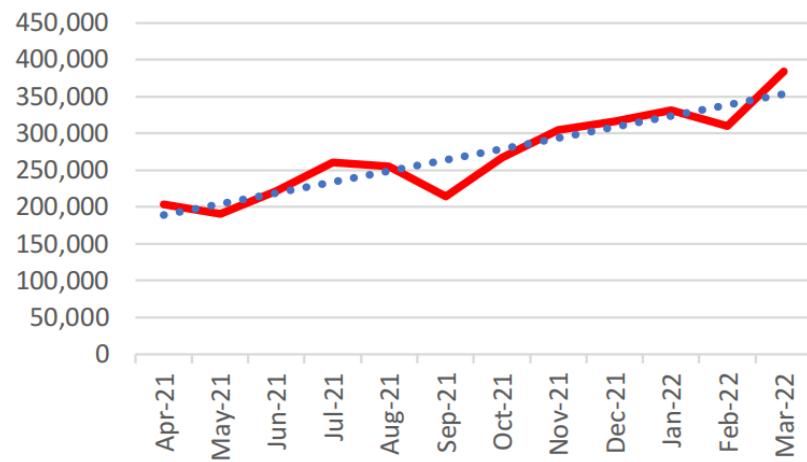
2. Thực trạng và tính cấp thiết của đề tài

2.1 Thực trạng chung hiện nay

Trong những năm trở lại đây, vấn đề tấn công lừa đảo (phishing attack) là một trong những vấn đề nghiêm trọng mà người dùng Internet, chính phủ và nhà cung cấp dịch vụ phải đối mặt. Theo báo cáo của Anti-Phishing Working Group (APWG) [2], trong quý đầu tiên của năm 2022, có tổng cộng 1,025,968 vụ tấn công lừa đảo. Đây là quý có nhiều vụ nhất từ trước đến nay, và cũng là quý đầu tiên có hơn một triệu vụ. Thêm vào đó, đã có 384,291 vụ vào tháng 3/2022, cũng là tháng kỷ lục về số vụ tấn công.

Trong quý đầu tiên của năm 2022, APWG chỉ rằng số lượng các vụ tấn công nhắm vào tài chính, bao gồm các ngân hàng, chiếm số lượng lớn nhất, lên đến 23,6% tổng các vụ tấn công. Những vụ tấn công nhắm vào webmail và nhà cung cấp phần mềm dạng dịch vụ (SaaS), phổ biến với những vụ tấn công vào các trang bán lẻ và thương mại điện tử giảm từ 17,3% xuống 14,6%. Những vụ lừa đảo tấn công vào phương tiện truyền thông mạng xã hội tăng từ 8,5% ở quý 4 năm 2021 lên 12,5% ở quý 1 năm 2022. Những vụ lừa đảo tấn công vào tiền mã hoá cũng tăng nhẹ từ 6,5% lên 6,6% trong thời gian đó.

Phishing Attacks, 2Q2021 - 1Q2022

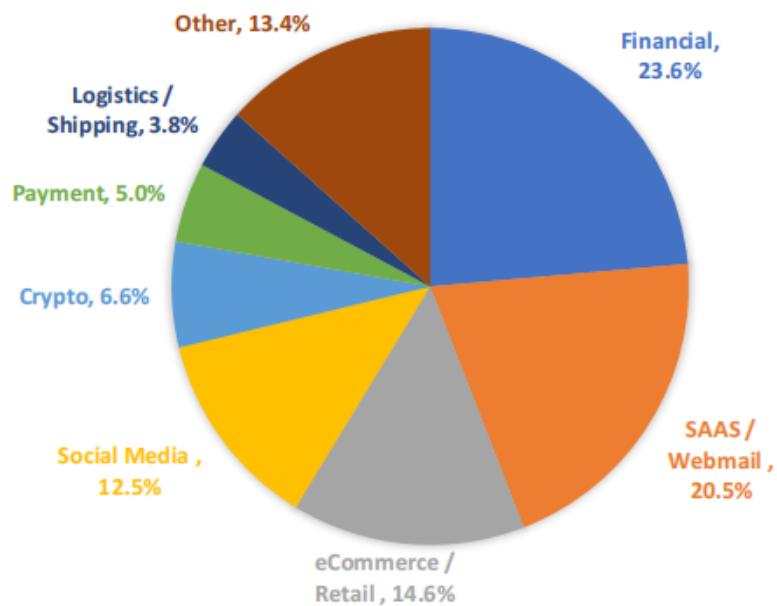


Hình 2: Số lượng tấn công lừa đảo từ tháng 4/2021 đến tháng 3/2022

Nguồn: https://docs.apwg.org/reports/apwg_trends_report_q1_2021.pdf

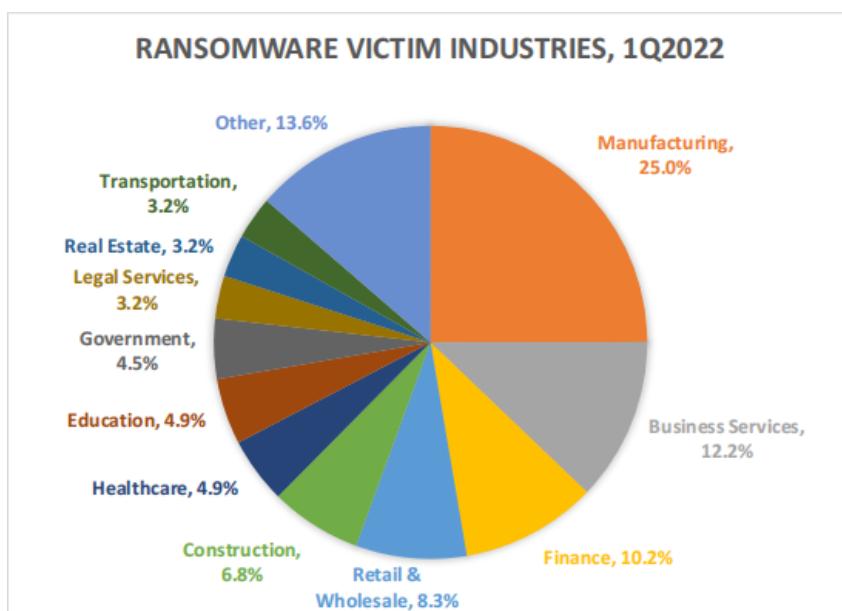
Về phương thức mã độc tống tiền (ransomware), tổng số lượng tấn công mã độc tống tiền đã giảm đi 25% vào 3 tháng đầu tiên của năm 2022, tương đương với quý 3/2021. Các ngành bị ảnh hưởng nặng nề nhất bởi mã độc tống tiền vào quý 4/2021 là ngành sản xuất, dịch vụ kinh doanh, tài chính, và các công ty bán lẻ và bán buôn.

MOST-TARGETED INDUSTRIES, 1Q2022



Hình 3: Các lĩnh vực bị tấn công trong quý 1 năm 2022

Nguồn: https://docs.apwg.org/reports/apwg_trends_report_q1_2021.pdf



Hình 4: Tấn công bằng hình thức mã độc tống tiền vào các lĩnh vực trong quý 1 năm 2022

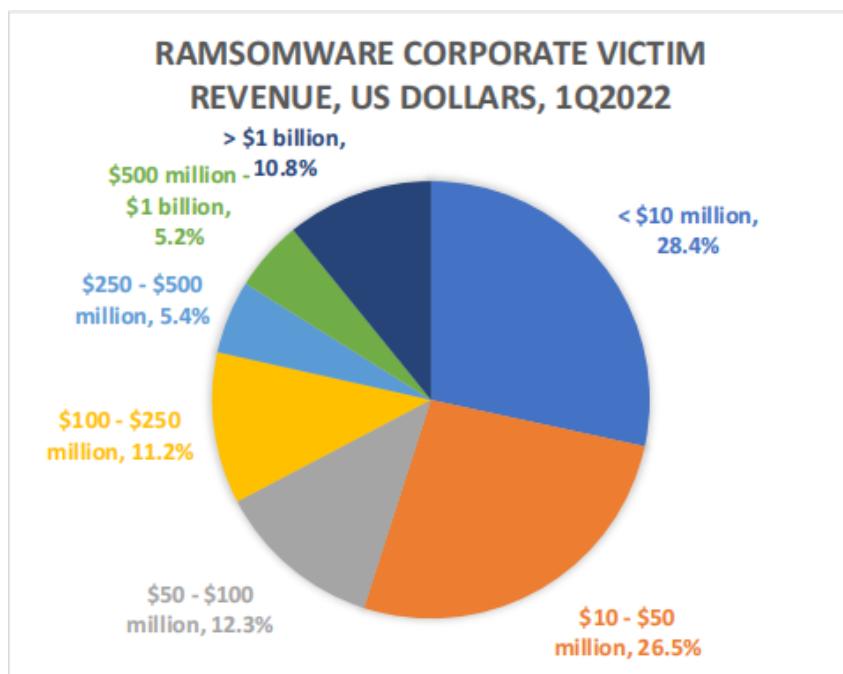
Nguồn: https://docs.apwg.org/reports/apwg_trends_report_q1_2021.pdf

Hầu như tất cả các ngành đều có sự sụt giảm về số lượng tấn công mã độc tống tiền. Ngoại lệ, với ngành dịch vụ tài chính có sự tăng lên đến 35% về số lượng các cuộc tấn công trong quý 1/2022.

Các kẻ tội phạm phát tán mã độc tống tiền có xu hướng nhắm vào những công ty đang trên đà phát triển, có khả năng chi trả một khoản lớn tiền chuộc xứng với công sức mà các kẻ đó gây ra, nhưng cũng không quá lớn để công ty vẫn có khả năng duy trì. Trong quý 1/2022, doanh thu trung bình hàng năm của các công ty là các nạn nhân của các cuộc tấn công mã độc là 31 triệu USD. Khoảng 55% trong số đó có doanh thu dưới 50 triệu USD, giảm từ 66% so với năm 2021. Các công ty nhỏ hơn dễ dàng bị tấn công hơn bởi vì không thể đầu tư số tiền lớn vào an ninh mạng. Nhưng vẫn có đến 11% các công ty có doanh thu hơn 1 tỉ USD là nạn nhân.

2.2 Thách thức hiện tại trong việc phòng chống tội phạm mạng

Hiện nay, đã có rất nhiều giải pháp được đưa ra để giải quyết vấn đề tấn công giả mạo, đồng thời cũng đã có rất nhiều nghiên cứu nhằm phát hiện vấn nạn lừa đảo [3, 4, 5, 6, 7]. tuy nhiên chúng ta thấy rằng không có giải pháp nào "đơn giản" để chống lại vấn đề này. Theo thời gian, vấn đề lừa đảo ngày càng trở nên phổ biến hơn trong việc thực hiện hành vi phạm tội qua mạng. Bất kể lúc nào khi một nhà nghiên cứu có ý tưởng mới để phát hiện và ngăn chặn điều đó, thì những kẻ lừa đảo luôn luôn có cách để khai thác các lỗ hổng trong giải pháp đó. Do đó chúng ta có thể nói rằng đây là cuộc đua giữa các nhà



Hình 5: Biểu đồ nạn nhân là các công ty dựa trên doanh thu

Nguồn: https://docs.apwg.org/reports/apwg_trends_report_q1_2021.pdf

nghiên cứu và các kẻ lừa đảo.

Những kẻ lừa đảo không ngừng tìm kiếm những phương thức mới và sáng tạo để đánh lừa người dùng, tạo ra những trang web và email giả mạo nhằm làm cho người nhận tin tưởng rằng họ đang truy cập vào các trang web chính thống. Hiện nay, có sự xuất hiện ngày càng nhiều các phương thức tấn công lừa đảo tiên tiến, nhắm trực tiếp vào những nạn nhân không đề phòng. Kẻ lừa đảo cũng tận dụng tâm lý của nạn nhân bằng cách gửi email và tạo ra các trang web giả mạo, tạo ra cảm giác khẩn cấp, tham lam hoặc tạo niềm tin cho nạn nhân. Điều này nhằm mục đích để lừa dối và xâm nhập vào thông tin cá nhân của họ. Từ đó, có thể thấy rõ sự khéo léo và tinh vi của những kẻ lừa đảo trong việc tạo ra các cuộc tấn công lừa đảo đầy nguy hiểm trong môi trường số hóa hiện nay.

Trong những năm gần đây, các cuộc tấn công lừa đảo trở thành một trong những mối đe dọa nghiêm trọng nhất mà người dùng Internet, các tổ chức và các nhà cung cấp dịch vụ phải đối mặt. Có nhiều đề xuất để phát hiện và lọc các cuộc tấn công lừa đảo, tuy nhiên chúng ta vẫn cần một giải pháp hoàn hảo hơn để bảo vệ khỏi các cuộc tấn công.

2.3 Tính cấp thiết của đề tài

Trong thời đại phát triển mạnh mẽ của mạng Internet hiện nay, người dùng dễ dàng tiếp cận và khám phá vô số tài nguyên. Tuy nhiên, không phải ai cũng có đủ kiến thức, thông tin cơ bản và kinh nghiệm để phòng tránh các trang web giả mạo. Điều này làm cho người dùng Internet trở nên dễ bị kẻ lừa đảo tấn công. Một số lý do chính mà người

dùng dễ dàng trở thành nạn nhân của các cuộc tấn công lừa đảo bao gồm:

- Thiếu kiến thức cơ bản về Uniform Resource Locator (URL): Đa số người dùng chưa nắm vững các khái niệm và quy tắc căn bản về URL, dẫn đến việc khó phân biệt giữa các địa chỉ trang web chính thống và giả mạo.
- Ẩn địa chỉ thật của trang web giả mạo: Kẻ lừa đảo thường giấu hoặc che đậy địa chỉ thật của trang web giả mạo, làm cho người dùng không nhận ra rằng họ đang bị điều hướng đến một trang web giả mạo.
- Thiếu kinh nghiệm trong việc phòng tránh trang web lừa đảo: Người dùng thiếu kinh nghiệm khi truy cập vào các trang web và không nhận ra các dấu hiệu đáng ngờ hoặc biết cách xác minh tính xác thực của trang web.
- Khả năng phân biệt kém giữa trang web chính thống và trang web lừa đảo: Nạn nhân không thể phân biệt được trang web chính thống và trang web lừa đảo, dẫn đến việc tin tưởng và cung cấp thông tin cá nhân cho những trang web không đáng tin cậy.

Các trang web lừa đảo mà những kẻ này tạo ra, thường có những đặc điểm sau:

- Giao diện và chức năng tương tự với trang web chính gốc, gây nhầm lẫn cho người dùng.
- URL kì lạ, không có tên miền đáng tin.
- Một số trang web lừa đảo vẫn sử dụng giao diện cũ của trang web chính thức để làm cho người dùng cảm thấy tin tưởng.

Điều này cho thấy sự tinh vi và sáng tạo của kẻ lừa đảo trong việc phát triển các phương thức lừa đảo mới nhằm đánh lừa người dùng. Họ không ngừng tìm kiếm cách thức và mưu mẹo để tạo ra các trang web giả mạo tinh vi, nhằm khiến người dùng tin rằng họ đang truy cập vào trang web chính thống. Kẻ lừa đảo cũng tận dụng tâm lý của nạn nhân bằng cách gửi email hoặc tạo ra các trang web giả mạo mang tính khẩn cấp hoặc hứa hẹn lợi ích lớn, nhằm kích thích sự tham lam hoặc tạo ra cảm giác cấp bách, từ đó thuyết phục nạn nhân tin tưởng và cung cấp thông tin cá nhân.

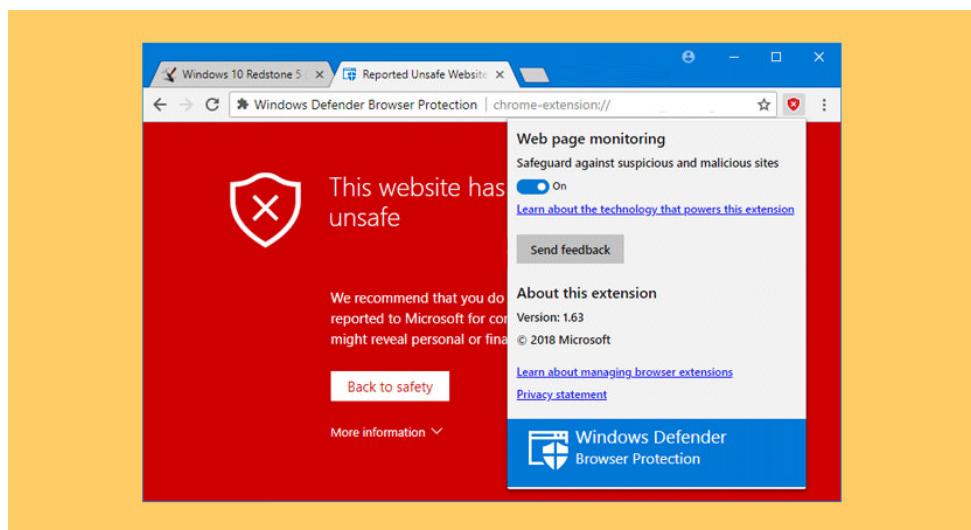
Trong bối cảnh này, việc nâng cao kiến thức và nhận thức của người dùng Internet về phòng tránh các trang web giả mạo trở nên cực kỳ quan trọng. Đồng thời, các tổ chức và chính phủ cần chủ động tăng cường công tác giáo dục, cung cấp thông tin, và đưa ra các biện pháp bảo vệ để giảm thiểu rủi ro và tổn thất gây ra bởi các cuộc tấn công lừa đảo.

3. Mục tiêu và giá trị khoa học mà đề tài này mang lại

3.1 Mục tiêu mà đề tài này hướng đến

Dựa vào tính cấp thiết hiện nay, nhóm mong muốn có thể phát triển một hệ thống phần mềm giúp phát hiện những trang Web giả mạo hay những email giả mạo để từ đó cảnh báo cho người dùng. Bên cạnh đó tích hợp phần mềm vào trình duyệt web như Chrome, Cốc cốc, Firefox,... như một tiện ích mở rộng (extension) đem lại hiệu quả tối đa.

Hình 6 đã minh họa được một phần nào ứng dụng của đề tài này, kết hợp với các mô hình học máy sẽ giúp cho phần mềm tăng độ chính xác cũng như tốc độ xử lý so với các giải pháp thông thường. Vì vậy trong đề tài này, ngoài việc xây dựng phần mềm ra còn chọn ra mô hình tối ưu nhất cho phần mềm.



Hình 6: Minh họa mục tiêu sản phẩm

Nguồn: <https://www.geckoandfly.com/25017/security-toolbar-phishing-websites/>

3.2 Giá trị khoa học mà đề tài này mang lại

Thời đại công nghệ ngày càng phát triển và phổ biến ở mọi lĩnh vực kinh tế, chính trị, giao thông, giải trí,... do đó hầu hết các thông tin, tài nguyên đều được lưu trữ trên các máy chủ, đám mây và người dùng tương tác với những dữ liệu đó thông qua mạng. Chính nhờ sự thuận tiện trong việc giao tiếp, xử lý thông tin đã dẫn đến một hệ lụy nghiêm trọng đó là người dùng hay doanh nghiệp có thể bị đánh cắp dữ liệu từ những kẻ giả mạo đánh cắp thông tin. Những kẻ giả mạo này ngày càng tinh vi và thông minh hơn vì thế các phương pháp thông thường sẽ càng khó có thể phát hiện, chỉ cần một cái nhấp chuột cũng có thể khiến cho dữ liệu bị rò rỉ, đây đều là những dữ liệu quan trọng, cần

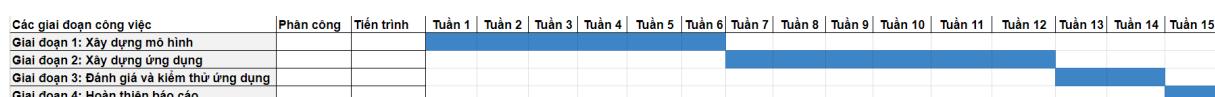
được bảo vệ hàng đầu. Vì vậy việc phát triển Trí tuệ nhân tạo (AI) và áp dụng vào mọi lĩnh vực sẽ góp phần phát triển mạnh mẽ mọi ngành công nghiệp đồng thời thúc đẩy về lĩnh vực an ninh mạng. Về việc bảo mật, các giải thuật AI sẽ mang lại tốc độ, độ chính xác và khả năng thực hiện mở rộng phạm vi dò tìm tốt hơn so với các giải thuật thông thường. Bên cạnh đó áp dụng các giải thuật AI sẽ giúp phát hiện thư rác, những email giả mạo và các loại tấn công khác nhau nhắm vào người dùng. Từ đó dự án này hy vọng sẽ góp phần thúc đẩy quá trình phát hiện những kẻ giả mạo tốt hơn thông qua các giải thuật AI phù hợp, hướng tới các giá trị quan trọng:

- Giá trị nhận thức: Việc nghiên cứu này sẽ giúp những người dùng tăng nhận thức hơn về bảo mật an ninh mạng, tránh các rủi ro có thể xảy ra, nâng cao chất lượng không gian mạng.
- Giá trị sáng tạo: Đề tài này còn là cơ sở cho các cá nhân, tổ chức khác phát triển, nâng cao khả năng bảo mật thông tin hay tích hợp vào các phần mềm nhận diện giả mạo tiên tiến hơn.
- Giá trị kinh tế: Việc nhận biết được các website lừa đảo sẽ giúp các doanh nghiệp có thể cung cấp thông tin của mình, tránh để các dữ liệu đến tay những người lừa đảo vì họ có thể sử dụng thông tin này để làm giàu bản thân và phá hoại nền kinh tế thị trường.

4. Kế hoạch thực hiện đồ án

Theo kế hoạch từ nhà trường, đề tài sẽ được hoàn thiện trong vòng 15 tuần. Vì vậy nhóm đã phân chia kế hoạch một cách chi tiết để mọi người có thể nắm bắt một cách dễ dàng tiến trình mà nhóm đang thực hiện.

Từ giản đồ hình 7 trên đề tài được chia thành 4 giai đoạn chính. Mỗi giai đoạn sẽ có nhiều nhiệm vụ riêng lẻ và thời gian thực hiện nhất định:



Hình 7: Giản đồ Gantt thực hiện đề tài

Giai đoạn 1: Xây dựng mô hình

Các giai đoạn công việc	Phân công	Tiến trình	Tuần 1	Tuần 2	Tuần 3	Tuần 4	Tuần 5	Tuần 6
Giai đoạn 1: Xây dựng mô hình								
Tìm hiểu về Ensemble Learning	Hoàng An	Đã hoàn thành						
Tìm hiểu những ứng dụng phát hiện website giả mạo hiện nay	Hoàng An	Đã hoàn thành						
Tìm phương pháp phát hiện Website giả mạo cổ điển	Đặng Hải	Đã hoàn thành						
Tìm nguồn dataset các trang Web giả mạo	Hoàng An + Đặng Hải	Đã hoàn thành						
Tìm hiểu giải thuật Bagging	Đặng Hải	Đã hoàn thành						
Tìm hiểu giải thuật Boosting	Đặng Hải	Đã hoàn thành						
Tìm hiểu giải thuật Stacking	Hoàng An	Đã hoàn thành						
Tổng hợp những đặc trưng giúp phát hiện Website giả mạo	Hoàng An	Đã hoàn thành						
Lấy dữ liệu từ nguồn dataset	Đặng Hải	Đã hoàn thành						
Trích xuất đặc trưng từ dữ liệu	Hoàng An + Đặng Hải	Đã hoàn thành						
Training mô hình bằng giải thuật SVM	Hoàng An	Đã hoàn thành						
Training mô hình bằng giải thuật Logistic Regression	Hoàng An	Đã hoàn thành						
Training mô hình bằng giải thuật Decision Tree	Đặng Hải	Đã hoàn thành						
Training mô hình bằng giải thuật KNN	Đặng Hải	Đã hoàn thành						
Training mô hình bằng giải thuật Random Forest	Hoàng An	Đã hoàn thành						
Trực quan hóa dữ liệu	Hoàng An	Đã hoàn thành						
Training mô hình bằng giải thuật Bagging	Hoàng An	Đã hoàn thành						
Training mô hình bằng giải thuật Boosting	Hoàng An	Đã hoàn thành						
Training mô hình bằng giải thuật Stacking	Hoàng An	Đã hoàn thành						
Viết mã giả cho quá trình training dữ liệu	Hoàng An	Đã hoàn thành						
Viết mã giả cho quá trình dự đoán của mô hình	Hoàng An	Đã hoàn thành						
So sánh thời gian xử lý giữa các mô hình	Hoàng An	Đã hoàn thành						
So sánh khả năng học tập giữa các mô hình	Hoàng An	Đã hoàn thành						
Nhận xét và so sánh các giải thuật với nhau	Hoàng An	Đã hoàn thành						

Hình 8: Giản đồ Gantt thực hiện giai đoạn 1

Giai đoạn 1 là giai đoạn nhóm bắt đầu tìm hiểu về các giải thuật Ensemble Learning, từ đó xây dựng các mô hình mới nâng cấp từ các mô hình cũ với cải thiện dữ liệu.

Giai đoạn 2: Xây dựng ứng dụng

Các giai đoạn công việc	Phân công	Tiến trình	Tuần 7	Tuần 8	Tuần 9	Tuần 10	Tuần 11	Tuần 12
Giai đoạn 2: Xây dựng ứng dụng								
Liệt kê các chức năng của ứng dụng	Hoàng An + Đặng Hải	Đã hoàn thành						
Thiết kế UI ứng dụng	Hoàng An + Đặng Hải	Đã hoàn thành						
Xây dựng hệ thống ứng dụng	Đặng Hải	Đã hoàn thành						
Build Extension	Đặng Hải	Đã hoàn thành						
Tìm hiểu về Azure App Service	Đặng Hải	Đã hoàn thành						
Thuê và sử dụng Azure App Service	Đặng Hải	Đã hoàn thành						
Build và chạy Server trên Azure App Service	Đặng Hải	Đã hoàn thành						
Viết code API trích xuất đặc trưng	Đặng Hải	Đã hoàn thành						
Viết code API giúp nhận diện web giả mạo	Đặng Hải	Đã hoàn thành						
Code giao diện ứng dụng	Đặng Hải	Đã hoàn thành						
Bổ sung chức năng white-black list	Đặng Hải	Đã hoàn thành						
Vẽ lược đồ Use-case hệ thống	Hoàng An + Đặng Hải	Đã hoàn thành						
Vẽ lược đồ Sequence cho chức năng cảnh báo người dùng	Hoàng An + Đặng Hải	Đã hoàn thành						
Vẽ lược đồ Sequence cho chức năng kiểm tra website	Hoàng An + Đặng Hải	Đã hoàn thành						
Vẽ lược đồ Sequence cho chức năng báo cáo website	Hoàng An + Đặng Hải	Đã hoàn thành						
Vẽ lược đồ back-end của hệ thống	Hoàng An + Đặng Hải	Đã hoàn thành						
Viết mã giả cho back-end của hệ thống	Đặng Hải	Đã hoàn thành						
Fix bug hệ thống	Đặng Hải	Đã hoàn thành						

Hình 9: Giản đồ Gantt thực hiện giai đoạn 2

Từ những ý tưởng đã được nêu ra từ Đồ án chuyên ngành, nhóm tiếp tục cải thiện và nâng cao các chức năng mới của ứng dụng. Từ đó trong giai đoạn này nhóm sẽ tập trung viết ứng dụng và cải thiện hệ thống.

Giai đoạn 3: Đánh giá và kiểm thử ứng dụng

Các giai đoạn công việc	Phân công	Tiến trình	Tuần 13	Tuần 14
Giai đoạn 3: Đánh giá và kiểm thử ứng dụng				
Kiểm thử tính năng cảnh báo người dùng	Hoàng An	Đã hoàn thành		
Kiểm thử tính năng kiểm tra website	Hoàng An	Đã hoàn thành		
Kiểm thử tính năng báo cáo website	Hoàng An	Đã hoàn thành		
Đánh giá thời gian phản hồi của từng tính năng	Hoàng An	Đã hoàn thành		
Đánh giá tốc độ hệ thống khi nhiều người sử dụng	Hoàng An	Đã hoàn thành		
Đánh giá của người dùng về ứng dụng	Đặng Hải	Đã hoàn thành		
Nhận xét và kết luận của người dùng	Hoàng An + Đặng Hải	Đã hoàn thành		

Hình 10: Giản đồ Gantt thực hiện giai đoạn 3

Sau khi hoàn thành hệ thống, cần phải đưa ra những use case phù hợp để đánh giá từng chức năng, thời gian phản hồi của từng module. Bên cạnh đó còn xem xét những ý kiến chủ quan từ người sử dụng khác để từ đó kết luận và đánh giá hướng cải thiện.

Giai đoạn 4: Hiện thực báo cáo

Các giai đoạn công việc	Phân công	Tiến trình	Tuần 15
Giai đoạn 4: Hoàn thiện báo cáo			
Kết luận những gì đã làm được	Hoàng An + Đặng Hải	Đã hoàn thành	
Những đóng góp của hệ thống	Hoàng An + Đặng Hải	Đã hoàn thành	
Những mặt hạn chế hiện tại	Hoàng An + Đặng Hải	Đã hoàn thành	
Hướng phát triển của hệ thống trong tương lai	Hoàng An + Đặng Hải	Đã hoàn thành	
Sửa lỗi chính tả + căn chỉnh lề, hình ảnh	Hoàng An + Đặng Hải	Đã hoàn thành	

Hình 11: Giản đồ Gantt thực hiện giai đoạn 4

Trong tuần cuối cùng nhóm sẽ tổng hợp lại và kết luận những gì đã làm được, những gì trội hơn so với các bài báo cáo đã tham khảo. Điểm lại những hạn chế chưa thực hiện được từ đó định hướng phát triển tương lai của hệ thống trong tương lai.

II. Tổng quan về kiến thức sử dụng

1. Giới thiệu về bài toán phát hiện bất thường bằng phương pháp học máy

Trước khi giới thiệu về bài toán phát hiện bất thường bằng học máy, cần hiểu rõ khái niệm về bất thường. Bất thường là những hành vi hoặc dữ liệu có sự khác biệt đáng kể so với mẫu chuẩn. Thuật ngữ "outliers" và "novelties" cũng liên quan đến bất thường, cả hai đều chỉ các dữ liệu có tính chất khác biệt trong tập dữ liệu. Outliers đại diện cho những điểm dữ liệu không giống hoặc khác biệt so với các điểm dữ liệu khác, trong khi novelties xuất hiện khi dữ liệu trong mô hình thay đổi theo thời gian, dẫn đến xuất hiện các điểm dữ liệu mới.

Tiêu chuẩn xác định bất thường dựa trên ngữ cảnh mà nó áp dụng. Ví dụ, một người có chiều cao 190 cm có thể được coi là bất thường ở Việt Nam, nhưng lại được coi là bình thường ở Hà Lan. Tóm lại, phát hiện bất thường là một phương pháp tự động được

sử dụng để xác định các dữ liệu đáng ngờ hoặc có sự khác biệt đáng kể so với phần lớn dữ liệu. Ví dụ, một tài khoản ngân hàng có thể bị coi là bất thường nếu có số lượng giao dịch lớn hơn bình thường tại một thời điểm hoặc địa điểm không phổ biến trong ngày.

Phát hiện bất thường bằng học máy là một phương pháp phổ biến trong lĩnh vực này. Nó sử dụng các thuật toán học máy để tìm ra các mẫu và đặc trưng trong dữ liệu để xác định những điểm dữ liệu bất thường. Các phương pháp phát hiện bất thường bằng học máy có thể dựa trên các mô hình như Gaussian Mixture Models (GMM), Support Vector Machines (SVM), Isolation Forest, và Autoencoders. Tùy thuộc vào loại dữ liệu và bài toán cụ thể, các phương pháp này có ưu điểm và hạn chế riêng.

2. Cách tiếp cận phát hiện sự bất thường bằng học máy

Sự phát hiện bất thường hay phát hiện ngoại lai này có thể được thực hiện bằng cách sử dụng các khái niệm của học máy, các kỹ thuật thường được chia thành 3 loại: Phát hiện bất thường có giám sát (Supervised Anomaly Detection), phát hiện bất thường không giám sát (Unsupervised Anomaly Detection) và phát hiện bất thường bán giám sát (Semi-supervised Anomaly Detection).

Phát hiện bất thường có giám sát

Trong phát hiện bất thường có giám sát, quá trình phân loại sẽ được đào tạo bằng cách sử dụng tập dữ liệu đã được gắn nhãn là "bình thường" hoặc "bất bình thường". Dựa trên tập huấn luyện đó, khi một dữ liệu mới xuất hiện thì mô hình sẽ phân loại dựa trên sự tương đồng với các dữ liệu đã được gán nhãn để từ đó xếp nhãn vào dữ liệu mới.

Ưu điểm của các mô hình được giám sát là chúng có thể cho ra mô hình phân loại với tỷ lệ phát hiện cao hơn so với các kỹ thuật khác. Điều này là do chúng có thể trả về độ chính xác dựa trên output so với kết quả, để từ đó kết hợp với các dữ liệu đã có từ trước đồng thời có thể tìm ra sự phụ thuộc các biến lẫn nhau để cải thiện dữ liệu cũng như cải thiện mô hình. Nhưng bên cạnh đó, việc dán nhãn dữ liệu sẽ rất mất thời gian vì tập dữ liệu rất lớn nên thông thường đối với các giải thuật này cần phải xử lý dữ liệu trước. Các phương pháp được giám sát phổ biến nhất bao gồm KNN, Decision Tree, SVM,...

Phát hiện bất thường không giám sát

Kỹ thuật được sử dụng phổ biến nhất trong việc phát hiện bất thường bằng học máy là phát hiện bất thường không giám sát. Phương pháp không giám sát được sử dụng bằng việc đào tạo mô hình máy học mà sử dụng tập dữ liệu không được gán nhãn, vì vậy tập dữ liệu sẽ không cần yêu cầu gán nhãn bằng thủ công. Quá trình đó được thực hiện khi ta dựa trên giả định rằng chỉ có một tỷ lệ nhỏ, dựa trên sự khác biệt về mặt thống kê, là các điểm bất thường. Sau đó thu nhập các điểm dữ liệu tương tự, bất kể điểm dữ liệu nào có sự khác biệt đáng kể so với các điểm bình thường sẽ được đánh dấu là bất bình thường.

Các thuật toán phát hiện bất thường không giám sát phổ biến bao gồm: Autoencoders, k-Means, GMMs, PCAs,...

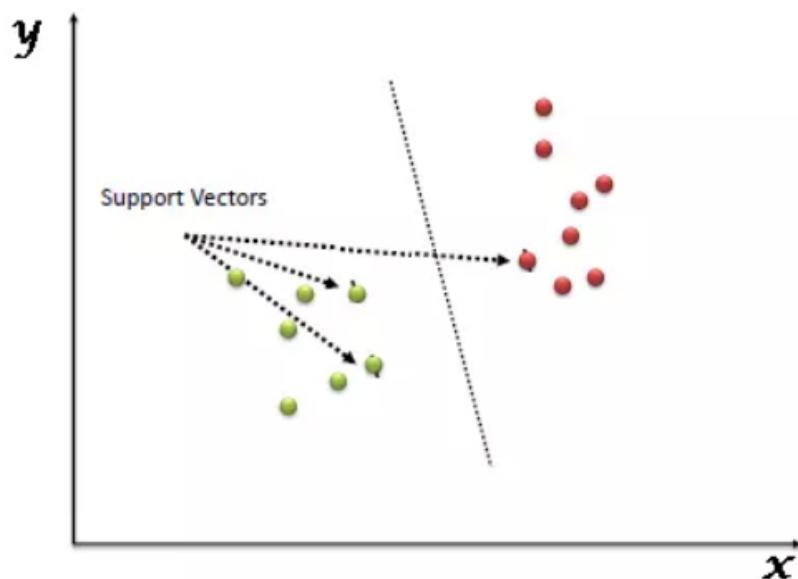
Phát hiện bất thường bán giám sát

Thông thường kỹ thuật bán giám sát có thể được đề cập đến cách tiếp cận training tập dữ liệu với các điểm bình thường trên một tập dữ liệu lớn chứa cả điểm bình thường và không bình thường nhưng không được gán nhãn. Vì vậy để sử dụng các giải thuật bán giám sát, tập dữ liệu cũng cần phải gán nhãn một phần bên trong. Sau đó xây dựng một thuật toán phân loại chỉ dựa trên tập hợp con dữ liệu đã được gán nhãn đó để phân loại gán nhãn phần còn lại của tập dữ liệu.

3. Các giải thuật sử dụng phát hiện bất thường

3.1 Giải thuật SVM

SVM hay Support Vector Machine [8] là một thuật toán giám sát được sử dụng chủ yếu cho việc phân loại. Trong thuật toán này, chúng ta vẽ đồ thị dữ liệu là các điểm trong n chiều (ở đây n là số lượng các tính năng) với giá trị của mỗi tính năng sẽ là một phần liên kết. Sau đó chúng ta thực hiện tìm "đường bay" (hyper-plane) phân chia các lớp. Hyper-plane nó chỉ hiểu đơn giản là 1 đường thẳng có thể phân chia các lớp ra thành hai phần riêng biệt.



Hình 12: Hyper-plane phân chia 2 lớp riêng biệt

Nguồn: <https://viblo.asia/p/gioi-thieu-ve-support-vector-machine-svm-6J3ZgPVElmB>

Mục tiêu của thuật toán này là tìm ra hyper-plane tốt nhất cho không gian n chiều thành các lớp phân biệt để ta có thể dễ dàng đưa điểm dữ liệu mới vào và phân loại nó

thuộc lớp nào. Bên cạnh đó tìm ra hyper-lane sao cho khoảng cách từ điểm gần nhất của mỗi class (các điểm được khoanh tròn) tới đường phân chia là như nhau, như thế thì mới công bằng. Khoảng cách như nhau này được gọi là *margin* (*lề*).

Giải thuật:

1. Tập hợp tập dữ liệu trong tập training set là tập vector x_i với $\forall x_i \in R^d$ trong đó d là số chiều của không gian. Với mỗi điểm dữ liệu sẽ được gán nhãn y_i tùy vào số lớp có trong tập dữ liệu.
2. Tính khoảng cách từ một điểm vector có tọa độ x_0 tới mặt siêu phẳng (hyperlane) có phương trình $w^\tau x + b = 0$ được xác định bởi công thức:

$$\frac{w^\tau x_0 + b}{\|w\|_2}$$

Với $\|w\|_2 = \sqrt{\sum w_i^2}$

3. Tính khoảng cách margin là khoảng cách gần nhất từ 1 điểm tới mặt đó:

$$\text{margin} = \min \frac{y_n(w^\tau x_n + b)}{\|w\|_2}$$

4. Tối ưu SVM bằng cách tìm giá trị w và b sao cho margin đạt giá trị lớn nhất:

$$(w, b) = \operatorname{argmax} \frac{1}{\|w\|_2}$$

Với điều kiện $y_n(w^\tau x_n + b) \geq 1$

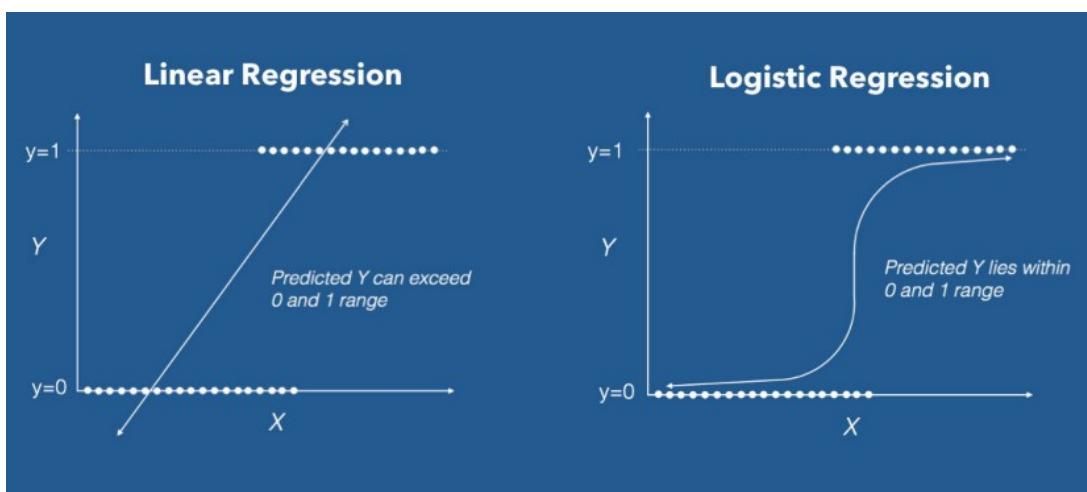
5. Sau khi tìm được mặt phân cách $w^\tau x + b = 0$ mới, xác định lại class của một vector x bất kỳ:

$$\text{class}(x) = \operatorname{sgn}(w^\tau x + b)$$

Trong đó hàm sgn là hàm xác định dấu.

3.2 Giải thuật Logistic Regression

Logistic Regression [8] là 1 thuật toán phân loại được dùng để gán các đối tượng cho 1 tập hợp giá trị rời rạc (như 0, 1, 2, ...). Bên cạnh đó Logistic Regression thường được sử dụng để so sánh với các thuật toán phân loại khác và là một phương pháp phân tích thống kê được sử dụng để dự đoán giá trị dữ liệu dựa trên các quan sát trước đó của tập dữ liệu. Mục đích của giải thuật này là ước tính xác suất của các sự kiện, bao gồm xác định mối quan hệ giữa các tính năng từ đó dự đoán xác suất của các kết quả. Logistic Regression cũng có thể được coi như là một mô hình của Linear Regression



Hình 13: Linear Regression VS Logistic Regression Graph

Nguồn: <https://medium.com/@maithilijoshi6/a-comparison-between-linear-and-logistic-regression-8aea40867e2d>

nhưng Logistic Regression sử dụng hàm chi phí phức tạp hơn đó là *hàm Sigmoid* thay vì hàm tuyến tính trong Linear Regression.

Hồi quy Logistic[8] có xu hướng giới hạn hàm chi phí xuống trong khoảng từ 0 đến 1 để dự đoán xác suất của các đối tượng, điều mà hồi quy tuyến tính không thể biểu diễn nó được vì giá trị dự đoán của hồi quy tuyến tính sẽ không bị giới hạn. Vì vậy cần sử dụng hàm Sigmoid để vào mô hình của hồi quy tuyến tính để từ đó dự đoán được xác suất của từng đối tượng .

Giải thuật:

1. Xây dựng **activation function** là một hàm Sigmoid để dự đoán output y từ tập dữ liệu x:

$$y = f(w^\top x) = \frac{1}{1 + e^{-w^\top x}}$$

2. Tính toán độ sai lệch so với giá trị y thực tế để từ đó tính toán **loss function**. Mục đích của chúng ta tìm các hệ số w sao cho $f(w^\top x_i)$ càng gần với 1 càng tốt với các điểm dữ liệu thuộc class 1 và càng gần với 0 càng tốt với những điểm thuộc class 0. Lúc này bài toán trở thành tìm giá trị nhỏ nhất của hàm mất mát:

$$J(w) = - \sum (y_i \log z_i + (1 - y_i) \log(1 - z_i))$$

Với chú ý rằng z_i là giá trị dự đoán từ hàm sigmoid và y_i là giá trị thực tế từ tập dữ liệu.

3. Sau khi có được hàm mất mát ta cần tối ưu hóa để tìm ra w phù hợp. Để tối ưu hóa cần sử dụng thuật toán **Gradient descent** để tìm ra tốc độ suy giảm. Đạo hàm hàm

mắt mát với điểm dữ liệu (x_i, y_i) :

$$\begin{aligned}\frac{\Delta J(w; x_i; y_i)}{\Delta w} &= \frac{z_i - y_i}{z_i(1 - z_i)} \frac{\Delta z_i}{\Delta w} \\ &= (z_i - y_i)x_i\end{aligned}$$

Cập nhật công thức lại cho logistic sigmoid regression:

$$w = w + \alpha(y_i - z_i)x_i$$

4. Lặp lại các bước trên đến khi tìm được hàm $w^T x$ phù hợp nhất.

3.3 Giải thuật Decision Tree

Cây quyết định (Decision Tree)[8] là thuật toán học có giám sát không tham số, được sử dụng cho cả phân loại và hồi quy. Nó có cấu trúc dạng cây, phân cấp, bao gồm nút gốc (root node), các nhánh, các nút bên trong (internal node) và các nút lá (leaf nodes).

Cây quyết định bắt đầu bằng một nút gốc, không có bất kỳ nhánh nào đến. Các nhánh đi từ nút gốc sau đó đưa vào các nút bên trong, còn được gọi là nút quyết định. Dựa trên các đặc điểm sẵn có, cả hai loại nút đều tiến hành đánh giá để tạo thành các tập con đồng nhất, được ký hiệu bằng các nút lá, hoặc các nút đầu cuối. Các nút lá đại diện cho tất cả các kết quả có thể có trong tập dữ liệu.

Cây quyết định là một kỹ thuật học thuật có giám sát có thể được sử dụng cho cả bài toán phân loại và bài toán hồi quy, nhưng chủ yếu nó được ưu tiên hơn để giải các bài toán phân loại. Nó là một bộ phân loại có cấu trúc cây, trong đó các nút bên trong đại diện cho các tính năng của tập dữ liệu, các nhánh đại diện cho các quy tắc quyết định và mỗi nút lá đại diện cho kết quả. Trong cây Quyết định, có hai nút, đó là Nút quyết định và Nút lá. Các nút quyết định được sử dụng để đưa ra bất kỳ quyết định nào và có nhiều nhánh, trong khi nút Lá là đầu ra của các quyết định đó và không chứa bất kỳ nhánh nào khác.

Giải thuật:

Thuật toán ID3:

Input: Một tập hợp các ví dụ. Mỗi ví dụ bao gồm các thuộc tính mô tả một tình huống, hay một đối tượng nào đó, và một giá trị phân loại của nó.

Output: Cây quyết định có khả năng phân loại đúng đắn các ví dụ trong tập dữ liệu rèn luyện, và hy vọng là phân loại đúng cho cả các ví dụ chưa gặp trong tương lai.

Lặp:

1. Chọn A \leq thuộc tính quyết định “tốt nhất” cho nút kế tiếp

2. Gán A là thuộc tính quyết định cho nút
3. Với mỗi giá trị của A, tạo nhánh con mới của nút
4. Phân loại các mẫu huấn luyện cho các nút lá
5. Nếu các mẫu huấn luyện được phân loại hoàn toàn thì dừng, ngược lại lặp với các nút lá mới.

Thuộc tính tốt nhất ở đây là thuộc tính có entropy trung bình thấp nhất theo thuộc tính kết quả với Entropy. Cho một phân phối xác suất của một biến rời rạc x có thể nhận n giá trị khác nhau x_1, x_2, \dots, x_n . Giả sử rằng xác suất để x nhận các giá trị này là $p_i = p(x = x_i)$ với $0 \leq p_i \leq 1$, $\sum_{i=1}^n p_i = 1$. Ký hiệu phân phối này là $p = (p_1, p_2, \dots, p_n)$. Entropy của phân phối này được tính:

$$H(p) = - \sum_{i=1}^n p_i \log(p_i)$$

3.4 Giải thuật KNN

Thuật toán K-Nearest Neighbors (KNN)[8] là một phương pháp học có giám sát trong máy học, được sử dụng để phân loại các quan sát mới bằng cách tìm các điểm dữ liệu tương đồng trong tập dữ liệu đã có.

KNN là một mô hình đơn giản và dễ hiểu, nhưng lại mang lại hiệu quả cao, vì nó không yêu cầu các tham số và không đặt giả định về phân phối dữ liệu. Nó cũng có thể được áp dụng trực tiếp để phân loại các lớp dữ liệu đa biến.

Trong bài toán phân loại, KNN suy ra nhãn của một điểm dữ liệu mới bằng cách xem xét nhãn của K điểm dữ liệu gần nhất trong tập huấn luyện. Có thể sử dụng phương pháp bầu chọn phần đông (major voting) giữa các điểm gần nhất để xác định nhãn của dữ liệu kiểm tra, hoặc có thể gán trọng số khác nhau cho các điểm gần nhất và từ đó suy ra nhãn.

Để tóm gọn, KNN là một thuật toán tìm ra đầu ra của một điểm dữ liệu mới dựa trên thông tin của K điểm dữ liệu gần nhất trong tập huấn luyện, mà không quan tâm đến sự hiện diện của nhiều trong số các điểm gần nhất này.

Giải thuật:

1. Thu thập, làm sạch,... dữ liệu.
2. Khởi tạo giá trị K (số điểm lân cận dùng để xét nhãn).
3. Với mỗi trường hợp sẵn có trong tập dữ liệu:
 - (a) Tính toán khoảng cách giữa trường hợp đó và trường hợp đang cần dự đoán.
Việc tìm khoảng cách giữa 2 điểm cũng có nhiều công thức có thể sử dụng,

tùy trường hợp mchúng ta lựa chọn cho phù hợp. Đây là 3 cách cơ bản để tính khoảng cách 2 điểm dữ liệu x, y có n thuộc tính:

- Công thức Euclidean:

$$(x, y) = \sqrt{\sum_{i=1}^n (x_i - y_i)^2}$$

- Công thức Manhattan:

$$(x, y) = \sum_{i=1}^n (|x_i - y_i|)$$

- Công thức Minkowski:

$$(x, y) = \left(\sum_{i=1}^n (|x_i - y_i|)^q \right)^{\frac{1}{q}}$$

- (b) Thêm giá trị khoảng cách của điểm vừa tính được vào tập giá trị khoảng cách.
4. Sắp xếp tập giá trị khoảng cách giữa các điểm có sẵn với điểm cần xét theo thứ tự từ nhỏ đến lớn.
 5. Chọn K giá trị đầu tiên trong tập giá trị khoảng cách đã sắp xếp.
 6. Lấy nhãn của K điểm đã được chọn ở trên.
 7. Trả về nhãn dự đoán từ K nhãn được chọn.

3.5 Giải thuật Random Forest

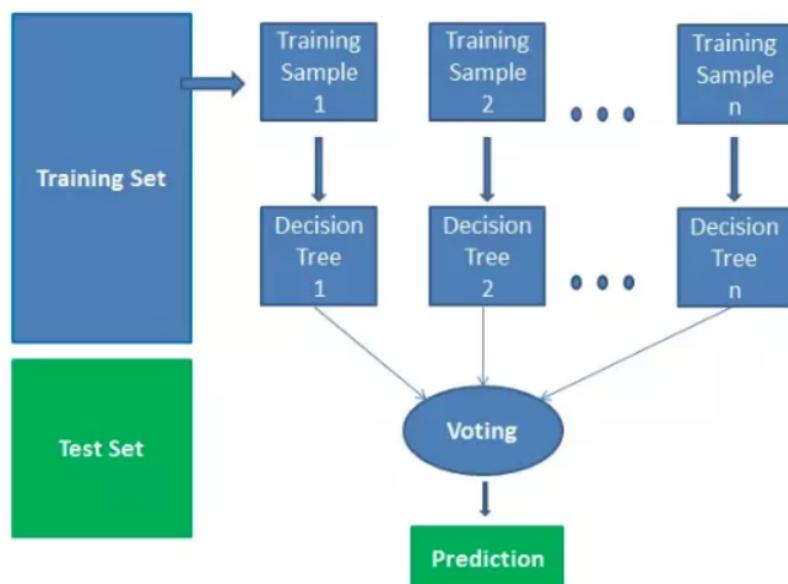
Random Forests[8] là thuật toán học có giám sát (supervised learning). Nó có thể được sử dụng cho cả phân lớp và hồi quy. Nó cũng là thuật toán linh hoạt và dễ sử dụng nhất. Random forests tạo ra cây quyết định trên các mẫu dữ liệu được chọn ngẫu nhiên, được dự đoán từ mỗi cây và chọn giải pháp tốt nhất bằng cách bỏ phiếu. Mỗi cây quyết định không được xây dựng từ toàn bộ tập dữ liệu cũng như không dùng tất cả các thuộc tính nên mỗi cây có thể dự đoán không tốt, khi đó mỗi cây quyết định không bị overfitting mà có thể bị underfitting. Tuy nhiên, kết quả cuối cùng của Random Forest lại tổng hợp từ nhiều cây quyết định, thế nên thông tin các cây sẽ bổ sung cho nhau, và kết quả thực nghiệm cho thấy mô hình có kết quả dự đoán tốt hơn so với 1 cây quyết định

Mô hình rừng cây được huấn luyện dựa trên sự phối hợp giữa luật kết hợp (ensembling) và quá trình lấy mẫu tái lập (bootstrapping). Cụ thể thuật toán này tạo ra nhiều cây quyết định mà mỗi cây quyết định được huấn luyện dựa trên nhiều mẫu con khác nhau và kết

quả dự báo là bầu cử (voting) từ toàn bộ những cây quyết định. Như vậy một kết quả dự báo được tổng hợp từ nhiều mô hình nên kết quả của chúng sẽ không bị chênh. Đồng thời kết hợp kết quả dự báo từ nhiều mô hình sẽ có phương sai nhỏ hơn so với chỉ một mô hình. Điều này giúp cho mô hình khắc phục được hiện tượng overfitting.

Giải thuật:

1. Chọn các mẫu ngẫu nhiên từ tập dữ liệu đã cho.
2. Thiết lập cây quyết định cho từng mẫu và nhận kết quả dự đoán từ mỗi quyết định cây
3. Bỏ phiếu cho mỗi kết quả dự đoán.
4. Chọn kết quả được dự đoán nhiều nhất là dự đoán cuối cùng.



Nguồn: <https://viblo.asia/p/phan-lop-bang-random-forests-trong-python-djeZ1D2QKWz>

4. Ensemble Learning

Ensemble Learning (EL) là một cách tiếp cận máy học giúp tăng hiệu suất dự đoán bằng cách kết hợp phép dự đoán từ nhiều mô hình khác nhau, giúp nâng cao tính tổng quát của mô hình học máy. Ý tưởng của việc kết hợp các mô hình khác nhau xuất phát từ một suy nghĩ hợp lý là: các mô hình khác nhau có khả năng khác nhau, có thể thực hiện tốt nhất các loại công việc khác nhau, khi kết hợp các mô hình này với nhau một cách hợp lý thì sẽ tạo thành một mô hình kết hợp mạnh có khả năng cải thiện hiệu suất tổng thể so với việc chỉ dùng các mô hình một cách đơn lẻ.

Các phương pháp Ensemble Learning được chia thành 3 loại sau đây:

- **Bagging:** Xây dựng một lượng lớn các mô hình (thường là cùng loại) trên những tập dữ liệu con khác nhau từ tập dữ liệu mẫu (lấy ngẫu nhiên dữ liệu trong tập dữ liệu mẫu để tạo 1 tập dữ liệu mới). Những mô hình này sẽ được huấn luyện độc lập và song song với nhau nhưng đầu ra của chúng sẽ được trung bình cộng để cho ra kết quả cuối cùng.
- **Boosting:** Xây dựng một lượng lớn các mô hình (thường là cùng loại). Tuy nhiên quá trình huấn luyện trong phương pháp này diễn ra tuần tự theo chuỗi. Trong chuỗi này mỗi mô hình sau sẽ học cách sửa những lỗi của mô hình trước (hay nói cách khác là dữ liệu mà mô hình trước dự đoán sai).
- **Stacking:** Xây dựng dựa trên việc phân chia mô hình thành các tầng, tầng cao sẽ là dung hợp của 1 hay nhiều mô hình ở cấp thấp hơn.

4.1 Max Voting

Voting ensemble [9] là phương pháp ensemble learning đơn giản nhất. Voting là tập hợp các mô hình khác nhau đưa ra dự đoán trên cùng tập dữ liệu, sử dụng phương pháp thống kê đơn giản.

- Trong bài toán Regression, Voting đưa ra mean hoặc median của các predictions từ các base-models
- Trong bài toán Classification, Voting sẽ sử dụng Hard-voting (Class được predicted nhiều nhất) hoặc Soft-voting (Class có tổng xác suất được predicted là cao nhất)

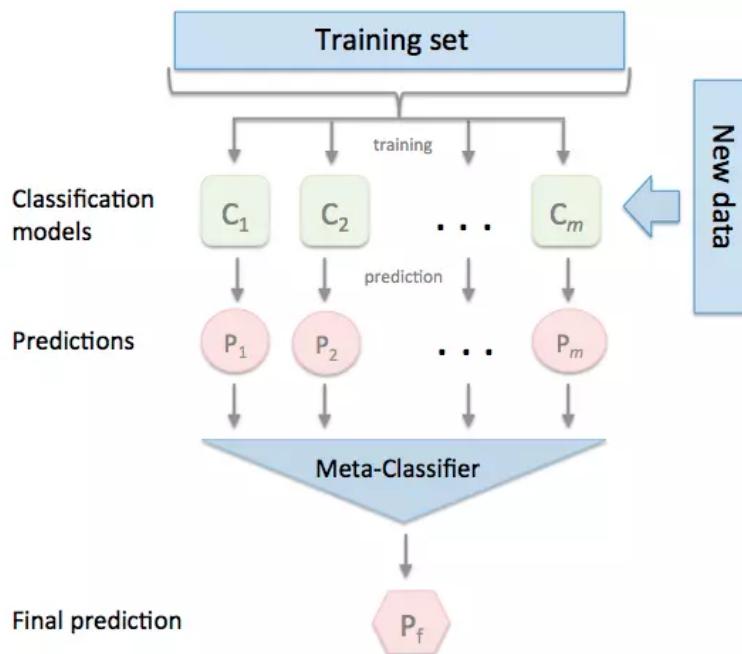
Trong Voting, tất cả các base-models được giả định có cùng độ quan trọng như nhau, cùng hiệu năng như nhau.

4.2 Stacked Generalization

Kỹ thuật này[10] sử dụng nhiều mô hình học máy khác nhau và học trên cùng một tập dữ liệu. Stacking sử dụng duy nhất một mô hình để đưa ra dự đoán tốt nhất từ các dự đoán của các mô hình khác. Để tránh vấn đề Overfitting, cần sử dụng kỹ thuật k-fold cross-validation cho tập dữ liệu. K-fold nghĩa là bộ dữ liệu sẽ được chia ra làm k phần bằng nhau từ bộ dữ liệu, và mô hình sẽ được huấn luyện k-lần, mỗi lần huấn luyện đc gọi là 1 run. Trong mỗi run, (k-1) phần sẽ được dùng để xây dựng mô hình huấn luyện và phần còn lại sẽ được dùng làm validation set. Kết quả cuối cùng là trung bình các kết quả thu được.

Giải thuật được chia làm 3 bước chính:

1. Sử dụng các base-models để học trên toàn bộ dữ liệu và đưa ra kết quả dự đoán ban đầu
2. Xây dựng bộ dữ liệu mới dựa trên outputs của các base-models
3. Huấn luyện Meta-model với bộ dữ liệu mới và đưa ra kết quả cuối cùng



Hình 14: Giải thuật Stacking

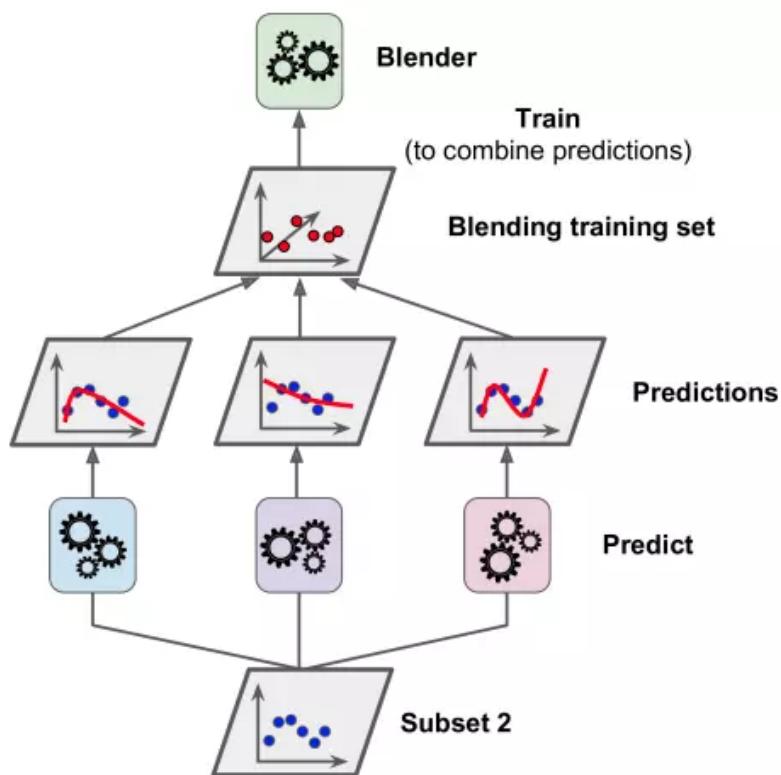
Nguồn:

<https://viblo.asia/p/1am-chu-stacking-ensemble-learning-Az45b0A6ZxY>

4.3 Blending

Với Stacking, vì chính việc sử dụng k-fold cross-validation khiến cho việc lần đầu tiếp cận với thuật toán khá là khó khăn, cho nên, Blending[10] sử dụng hold-out để chia training set làm 2 phần là subset-1 và subset-2. Về phương pháp hold-out, sẽ chia bộ dữ liệu ra làm 2 phần là training và validation (testing) một cách ngẫu nhiên, khi đó từ một bộ dữ liệu, ta đã có 2 bộ có thể dùng để kiểm thử hiệu năng mô hình của mình.

N-base-models sẽ được huấn luyện trên tập subset-1, sau đây các mô hình sẽ đưa dự đoán trên tập subset-2. subset-2 cùng với predictions của các base-models được sử dụng như các features để huấn luyện Meta-model.



Hình 15: Giải thuật Blending

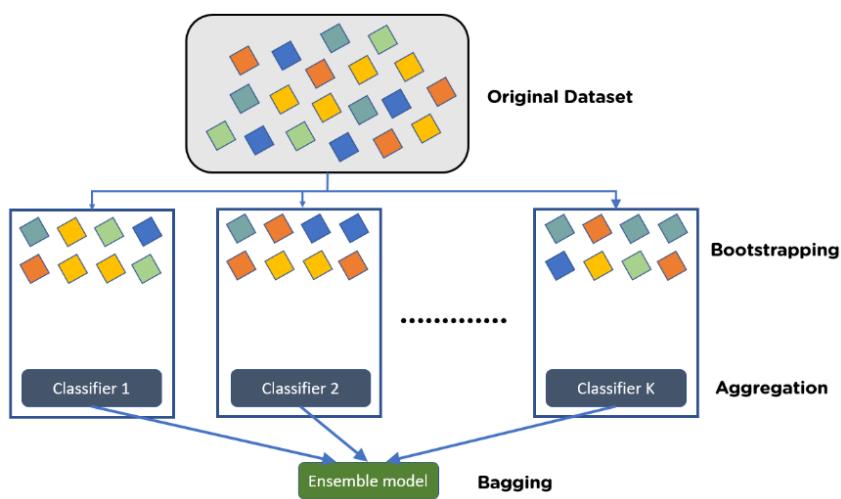
Nguồn:

<https://viblo.asia/p/lam-chu-stacking-ensemble-learning-Az45b0A6ZxY>

4.4 Bagging Classifier

Là kỹ thuật[11] ước tính phân loại dựa trên từng bộ tập training ngẫu nhiên của tập dữ liệu gốc thông qua một mô hình machine learning duy nhất và sau đó tổng hợp các dự đoán bằng cách voting hoặc lấy trung bình .

1. Sử dụng kỹ thuật Bootstrapping để tạo nên n mẫu dữ liệu ngẫu nhiên từ dữ liệu gốc
2. Tiến hành training n mẫu dữ liệu đó dựa trên mô hình đưa vào gọi là estimator_model
3. Đưa dữ liệu dự đoán và n mô hình đó và dùng kỹ thuật voting hoặc lấy trung bình để chọn ra kết quả cuối cùng.

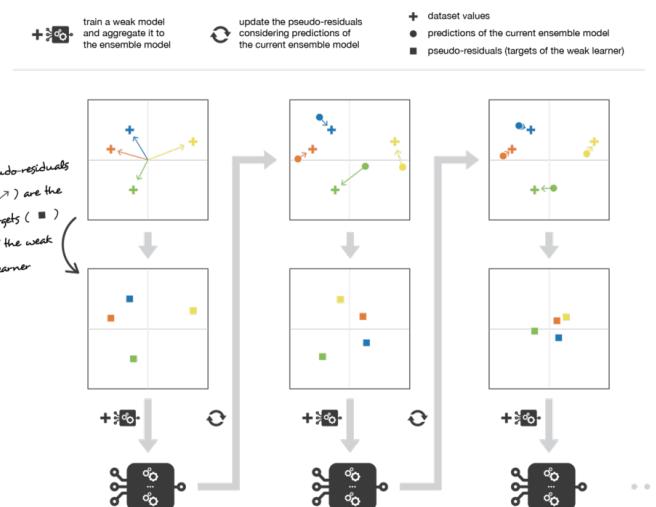


Hình 16: Kỹ thuật Bagging

Nguồn: <https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.BaggingClassifier.html>

4.5 Gradient Boosting Classifier

Thuật toán này[12] xây dựng một mô hình theo phong cách chuyển tiếp liên tục, nó cho phép tối ưu hóa hàm mất mát trong mỗi lần chuyển hóa. Gradient Boost cũng là một thuật toán tổng hợp sử dụng các phương pháp thúc đẩy (boosting) để phát triển một công cụ dự đoán nâng cao.



Hình 17: Kỹ thuật Gradient Boost

Nguồn: <https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.GradientBoostingClassifier.html>

Gradient Boost bắt đầu bằng cách xây dựng một cây để cố gắng phù hợp với dữ liệu và các cây tiếp theo được xây dựng nhằm mục đích giảm phần dư (lỗi).

4.6 Extreme Gradient Boosting

Extreme Gradient Boosting[12] là một giải thuật được base trên gradient boosting, tuy nhiên kèm theo đó là những cải tiến to lớn về mặt tối ưu thuật toán, về sự kết hợp hoàn hảo giữa sức mạnh phần mềm và phần cứng, giúp đạt được những kết quả vượt trội cả về thời gian training cũng như bộ nhớ sử dụng. Nó tương tự như Gradient Boost nhưng có một vài tính năng bổ sung làm cho nó mạnh hơn nhiều bao gồm:

- Sử dụng để cải thiện tính tổng quát của mô hình
- Cung cấp một tuyến đường trực tiếp đến cực tiểu thay vì giảm độ dốc, làm cho nó nhanh hơn nhiều
- Giảm mối tương quan giữa các cây, cuối cùng cải thiện sức mạnh của nhóm

4.7 Light Gradient Boosting

LightGBM[12] là một loại thuật toán thúc đẩy khác giúp tăng độ hiệu quả của mô hình và giảm mức sử dụng bộ nhớ. Điều làm cho LightGBM trở nên khác biệt là nó sử dụng một kỹ thuật độc đáo được gọi là Lấy mẫu một phía dựa trên Gradient (GOSS – Gradient-based One-Side Sampling) để lọc ra các cá thể dữ liệu nhằm tìm ra giá trị phân tách. Điều này khác với XGBoost sử dụng các thuật toán được sắp xếp trước và dựa trên biểu đồ để tìm ra sự phân chia tốt nhất. LightGBM sử dụng thuật toán leaf-wise tree growth, sự phân chia với mức tăng lớn nhất sẽ được chọn làm nhánh tiếp theo, bắt kể độ sâu của nó.

5. Thông số đánh giá

5.1 Confusion Matrix

Sau khi xử lý dữ liệu, sử dụng mô hình cho ra kết quả thì confusion matrix sẽ giúp đánh giá độ hiệu quả của mô hình đó. Có thể hiểu confusion matrix là một phương pháp đánh giá kết quả của những bài toán phân loại với việc xem xét cả những chỉ số về độ chính xác và độ bao quát của các dự đoán cho từng lớp. Một confusion matrix gồm 4 chỉ số sau đối với mỗi lớp phân loại:

Bảng trên hình 18 là một ma trận 2 chiều: Giá trị thực tế và giá trị dự đoán.

- **True Positive (TP):** Các giá trị thực sự Positive và được dự đoán là Positive.
- **False Positive (FP):** Các giá trị thực sự là Negative nhưng được dự đoán sai là Positive.

		Predicted Class	
		True Positive (TP)	False Negative (FN)
True Class	True Positive (TP)		
	False Positive (FP)	True Negative (TN)	

Hình 18: Confusion matrix cho phân loại nhị phân

Nguồn:

<https://www.sciencedirect.com/topics/engineering/confusion-matrix>

- **False Negative (FN):** Các giá trị thực sự là Positive nhưng được dự đoán sai là Negative.
- **True Negative (TN):** Các giá trị thực sự Negative và được dự đoán là Negative.

Những chỉ số trên sẽ là cơ sở quan trọng giúp tính toán những thông số accuracy, precision, recall, f1-score và AUC.

5.2 Accuracy

Một trong những số liệu phổ biến nhất trong khi đánh giá độ chính xác của mô hình là accuracy. Công thức tính toán accuracy của mô hình như sau:

$$\text{Accuracy} = \frac{TN + TP}{TN + TP + FP + FN}$$

Tuy nhiên, một mô hình có độ chính xác cao chưa hẳn đã tốt. Accuracy lộ rõ hạn chế khi được sử dụng trên bộ dữ liệu không cân bằng (imbalanced dataset). Trong trường hợp này độ chính xác của mô hình, tức là tỉ lệ số lượng dự đoán đúng so với tổng số trang web.

5.3 Precision

Precision trả lời cho câu hỏi trong các trường hợp được dự báo là positive thì có bao nhiêu trường hợp là đúng? Và tất nhiên precision càng cao thì mô hình của chúng ta càng tốt trong việc phân loại. Công thức của precision như sau:

$$\text{Precision} = \frac{TP}{TP + FP}$$

Trong trường hợp này precision cho biết tỉ lệ dự đoán đúng trang web giả mạo thật sự trong tổng số trang web được dự đoán là giả mạo. Giá trị này càng cao thì chứng tỏ tỉ lệ dự đoán chính xác trang web giả mạo của mô hình càng cao

5.4 Recall

Recall trả lời cho câu hỏi trong các trường hợp ta đoán đúng là positive, thì số lượng ta đoán đúng chiếm bao nhiêu phần trăm trong tổng số positive thực sự.

$$\text{Recall} = \frac{TP}{TP + FN}$$

Trong trường hợp này recall cho biết tỉ lệ mà mô hình có thể dự đoán đúng số trang web giả mạo trên tổng số trang web giả mạo thực tế. Giá trị này càng cao thì chứng tỏ tỉ lệ mà mô hình có thể xác định được càng nhiều trang web giả mạo hơn.

5.5 F1-score

Để hiểu rõ F1-score, ta cần nắm được tầm quan trọng của Precision và Recall. Nếu không có recall, chúng ta sẽ không tin tưởng lắm vào kết quả dự đoán “không” cho vấn đề đang xét. Còn nếu không có precision, chúng ta sẽ không tin tưởng vào các kết quả dự đoán “có” cho vấn đề đang xét. Để xác định trang web có phải là giả mạo hay không, hệ thống phải có một ngưỡng nào đó để xác định liệu trang web đó có phải là giả mạo hay không. Ngưỡng đó được xây dựng trên cơ sở precision và recall - F1-score.

$$F_1 = \frac{2 \cdot \text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}}$$

Để nhận thấy, giá trị của f1-score rơi vào khoảng $(0, 1]$. Thông thường, f1-score nhận giá trị càng cao, độ phân lớp và mô hình của chúng ta càng tốt. Trong trường hợp chúng ta quan tâm một trong hai recall và precision hơn, chúng ta cần dùng công thức tổng quát của f1-score.

$$F_\beta = (1 + \beta^2) \frac{\text{precision} \cdot \text{recall}}{\beta^2 \cdot \text{precision} + \text{recall}}$$

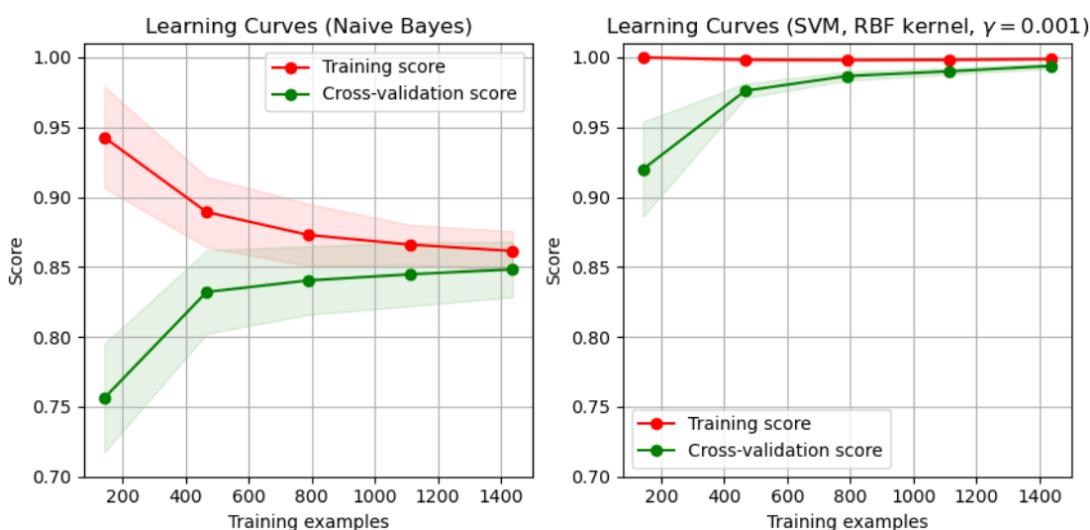
Để nhận thấy, f1-score chỉ là trường hợp đặc biệt khi $\beta=1$. Nếu chúng ta coi trọng precision hơn recall, ta chọn $\beta<1$. Ngược lại, nếu recall được coi trọng hơn precision, ta chọn $\beta>1$.

6. Learning curve

Learning curve hay đường cong học tập là một biểu đồ về hiệu suất của mô hình mạng học máy theo thời gian. Các đồ thị đường cong học tập cung cấp cái nhìn sâu sắc về động lực học tập của mô hình, chẳng hạn như liệu mô hình có đang học tốt hay không, liệu nó

có underfitting với tập dữ liệu đào tạo hay không hoặc nó có overfitting với tập dữ liệu đào tạo hay không.

Trước tiên, bạn phải cập nhật lệnh gọi của mình thành hàm fit để bao gồm tham chiếu đến tập dữ liệu xác thực. Tập xác thực là một phần của tập huấn luyện không được sử dụng để đào tạo mô hình mà thay vào đó được sử dụng để đánh giá hiệu suất của mô hình trong quá trình huấn luyện. Bạn có thể chia nhỏ dữ liệu theo cách thủ công và chỉ định đối số validation_data hoặc bạn có thể sử dụng đối số validation_split và chỉ định phần trăm phân chia của tập dữ liệu đào tạo và để API thực hiện phân tách cho bạn.



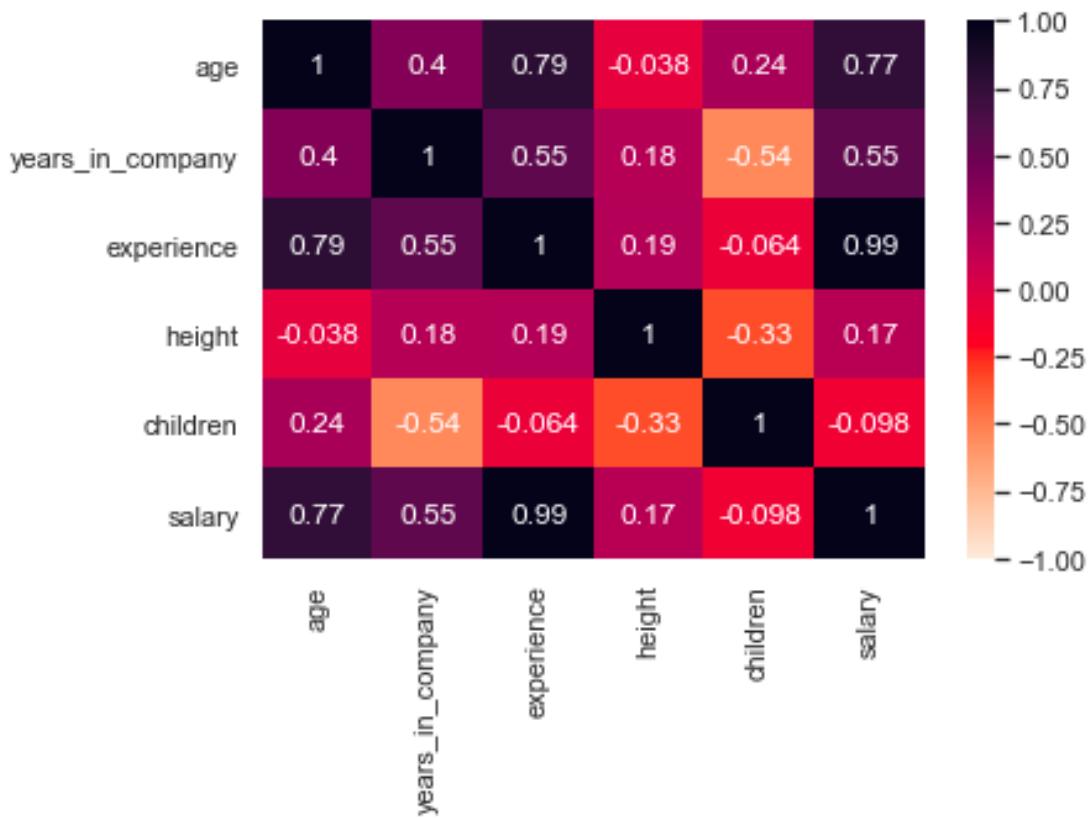
Hình 19: Đồ thị học tập Naive Bayes và SVM

Nguồn: https://scikit-learn.org/stable/modules/learning_curve.html

Đường cong học tập hiển thị training score và validation-score trên từng dữ liệu mẫu khác nhau. Nó là một công cụ giúp ta hiểu xem ta được lợi bao nhiêu từ việc tập dữ liệu ngày càng lớn và là công cụ ước tính xem mô hình chúng ta xây dựng có mắc các lỗi variable-error hay bias-error. Xem xét ví dụ sau khi sử dụng đồ thị học tập cho mô hình phân loại Naive Bayes và SVM: Đối với mô hình Naive Bayes, cả điểm validation-score và training-score đều dần hội tụ tại một điểm khá thấp khi tập dữ liệu ngày càng tăng. Do đó chúng ta sẽ không nhận được gì nhiều lợi ích khi đào tạo mô hình với dữ liệu lớn hơn. Ngược lại, với mô hình SVM khi đào tạo cùng lượng tập dữ liệu thì training-score lớn hơn nhiều so với validation-score. Việc tập dữ liệu ngày càng tăng sẽ cho độ chính xác của mô hình tăng thêm. So sánh 2 mô hình trên với nhau ta thấy được mô hình SVM sẽ hiệu quả hơn trong việc phát triển với tập dữ liệu lớn hơn.

7. Hệ số tương quan (Correlation coefficient)

Hệ số tương quan, hay còn được gọi là hệ số tương quan Pearson, là một đại lượng thống kê được sử dụng để đo độ tương quan tuyến tính giữa hai biến số đại diện cho hai đặc trưng. Hệ số tương quan nằm trong khoảng từ -1 đến 1, với giá trị gần -1 cho thấy mối tương quan âm mạnh giữa hai biến số, giá trị gần 1 cho thấy mối tương quan dương mạnh giữa hai biến số, và giá trị gần 0 cho thấy không có mối tương quan tuyến tính giữa hai biến số.



Hình 20: Ví dụ về hệ số tương quan

Nguồn:

<https://www.datacamp.com/tutorial/tutorial-datails-on-correlation>

Hệ số tương quan là một công cụ hữu ích để lựa chọn đặc trưng trong các bài toán Machine Learning, khi ta muốn tìm ra các đặc trưng quan trọng nhất để đưa vào mô hình. Nếu hai đặc trưng có hệ số tương quan cao, có thể rút ra được rằng chúng liên quan mật thiết với nhau và ta có thể giảm bớt số lượng đặc trưng để tăng tính đơn giản và hiệu quả của mô hình. Tuy nhiên, cần lưu ý rằng hệ số tương quan chỉ đo độ tương quan tuyến tính giữa hai biến số, không phải đo độ tương quan phi tuyến tính hoặc quan hệ nguyên nhân và kết quả. Vì vậy, việc lựa chọn đặc trưng chỉ dựa trên hệ số tương quan không đảm bảo rằng mô hình sẽ hoạt động tốt với tập dữ liệu mới.

Tóm lại, hệ số tương quan là một công cụ hữu ích trong việc lựa chọn đặc trưng để đưa vào mô hình Machine Learning. Tuy nhiên, cần kết hợp với các phương pháp khác để đảm bảo tính chính xác và độ tin cậy của mô hình.

8. Ngôn ngữ lập trình sử dụng

8.1 Python

Python là ngôn ngữ lập trình thông dịch bậc cao (high-level interpreted language) cho các mục đích lập trình đa năng, do Guido van Rossum tạo ra và lần đầu ra mắt vào năm 1991. Python là một ngôn ngữ lập trình được sử dụng rộng rãi trong các ứng dụng web, phát triển phần mềm, khoa học dữ liệu và máy học (ML) vì những tính năng:

- Dễ đọc, dễ học và dễ duy trì do có hình thức sáng sửa và cấu trúc rõ ràng
- Cung cấp một thư viện tiêu chuẩn lớn
- Cung cấp chế độ tương tác (interactive mode) để kiểm thử và sửa lỗi
- Chạy trên đa dạng các nền tảng phần cứng khác nhau
- Dễ dàng để mở rộng bằng cách thêm các module bậc thấp vào trình thông dịch

Python được hỗ trợ rất mạnh về học máy (machine learning), học sâu (deep learning) với các thư viện như TensorFlow, Pytorch, Keras,... Bên cạnh đó, Python còn có các thư viện hỗ trợ mạnh cho việc xử lý dữ liệu như là Pandas, Numpy, Scipy,...

8.2 HTML

HTML (HyperText Markup Language) là ngôn ngữ dùng để xây dựng lên nội dung của một trang web như đoạn văn, đường dẫn, hình ảnh, video, bảng, form,... HTML được ra mắt vào năm 1991 do Tim Berners-Lee tạo ra. HTML là sự kết hợp của ngôn ngữ siêu văn bản (Hypertext) và ngôn ngữ đánh dấu (Markup), trong đó ngôn ngữ siêu văn bản xác định sự liên kết giữa các trang web, còn ngôn ngữ đánh dấu được sử dụng để xác định tài liệu văn bản bên trong các thẻ định nghĩa cấu trúc của trang web. HTML sử dụng các thẻ (tag) và thuộc tính (attribute). Thẻ này thể hiện một chức năng nhất định trong HTML. Các thuộc tính để xác định các thao tác cần được thực hiện với văn bản, hình ảnh và các nội dung khác, nhằm hiển thị chúng ở một định dạng nhất định.

8.3 CSS

CSS (Cascading Style Sheets) là ngôn ngữ dùng để thiết kế, định dạng, chia bố cục, phong cách của các nội dung trong trang web. CSS được sử dụng để đơn giản hóa quy

trình làm cho trang web trở nên bắt mắt và ấn tượng hơn. Bên cạnh đó, việc định dạng bằng CSS có thể tách biệt với HTML, khiến cho việc tái sử dụng các định dạng này để áp dụng cho các trang web khác nhau là hoàn toàn khả thi, giúp giảm thời gian thiết kế trang web.

8.4 Javascript

Javascript là ngôn ngữ lập trình thông dịch, dễ học dễ sử dụng, có độ phổ biến cao do sử dụng được cả ở bên phía khách (client) và máy chủ (server). Về phía khách, Javascript được hỗ trợ bởi hầu hết mọi trình duyệt web, sử dụng để làm trang web trở nên sinh động, giúp tương tác với các nội dung của trang web. Về phía máy chủ, với sự xuất hiện của Nodejs - trình thông dịch thực thi mã Javascript giúp nó chạy được trên máy dưới dạng một ứng dụng độc lập, Javascript giờ đây được sử dụng để viết các ứng dụng của máy chủ. Javascript kết hợp với HTML và CSS tạo thành bộ ba ngôn ngữ căn bản nhất để xây dựng nên một trang web.

9. Các frameworks và thư viện sử dụng

9.1 Flask

Flask là một micro web application framework của Python, cho phép xây dựng các ứng dụng web trong một khoảng thời gian ngắn và có thể phát triển quy mô của ứng dụng tùy theo yêu cầu. Flask cung cấp một lõi chức năng súc tích nhất cho các ứng dụng web theo triết lý 'micro', nhưng có thể dễ dàng mở rộng bất cứ khi nào theo nhu cầu sử dụng và triển khai với các thành phần tiện ích mở rộng như: cơ sở dữ liệu (database), xác thực biểu mẫu, xử lý upload, các công nghệ xác thực, template, email, RESTful API, ... Flask dựa trên Werkzeug (một thư viện tiện ích WSGI) và Jinja2 (template engine).

Flask phù hợp với việc xây dựng các ứng dụng web có quy mô vừa và nhỏ, các API và web service:

- Cấu trúc gọn gàng và rõ ràng khi xây dựng các ứng dụng web thông qua việc viết các module Python.
- Kiến trúc nhỏ, gọn, không bị bó buộc bởi bộ khung cồng kềnh.
- Linh hoạt sử dụng các thành phần (component) cần thiết trong hệ thống tùy thuộc vào đặc điểm và tính năng riêng của ứng dụng web.

Với những lý do trên, nhóm sử dụng framework Flask để xây dựng server trung gian (Intermediary server).

9.2 Numpy

Numpy là một thư viện toán học phổ biến và mạnh mẽ của Python, cho phép làm việc hiệu quả với ma trận và mảng, đặc biệt là dữ liệu ma trận và mảng lớn với tốc độ xử lý nhanh hơn nhiều lần khi chỉ sử dụng “core Python” đơn thuần. Mục đích Numpy được tạo ra nhằm xử lý các khối lượng dữ liệu mảng và ma trận lớn, do đó các hàm được sử dụng trên đây đều được tối ưu hơn so với các hàm Python thông thường.

Những chức năng có thể thao tác khi sử dụng Numpy:

- Các phép toán học và logic trên mảng.
- Các phép biến đổi tuyến tính, biến đổi nghịch đảo và các phương thức thay đổi shape.
- Các phép biến đổi đại số tuyến tính của ma trận.

9.3 Pandas

Pandas là một thư viện Python cung cấp các cấu trúc dữ liệu nhanh, mạnh mẽ, linh hoạt và mang hàm ý. Pandas được thiết kế để làm việc dễ dàng và trực quan với dữ liệu có cấu trúc (dạng bảng, đa chiều, có tiềm năng không đồng nhất) và dữ liệu chuỗi thời gian. Pandas đặc biệt hữu dụng khi được ứng dụng trong khoa học dữ liệu (data science) do hỗ trợ sẵn nhiều chức năng cũng như dễ thao tác, dễ sử dụng và thời gian thực thi nhanh. Hai thành phần quan trọng nhất của Pandas là Series và DataFrame - thực thể lưu trữ dữ liệu khi xử lý và đi kèm theo các chức năng trên các thực thể như thống kê, làm sạch, trực quan hóa dữ liệu,... Ngoài ra, Pandas còn hỗ trợ nhập (import) và xuất (export) dữ liệu với nhiều định dạng khác nhau như txt, csv, xlsx, db/sql,... khiến cho việc lưu trữ trở nên dễ dàng hơn.

9.4 Scikit-learn

Scikit-learn (Sklearn) là thư viện mạnh mẽ dành cho các thuật toán học máy được viết trên ngôn ngữ Python. Thư viện cung cấp một tập các công cụ xử lý các bài toán học máy và các bài toán mô hình thống kê gồm: phân loại (classification), hồi quy (regression), phân cụm (clustering), và giảm chiều dữ liệu (dimensionality reduction).

Scikit-learn được nhóm sử dụng trong việc phân tách tập dữ liệu train, test, validate và đánh giá độ hiệu quả của mô hình thông qua các metrics như: confusion matrix, accuracy, precision, recall, f1

9.5 Whois

Whois giúp người dùng truy cập vào cơ sở dữ liệu lưu trữ thông tin liên quan đến tên miền, chẳng hạn như: thông tin liên hệ của người đăng ký, quản trị viên và người phụ trách kỹ thuật trong gói đăng ký, tổ chức đăng ký tên miền có vai trò tài trợ, ngày tạo, cập nhật và ngày hết hạn, máy chủ định danh và trạng thái miền. Thông tin lưu trữ trong cơ sở dữ liệu WHOIS có thể công khai hoặc không công khai tùy vào chế độ cài đặt của người dùng. Tổ chức quản lý tên miền này cũng kiểm soát những trường sẽ hiển thị. Cho nên sẽ có một vài tên miền không truy cập được do tính bảo mật của tổ chức quản lý tên miền đó.

9.6 Công cụ Google Colab

Colaboratory hay còn gọi là Google Colab, là một sản phẩm từ Google Research, nó cho phép thực thi Python trên nền tảng đám mây dựa trên Jupyter Notebook. Google Colab cung cấp cho người dùng CPU/GPU miễn phí và dung lượng bộ nhớ lớn để lưu trữ. Colab cung cấp nhiều loại GPU, thường là Nvidia K80s, T4s, P4s và P100s. Loại GPU được cấp phát ở mỗi phiên hoạt động là ngẫu nhiên. Mỗi phiên cũng có giới hạn về thời gian hoạt động. Các cấu hình phần cứng mà Google Colab cung cấp được mô tả ở bảng 1.

CPU	GPU	TPU
Intel Xeon Processor with two cores @ 2.30 GHz and 13 GB RAM	Up to Tesla K80 with 12 GB of GDDR5, Intel Xeon Processor with two cores @ 2.20 GHz and 13 GB RAM VRAM	Cloud TPU with 180 teraflops of computation, Intel Xeon Processor with two cores @ 2.30 GHz and 13 GB RAM

Bảng 1: Cấu hình phần cứng Google Colab cung cấp

Nguồn: <https://codelearn.io/sharing/google-colab-la-gi>

10. Azure App Service

Azure App Service là một dịch vụ HTTP dùng để lưu trữ ứng dụng web, API REST và mobile backends. Người dùng có thể phát triển ứng dụng bằng ngôn ngữ lập trình mà họ yêu thích như .NET, .NET Core, Java, Ruby, Node.js, PHP hoặc Python. Ứng dụng có thể dễ dàng chạy và mở rộng trên cả hai môi trường là Windows và Linux.

Không chỉ cung cấp sức mạnh của Microsoft Azure cho ứng dụng, App Service còn có những tính năng bảo mật, cân bằng tải, tự động cân bằng và quản lý tự động. Bên cạnh đó, người dùng còn có thể tận dụng các tính năng DevOps của nó, bao gồm triển khai liên tục từ Azure DevOps, GitHub, Docker Hub và các nguồn khác, hệ thống quản lý gói, môi trường dàn dựng, tên miền tùy chỉnh và chứng chỉ SSL.

Với Azure App Service, người dùng chỉ cần thanh toán cho số lượng tài nguyên Azure mà họ sử dụng. Số lượng tài nguyên Azure mà người dùng sử dụng được xác định bởi gói App Service mà họ chọn để sử dụng cho ứng dụng của mình.

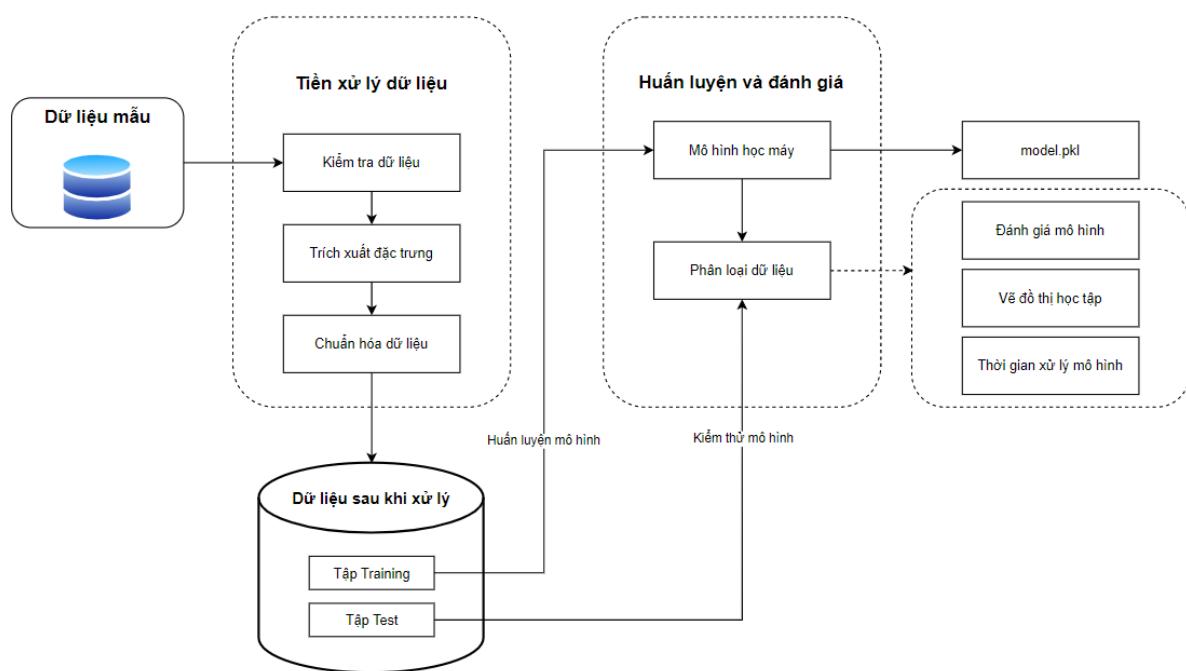
III. Thiết kế xây dựng mô hình học máy

1. Mô hình Machine Learning phát hiện giả mạo

Trong phần này, việc thiết kế mô hình học máy giúp phát hiện lừa đảo, cũng như nghiên cứu [13] sẽ được tập trung vào 2 giai đoạn chính bao gồm: **Đào tạo mô hình phân loại** và **Phát hiện giả mạo dựa trên mô hình phân loại**.

1.1 Đào tạo mô hình phân loại

Như đã thấy ở hình 21, giai đoạn này sẽ bao gồm giai đoạn tiền xử lý (pre-processing) và giai đoạn đào tạo (training). Ở giai đoạn tiền xử lý ta cần chuẩn bị tập dữ liệu là các địa chỉ URL với các đặc tính được chọn từ trước. Các địa chỉ trang Web trong tập dữ liệu phải được đánh dấu là giả mạo hay không để dễ dàng xử lý. Từ các đặc tính đã được chọn lọc, ta trực quan hóa dữ liệu giúp xác định được đặc tính cụ thể của một trang Web giả mạo và một trang Web hợp lệ khác nhau như thế nào, tóm tắt các đặc tính tương đồng đó và lọc lại dữ liệu dựa trên các đặc tính đó. Sau khi lọc dữ liệu và phân loại dựa trên đặc điểm của từng đặc tính, tập dữ liệu sẽ được phân ra thành tập dữ liệu để training và tập dữ liệu dùng để kiểm tra độ chính xác. Với tập dữ liệu dùng để training, ta đưa vào một mô hình phân loại cụ thể để huấn luyện mô hình đó, sau đó dùng tập test để kiểm tra độ chính xác của mô hình.



Hình 21: Quy trình huấn luyện mô hình, tham khảo [13]

Algorithm 1: Hàm tiền xử lý dữ liệu

Data: Tập dữ liệu mẫu URL được gán nhãn

Result: Tập dữ liệu đã chứa các đặc trưng của từng URL

- 1 Hàm thực thi tiền xử lý (*tập dữ liệu mẫu*):
 - 2 Tập dữ liệu mẫu \leftarrow Lọc các URL còn truy cập được
 - 3 Tập dữ liệu mẫu \leftarrow kiểm tra các dữ liệu bị trùng
 - 4 Tập dữ liệu mẫu \leftarrow kiểm tra tính hợp lệ cấu trúc của URL
 - 5 *result[]* \leftarrow Khởi tạo dataframe rỗng
 - 6 For each **URL**:
 - 7 Trích xuất đặc trưng(*URL*)
 - 8 Chuẩn hóa dữ liệu(*URL*)
 - 9 *result[]* \leftarrow chuỗi các đặc trưng đã được xử lý
 - 10 Trả về dataframe chứa các đặc trưng của tập dữ liệu mẫu
-

Algorithm 2: Huấn luyện mô hình

Data: Tập dữ liệu mẫu URL đã được gán nhãn

Result: Mô hình đã được huấn luyện dựa trên tập dữ liệu mẫu

1 Begin

2 Tập dữ liệu chứa các đặc trưng của URL ←

 Tiền xử lý tập dữ liệu (*tập dữ liệu mẫu*)

3 Tách dữ liệu thành tập training và tập test ←

 80% cho training và 20% dùng để test

4 For each Model:

5 Model ← Huấn luyện mô hình dựa trên tập training đã được phân chia

6 Độ chính xác ← Đánh giá độ chính xác thông qua tập test

7 Đồ thị học tập ← Vẽ đồ thị học tập thông qua điểm số của tập training

8 Thời gian xử lý ← Test 1 URL bất kỳ để kiểm tra tốc độ xử lý của mô hình

9 model.pkl ← Lưu mô hình vào file .pkl thông qua thư viện joblib

10 End

1.2 Phát hiện giả mạo dựa trên mô hình phân loại

Theo như hình 22, mô hình sau khi được huấn luyện có thể được dùng để phát hiện các trang web giả mạo. Khi người dùng mở một trang web mới, trang Web đó với mã URL sẽ được xử lý dữ liệu thông qua các đặc tính được chọn, các dữ liệu từ thanh địa chỉ đó sẽ được đưa qua mô hình mà ta đã chọn để phân loại xem nó có phải là giả mạo hay không. Khi phân loại xong, URL đó sẽ được dán nhãn và từ đó hệ thống dựa vào nhãn đó mà cảnh báo đến cho người dùng.

Algorithm 3: Phân loại giả mạo dựa trên mô hình phân loại

Data: URL cần được phân loại

Result: Kết quả phân loại của URL đó

1 Begin

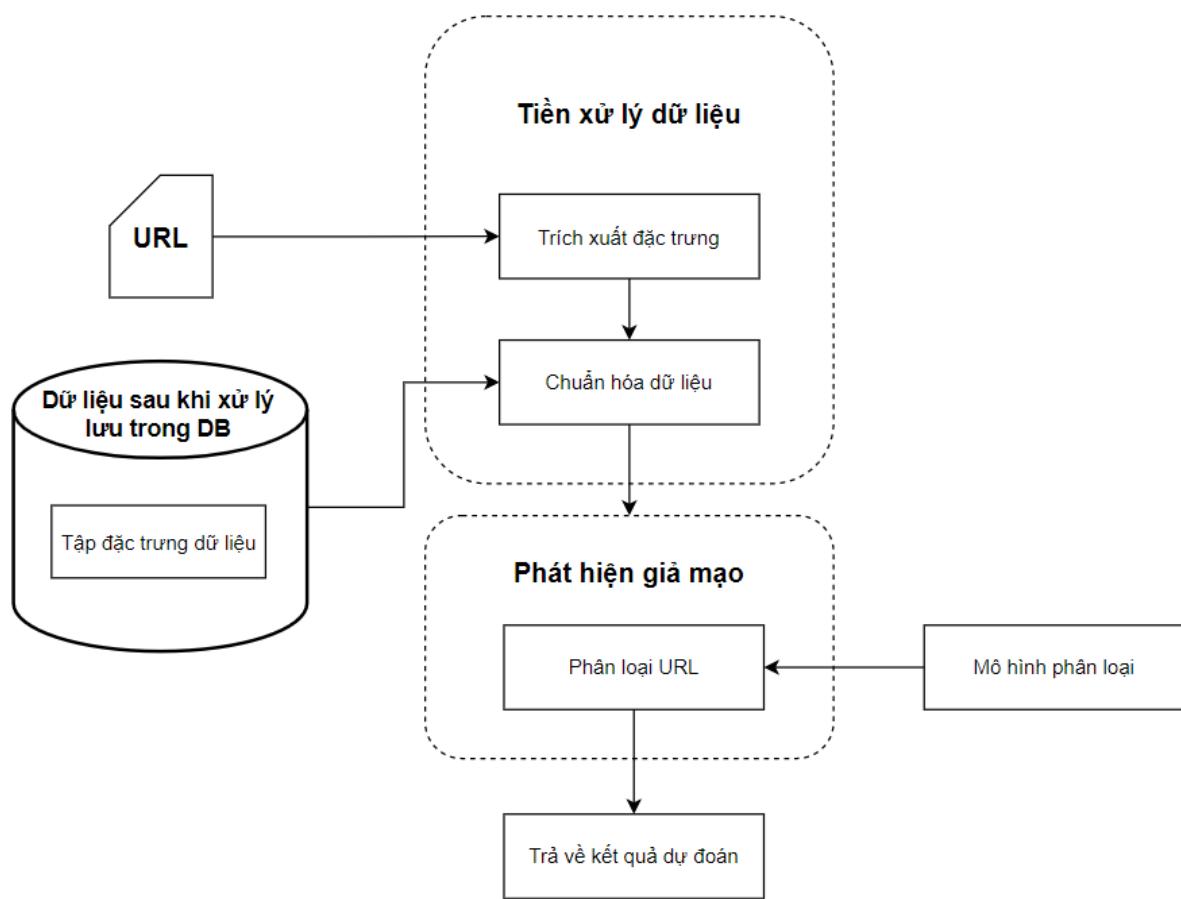
2 Dãy các đặc trưng của URL ← trích xuất đặc trưng (**URL**)

3 Dãy đặc trưng chuẩn hóa ← Chuẩn hóa dữ liệu (*Dãy các đặc trưng của URL*)

4 Kết quả phân loại ← Dự đoán qua mô hình học máy (*Dãy đặc trưng chuẩn hóa*)

5 Trả về kết quả phân loại

6 End

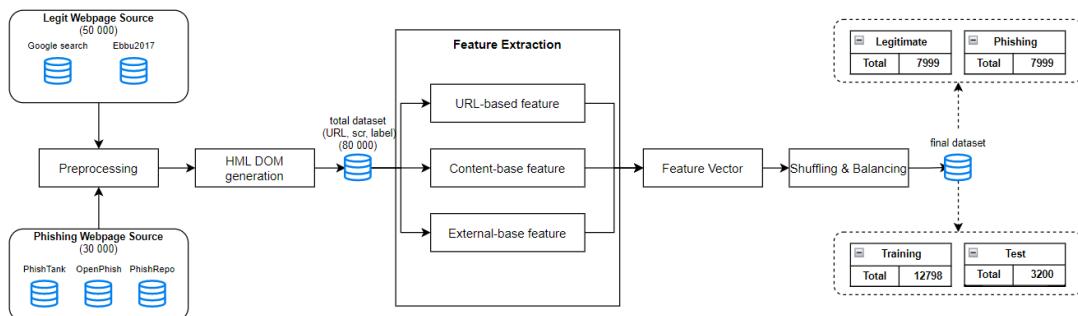


Hình 22: Quy trình phát hiện giả mạo, tham khảo [13]

2. Mô tả dữ liệu

2.1 Tổng quan về tập dữ liệu

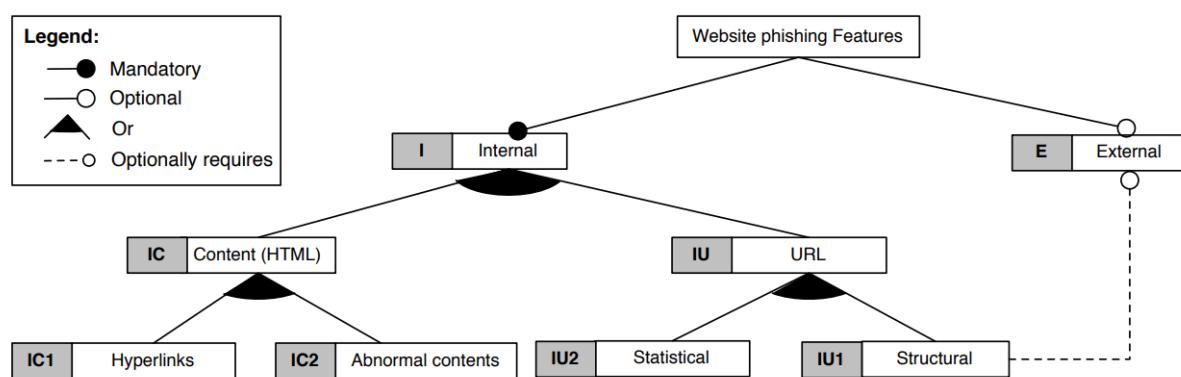
Tập dữ liệu mà chúng tôi sử dụng được tham khảo từ [14]. Tập dữ liệu là một tập hợp bao gồm các trang web hợp lệ và trang web giả mạo chính thống. Tập dữ liệu này sẽ đóng vai trò giúp đào tạo mô hình cũng như dùng để đánh giá hiệu năng giữa các thuật toán khác nhau.



Hình 23: Quy trình xây dựng bộ dữ liệu

2.2 Phân loại các đặc trưng của trang web

Thông qua các nghiên cứu được tham khảo từ [15], các đặc trưng của một trang web có thể phân loại các đặc tính khác nhau của một trang web lừa đảo. Sơ đồ 24 sẽ hiển thị các lớp đặc tính phổ biến được áp dụng cho việc phân loại các trang web lừa đảo. Phân loại này dùng để xếp loại các đặc tính chung với nhau, từ đó dễ dàng cho việc nghiên cứu.



Hình 24: Sơ đồ phân loại các đặc tính

Các hệ thống chống lừa đảo hiện nay dựa trên machine learning được thấy đều sử dụng các đặc trưng bên trong và bên ngoài của một trang web. Để quá trình nghiên cứu được dễ dàng, quá trình trích xuất đặc tính hoàn toàn tự động bằng cách sử dụng những câu lệnh của Python. Tổng hợp các đặc tính lại, tìm ra những đặc tính đặc trưng nhất của một trang web giả mạo. Những đặc tính của một trang web sẽ được phân loại theo như bảng 24.

- URL-based features:** URL-based features có được dựa trên việc phân tích địa chỉ URL. Các đặc trưng này có thể mang số liệu thống kê hoặc mang tính cấu trúc (vị trí).
- Content-base features:** Content-based features được trích xuất bằng cách tải các trang web và xem nội dung của HTML. Từ đó các đặc tính dựa trên sự bất thường hoặc các tập lệnh thực hiện các hành vi đáng ngờ được xác định.
- External-based features:** External-based features được xác định bằng cách truy vấn vào các dịch vụ bên thứ ba để tham chiếu từ đó phát hiện ra các trang web đáng nghi.

2.3 Tầm quan trọng việc chọn lọc những đặc trưng

Việc xác định những đặc tính quan trọng là một trong những kỹ thuật quan trọng giúp mô hình phân loại trở nên tốt hơn. Xác định và loại bỏ những đặc điểm dư thừa không cần thiết trong tập đào tạo còn được gọi là kỹ thuật "feature selection". Một khi các đặc trưng quan trọng được xác định, các đặc trưng khác ít quan trọng hơn có thể được loại bỏ mà không ảnh hưởng đến hiệu suất phân loại của mô hình. Có rất nhiều kỹ thuật giúp chọn ra những đặc trưng quan trọng, những thuật toán phổ biến bao gồm: Filter methods, Wrapper methods và Embedded methods. Những lợi ích trong việc tìm ra những đặc trưng quan trọng:

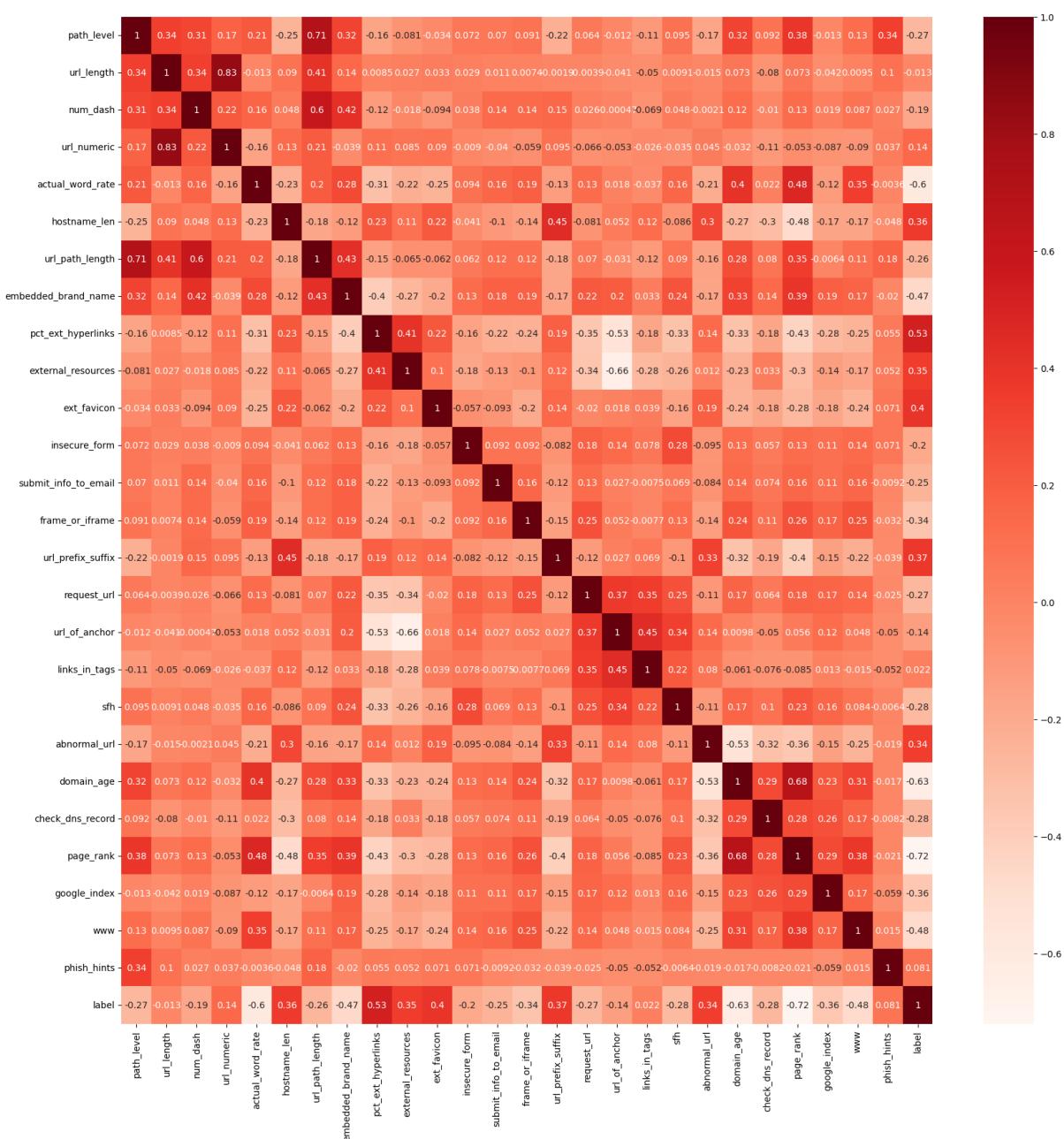
- **Mô hình đơn giản hơn:** Các mô hình đơn giản hơn, không quá phức tạp tránh xảy ra overfitting.
- **Làm ngắn thời gian đào tạo mô hình:** Tập hợp các đặc tính cần thiết sẽ giúp tăng tốc độ đào tạo mô hình.
- **Giảm nhiễu từ tập dữ liệu:** Loại bỏ những đặc trưng ít quan trọng giúp tăng độ chính xác của mô hình và loại bỏ những yếu tố gây nhiễu không quá quan trọng.

Các dữ liệu quan trọng phổ biến thường: các biến số (int, float,...), các biến phân loại (bool, ordinal, nominal,...). Bên cạnh đó còn có các thư viện như sklearn hay R giúp chọn lọc những đặc tính quan trọng.

2.4 Những đặc tính quan trọng giúp phát hiện trang web giả mạo

Dựa vào tập dữ liệu mà từ trước, sau quá trình trích xuất dữ liệu từ dữ liệu mẫu. Chúng ta cần làm rõ các đặc trưng cần thiết giúp nhận diện một trang Web giả mạo thông qua ma trận tương quan của dữ liệu.

Từ ma trận tương quan trên, chọn ra những đặc tính có tỉ lệ tương quan với "label" trên 0.2. Từ đó visualize đặc trưng đó và tìm điểm khác biệt của trang Website Legit và Phishing.



Hình 25: Ma trận tương quan của đặc trưng

URL-based features

STT	Feature	Type	Mô tả
1	Path_level	Int	Trả về mức độ truy cập của một trang Web. Đường dẫn thường nằm phía sau hostname và được phân tách bằng dấu "/"
2	Url_length	Int	Những kẻ lừa đảo có thể che giấu các thành phần đáng ngờ của website giả mạo bằng cách sử dụng URL có độ dài lớn.

3	Num_dash	Int	Đếm số lượng dấu "-" trong địa chỉ Url. Những trang web giả mạo thường có số lượng "-" bất thường.
4	Url_numeric	Int	Đếm số lượng chữ số mà Url sử dụng. Số lượng bất thường của các chữ số xuất hiện.
5	Actual_word_rate	Float	Tỷ lệ các chữ có nghĩa xuất hiện trên URL. Thông thường những kẻ lừa đảo sẽ sử dụng những từ ngữ random để tạo website thông qua API của Datamuse
6	Hostname_len	Int	Đếm số lượng chữ cái trong hostname.
7	Url_path_length	Int	Đếm độ dài đường dẫn của website sau tên miền.
8	Embedded_brand_name	Int	Phát hiện mức độ liên kết chặt chẽ giữa trang web và thương hiệu bằng cách kiểm tra xem tên thương hiệu có xuất hiện thường xuyên trên trang web hay không
9	www	Int	Thông thường các trang Web giả mạo sẽ chứa từ khóa "www" trong đường dẫn. Kiểm tra số lượng từ khóa "www" để đánh giá website giả mạo hay không
10	Phish_hints	Int	Các trang web giả mạo thường sẽ sử dụng các từ nhạy cảm như ['wp', 'login', 'includes', 'admin', 'content', 'site', 'images', 'js', 'alibaba', 'css', 'myaccount', 'dropbox', 'themes', 'plugins', 'signin', 'view']

Bảng 2: Các đặc trưng URL-based cho việc phát hiện giả mạo

Content-based features

STT	Feature	Type	Mô tả
11	Pct_ext_hyperlinks	Float	Kiểm tra tỷ lệ sử dụng external hyperlinks của website.

12	External_resources	Float	Tỷ lệ sử dụng các nguồn tài nguyên từ bên ngoài. Các trang Web giả mạo có xu hướng sử dụng các nguồn tài nguyên có sẵn cho website
13	Ext_favicon	0/1	Kiểm tra xem favicon của website có lấy từ nguồn bên ngoài hay không.
14	Insecure_form	0/1	Kiểm tra xem thuộc tính form có chứa URL mà không chứa HTTPS protocol hay không
15	Submit_info_to_mail	0/1	Kiểm tra source của website có chứa thuộc tính "mailto:" hay không.
16	Frame_or_iframe	0/1	Kiểm tra thẻ frame hay iframe để hiển thị trang web bổ sung có được sử dụng hay không.
17	Url_prefix_suffix	0/1	Được phân các bởi ký tự '-' trong URL. Những website giả mạo có thể sử dụng tiền tố, hậu tố để đánh lừa người dùng rằng họ đang sử dụng website uy tín.
18	Request_url	Float	Request URL kiểm tra xem các đối tượng có trong trang web như hình ảnh, video và âm thanh có được tải từ một miền khác hay không. Trong các trang web hợp pháp, địa chỉ trang web và hầu hết các đối tượng được nhúng trong trang web chia sẻ cùng một miền
19	Url_of_anchor	Float	Một anchor là một phần tử được định nghĩa bởi thẻ <a>. Nó cũng được xử lý tương tự như Request URL.
20	Links_in_tag	Float	Kiểm tra xem những thẻ <meta>, <script> và <link> được liên kết tới cùng tên miền với webpage hay không.
21	SFH	Float	Các SFH chứa chuỗi trống hoặc "about:blank" hoặc tên miền trong SFH khác với tên miền của trang web

Bảng 3: Các đặc trưng Content-based cho việc phát hiện giả mạo

External-based features

STT	Feature	Type	Mô tả
-----	---------	------	-------

22	Abnormal_url	0/1	Kiểm tra xem tên miền có được đăng ký trên cơ sở dữ liệu WHOIS không
23	Domain_age	Int	Thời gian đăng ký tên miền càng lâu dài thì trang web đó càng đáng tin cậy.
24	Check_dns_record	0/1	Là bản ghi nằm trong DNS servers cung cấp thông tin về cơ sở dữ liệu DNS, cho biết các tên miền, địa chỉ IP gắn với tên miền và cách xử lý các yêu cầu với tên miền đó
25	Page_rank	Int	Các trang web lừa đảo thường không phổ biến do đó có thể dễ dàng nhờ bên thứ 3 đánh giá những website có độ phổ biến hay xếp hạng thấp
26	Google_index	0/1	Các trang web lừa đảo sẽ có thời gian tồn tại ngắn do đó chúng sẽ không có chỉ mục trên google index. Thông qua đó có thể nhờ trang google mà search tìm website giả mạo

Bảng 4: Các đặc trưng External-based cho việc phát hiện giả mạo

3. Huấn luyện mô hình học máy

Trong phần này, các kết quả chi tiết đã được thực hiện với từng mô hình học máy tổng hợp được sẽ được trình bày. Tổng cộng có 2 thử nghiệm được so sánh để tìm ra mô hình học máy phù hợp nhất cho việc tìm ra website giả mạo. Trong kết quả thử nghiệm đầu tiên, ta tìm ra được độ chính xác của từng mô hình học máy làm được dựa trên tập dữ liệu đã nói trên. Sau đó tổng hợp những mô hình học máy có độ chính xác cao nhất, tiến hành qua thử nghiệm thứ hai, vẽ đồ thị học tập của từng mô hình và xem xét dạng của đồ thị học tập. Sau 2 thí nghiệm, ta sẽ tìm ra mô hình phù hợp nhất cho ứng dụng trong việc phát hiện website giả mạo.

3.1 Hiệu năng của từng mô hình học máy

Dựa vào các đặc tính được nêu cụ thể ở phần [16], sử dụng mô hình SVM, Logistic regression, Decision Tree, KNN và Random Forest để đánh giá độ chính xác. Kiểm tra từng ảnh hưởng của từng đặc trưng cũng như so sánh độ chính xác các mô hình đã làm được.

Model	Selected Features	Total data	Macro F1	Accuracy

SVM	30	3170	86%	86%
Logistic Regression	30	3170	82%	83%
Decision Tree	30	3170	82%	83%
KNN	26	3170	85%	86%
Random Forest	26	3170	88%	88%
SVM	26	15998	79%	79%
Logistic Regression	26	15998	89%	89%
Decision Tree	26	15998	94%	94%
KNN	26	15998	91%	91%
Random Forest	26	15998	97%	97%

Mặc dù các số liệu về F1-score và Accuracy của 5 giải thuật nêu trên khá tương đồng, tuy nhiên do dữ liệu được sử dụng từ paper 2015 nên không phù hợp để áp dụng cho các website hiện tại. Vì vậy thông qua việc chọn lọc các đặc trưng quan trọng mà tìm ra dữ liệu mới phù hợp hơn để từ đó so sánh độ chính xác với mô hình từ dữ liệu cũ. Sau khi chọn lọc các đặc trưng quan trọng và nâng tập dữ liệu lên, có thể thấy được rằng Random Forest vượt trội hơn tất cả các mô hình phân loại khác với tỷ lệ chính xác lên đến 97%. Ngoài ra, mô hình SVM cho hiệu suất kém nhất khi thay đổi tập dữ liệu và các đặc trưng quan trọng.

Bên cạnh đó so sánh với các giải thuật mới đã tìm hiểu về Ensemble Learning, so sánh độ chính xác cũng như những thay đổi trong mô hình ảnh hưởng đến độ chính xác như thế nào. Tất cả mô hình ensemble đều sử dụng tập dữ liệu 15998 với 26 đặc trưng, thông qua 5 base model đã được nêu trên: SVM, Logistic regression, Decision Tree, Knn và Random Forest.

Model	Estimator-model	Macro F1	Accuracy
Max Voting		95%	95%
Stacked Generalization	Decision Tree	97%	97%
Stacked Generalization	KNN	97%	97%
Stacked Generalization	SVM	97%	97%
Stacked Generalization	Logistic Regression	97%	97%
Stacked Generalization	Random Forest	97%	97%
Blending	Decision Tree	97%	97%
Blending	KNN	97%	97%

Blending	SVM	97%	97%
Blending	Logistic Regression	96%	97%
Blending	Random Forest	97%	97%
Bagging Classifier	Decision Tree	94%	94%
Bagging Classifier	KNN	91%	91%
Bagging Classifier	SVM	79%	79%
Bagging Classifier	Logistic Regression	91%	91%
Bagging Classifier	Random Forest	97%	97%
Gradient Boosting Classifier		97%	97%
XGBoosting		97%	97%
LightGBM Classifier		98%	98%

Kết quả cho thấy rằng việc áp dụng Ensemble Learning đã cải thiện độ chính xác của mô hình thay vì chỉ áp dụng riêng lẻ một mô hình. Hơn nữa khi áp dụng giả thuật LightGBM cho việc phân loại, kết quả thu được độ chính xác cao nhất là 98%. Mặc dù những mô hình khác có định chính xác tương đối cao 97%, chúng ta cần phải so sánh thêm khả năng học tập từ việc training của các mô hình để chọn ra được mô hình có độ chính xác cao nhất.

3.2 Thời gian xử lý của các mô hình học máy

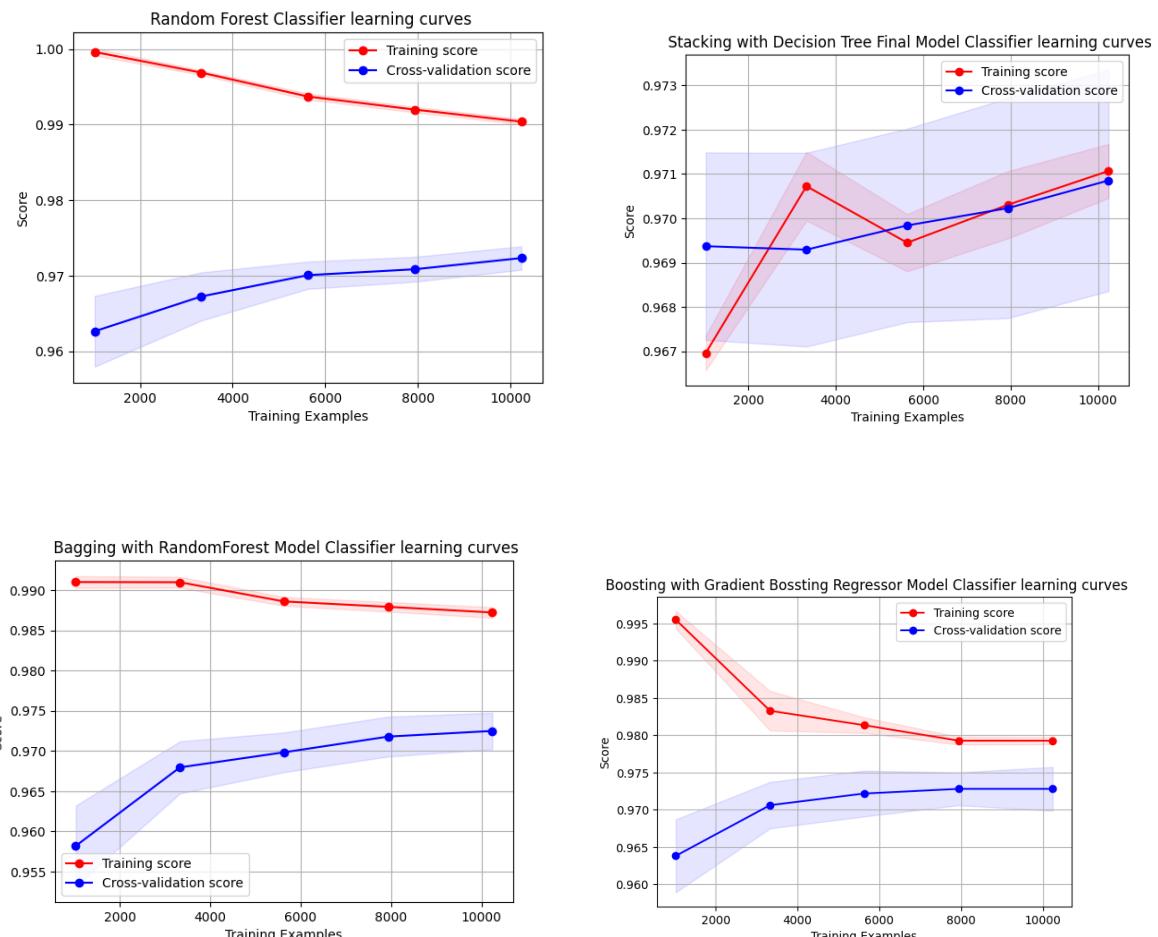
Model	Estimator-model	Accuracy	Time
Random Forest		97%	0.003s
Stacked Generalization	Decision Tree	97%	0.014s
Stacked Generalization	KNN	97%	0.024s
Stacked Generalization	SVM	97%	0.023s
Stacked Generalization	Logistic Regression	97%	0.025s
Stacked Generalization	Random Forest	97%	0.021s
Blending	Decision Tree	97%	0.027s
Blending	KNN	97%	0.025s
Blending	SVM	97%	0.017s
Blending	Logistic Regression	96%	0.016s
Blending	Random Forest	97%	0.016s
Bagging Classifier	Random Forest	97%	0.05s

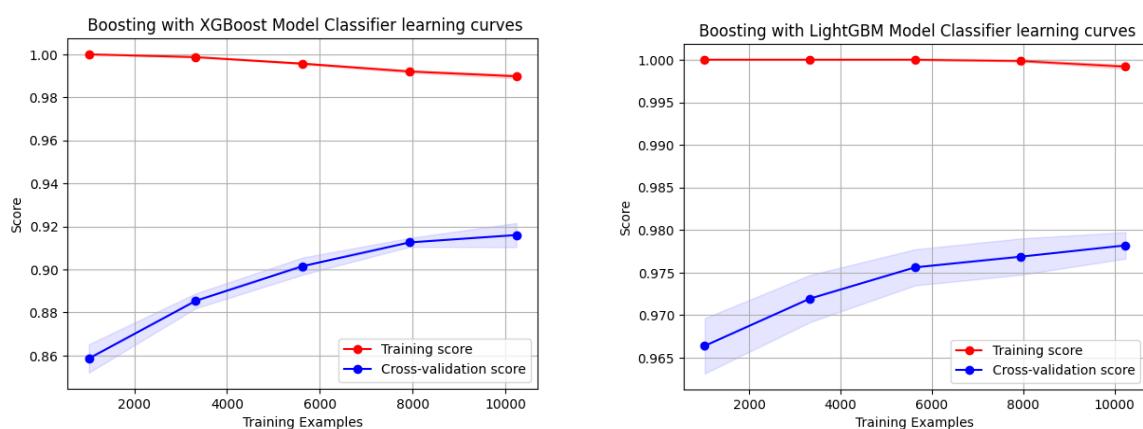
Gradient Boosting Classifier	97%	0.007s
XGBoosting	97%	0.008s
LightGBM Classifier	98%	0.005s

Có thể thấy được thời gian xử lý của các mô hình Ensemble Learning sẽ có tốc độ chậm hơn gấp 4-5 lần so với một mô hình thông thường. Để có thể tiếp kiệm thời gian xử lý của một URL, chọn ra những mô hình có thời gian xử lý thấp hơn 0.015s.

3.3 Đồ thị học tập

Để có thể so sánh những mô hình có độ chính xác tương đương nhau, cần đánh giá thêm khả năng học tập của mô hình thông qua đồ thị học tập đã được nêu ở mục II.7





Hình 26: So sánh đồ thị học tập của từng mô hình

Dựa trên đồ thị, 2 đường *training score* và *cross-validation score* của mô hình Random Forest, Stacking (Decision Tree), Bagging (Random Forest) và Gradient Boosting có xu hướng hội tụ ngang hoặc đã hoàn toàn hội tụ khiến cho độ chính xác của mô hình không còn được tăng cao. Vì vậy khi càng nhiều dữ liệu, mô hình sẽ không cải thiện được nhiều và sẽ lãng phí tài nguyên cho việc training. Trong khi đó các mô hình còn lại có chiều hướng tăng lên khi tập dữ liệu của mô hình tăng lên. Vì vậy khi lượng dữ liệu càng nhiều thì độ chính xác càng cao, phù hợp với việc phát triển ứng dụng sau này.

Vì vậy để chọn mô hình phù hợp phát triển trong việc phát hiện website giả mạo, có 2 mô hình đáp ứng được về độ chính xác, thời gian xử lý và khả năng học hỏi trong tương lai đó là **XGBoost Model** và **LightGBM Model**. Tạm thời nhóm sẽ sử dụng mô hình LightGBM Model để phát triển ứng dụng.

IV. Thiết kế kiến trúc hệ thống phần mềm ứng dụng

1. Giới thiệu tổng quan về phần mềm

Tiện ích mở rộng trình duyệt hay tiện ích mở rộng (browser extension) là một phần mềm nhỏ nhằm bổ sung các tính năng hoặc tùy biến cho trình duyệt web mà không cần dùng đến phần mềm thứ ba. Tiện ích mở rộng là một tệp các file: HTML, CSS, Javascript, các phần thành phần tĩnh (static) như ảnh (image), file txt,... và bất kì các thành phần khác cần sử dụng được nén lại trong một thư mục. Chúng thực chất là những trang web và có thể sử dụng tất cả các APIs mà trình duyệt cung cấp, từ XMLHttpRequest tới JSON, HTML5.

Tiện ích mở rộng phù hợp làm nền tảng để phát triển vì nhiều lý do. Thứ nhất, các tiện ích mở rộng được cài đặt vào trong trình duyệt, nên sẽ không ảnh hưởng nhiều đến tài nguyên của hệ thống máy tính. Thứ hai, người dùng khi truy cập vào một URL bất kì đều thông qua trình duyệt, cho nên những ứng dụng được tích hợp sẵn như tiện ích

mở rộng sẽ là cách đơn giản và hiệu quả nhất để phát hiện trang web lừa đảo. Thứ ba, tiện ích mở rộng có kích thước nhỏ và dễ dàng cài đặt vào trong trình duyệt. Tiện ích mở rộng Google Chrome an toàn và thân thiện với người dùng vì nó chỉ cần một cú nhấp chuột là đã có thể cài đặt. Khi người dùng không muốn dùng nữa, chỉ cần một cú nhấp chuột để tắt tiện ích mở rộng.

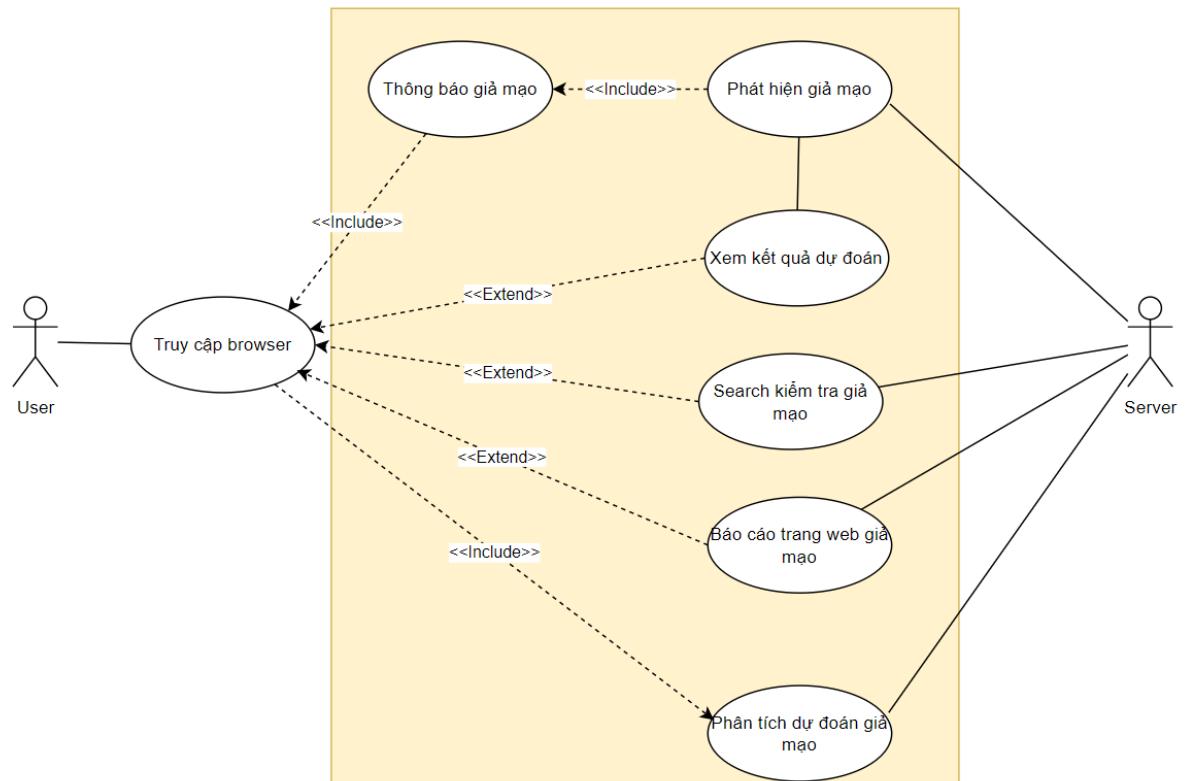
2. Lược đồ mô tả hệ thống

2.1 Lược đồ Use-case

Sơ đồ use-case thể hiện tổng quan cách vận hành của hệ thống. Người dùng có thể cài đặt extension vào browser và truy cập web như bình thường. Hệ thống sẽ tự động trích xuất và xử lý URL. Nếu phát hiện bất kỳ bất thường nào, hệ thống sẽ cảnh báo cho người dùng.

Pre-condition: Người dùng cài đặt extension và truy cập một website.

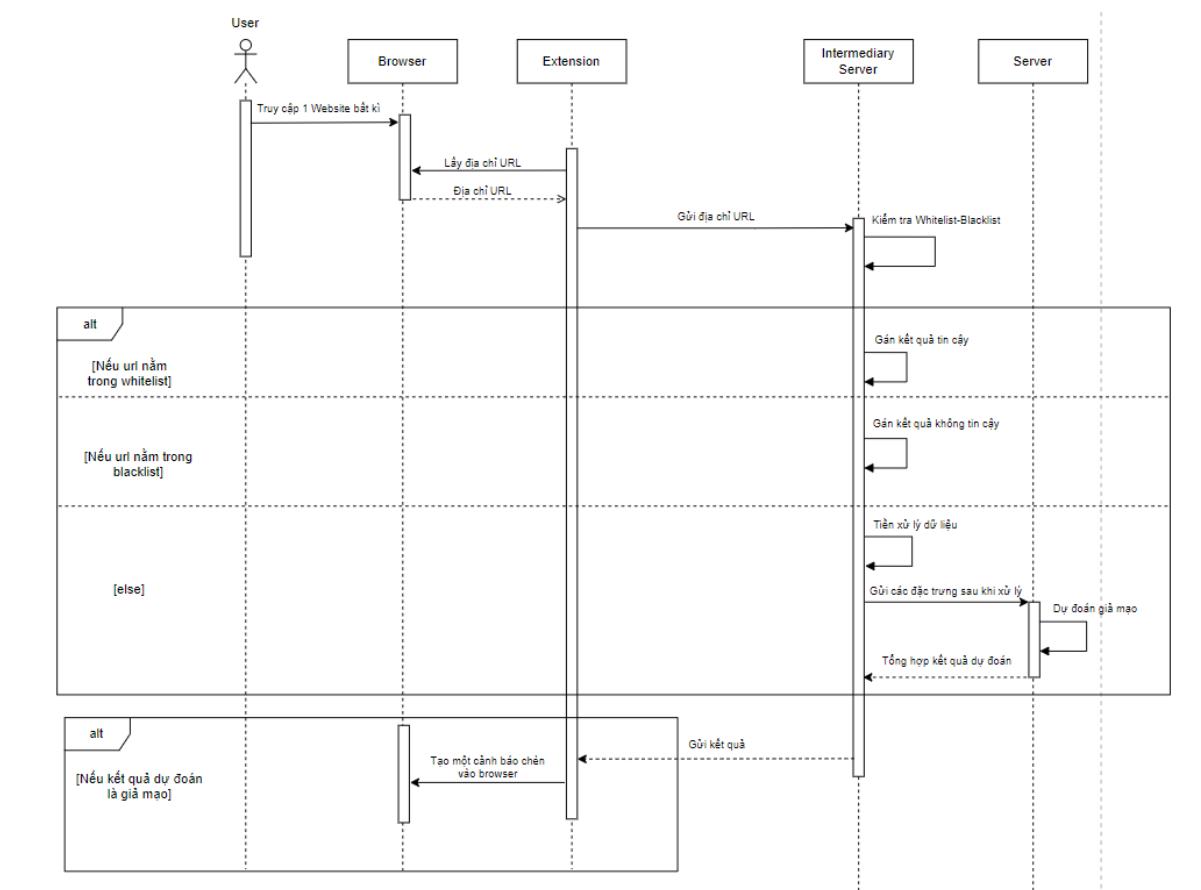
Post-condition: Người dùng được cảnh báo trong trường hợp đó là trang web giả mạo, có thể search kiểm tra một website bất kỳ có phải giả mạo hay không và báo cáo đến server khi biết đến một website giả mạo bất kỳ.



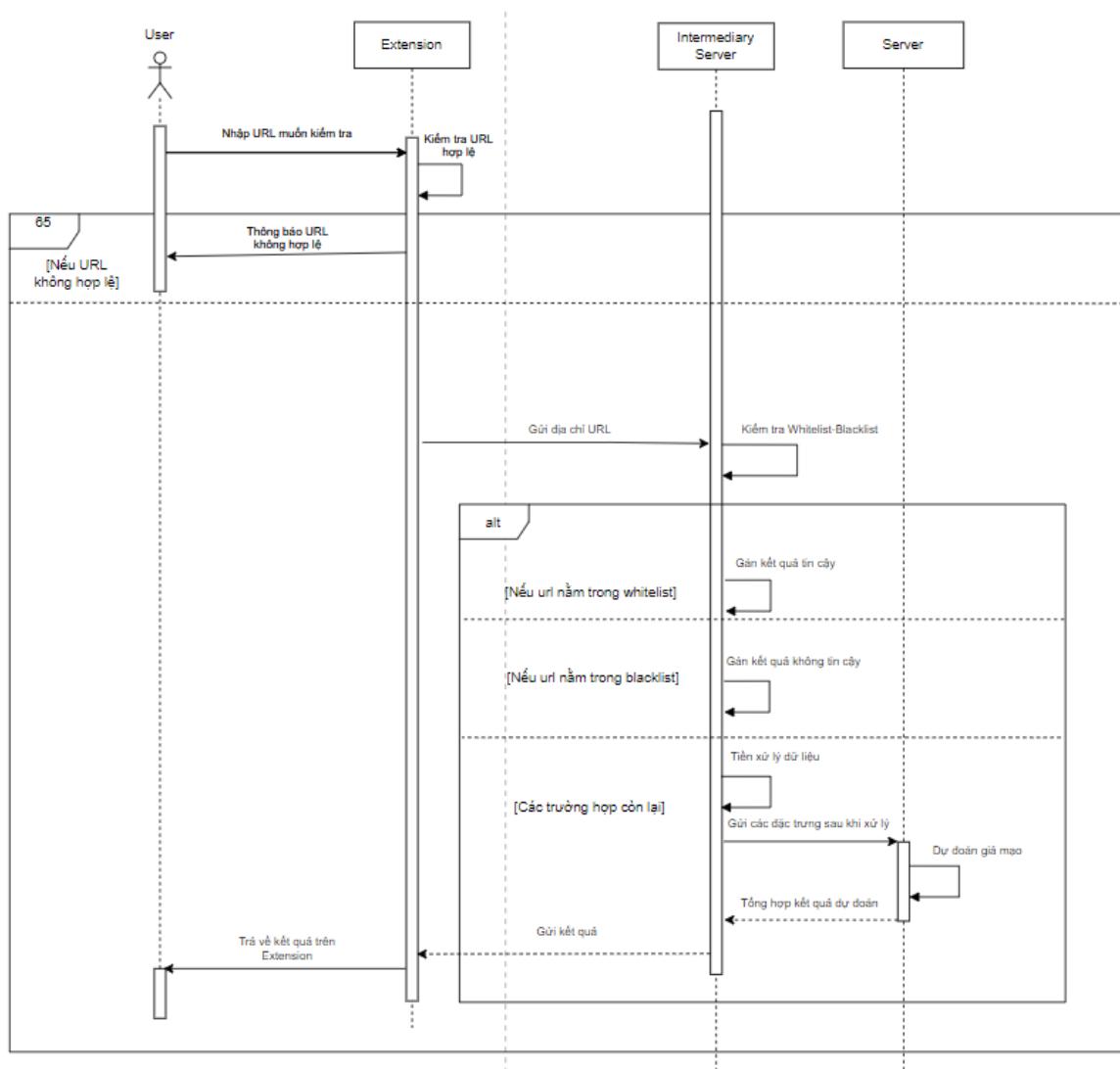
Hình 27: Lược đồ Use-case của hệ thống

2.2 Lược đồ Sequence

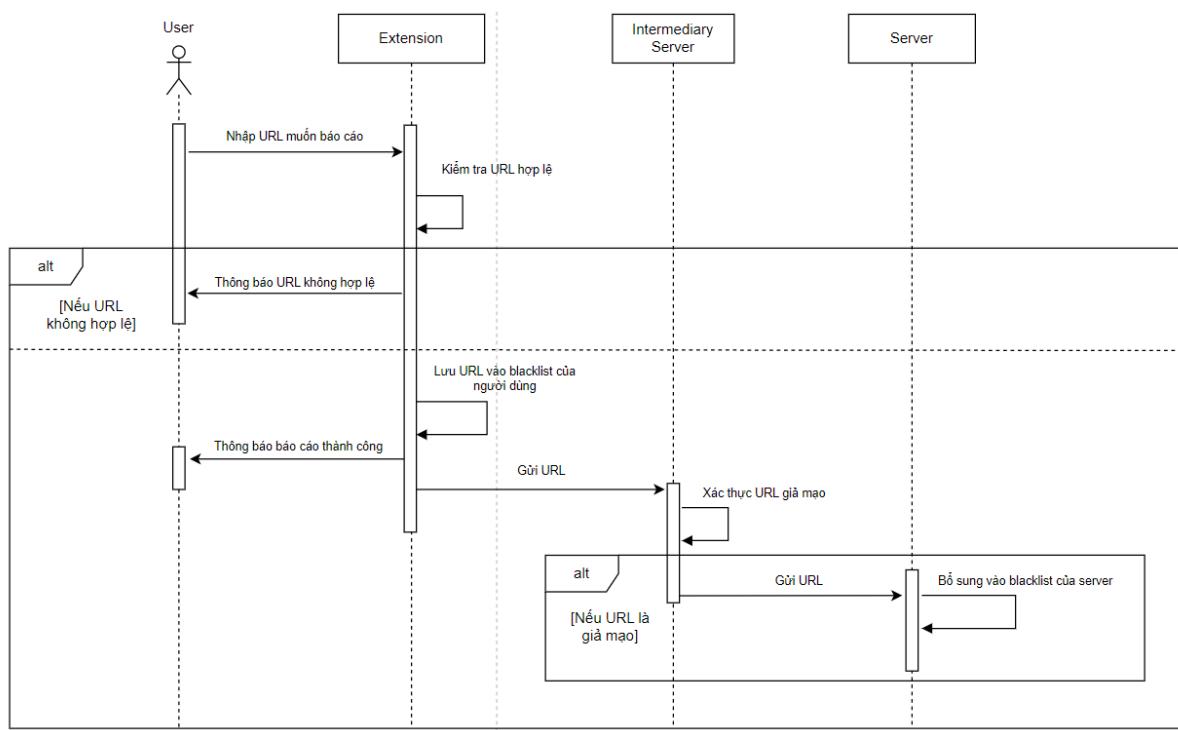
Các lược đồ Sequence diagram mô tả chức năng chính của hệ thống được thể hiện ở hình 28, 29, 30.



Hình 28: Lược đồ Sequence cho usecase cảnh báo người dùng



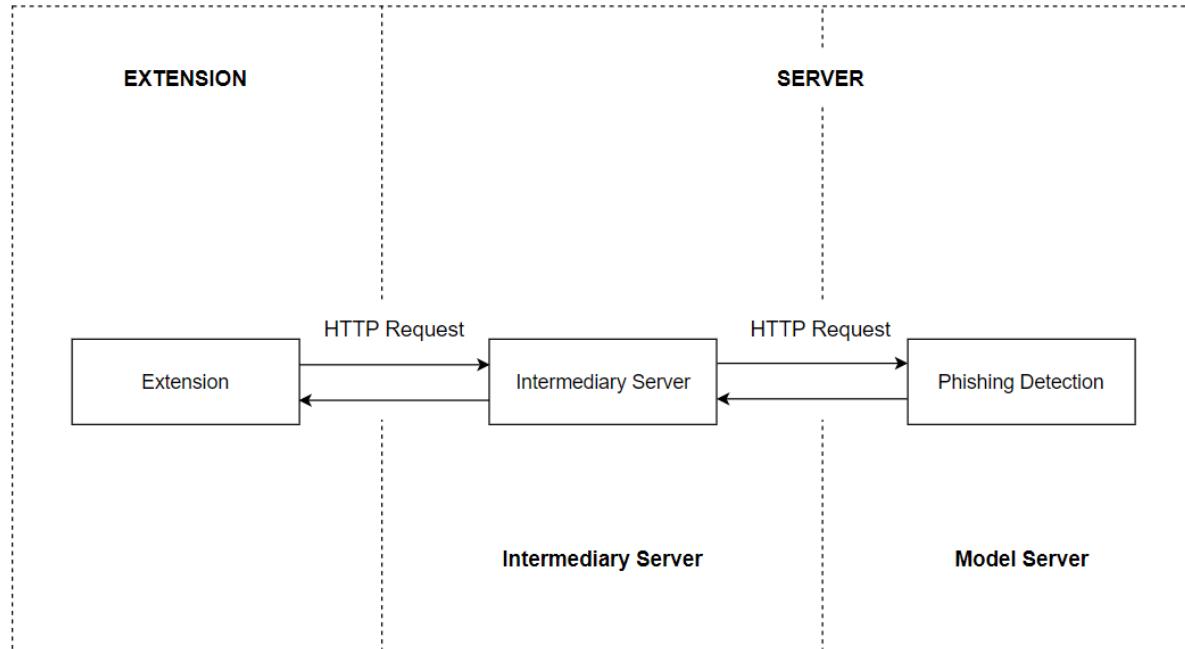
Hình 29: Lược đồ Sequence cho usecase kiểm tra website có phải giả mạo hay không



Hình 30: Lược đồ Sequence cho usecase báo cáo website giả mạo

3. Kiến trúc hệ thống

3.1 Tổng quan về kiến trúc hệ thống



Hình 31: Kiến trúc tổng quan của hệ thống phát hiện lừa đảo

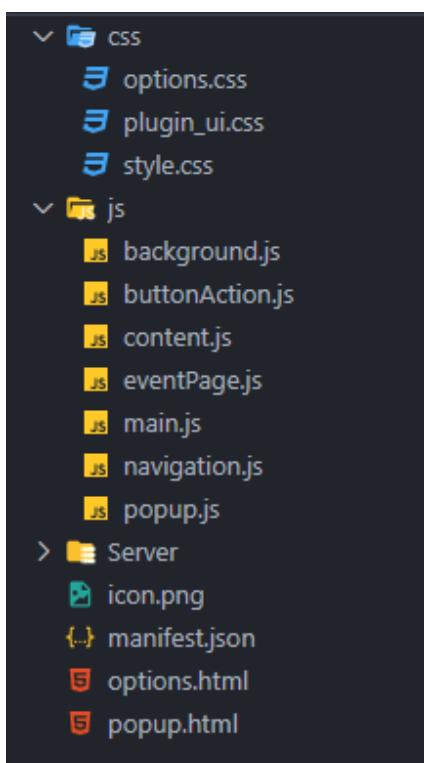
Mô hình kiến trúc tổng thể của hệ thống (hình 31) gồm 2 phần chính: Extension và Server.

- Tiện ích mở rộng (Extension): là thành phần của hệ thống nằm bên phía người dùng. Tiện ích được thực hiện trên trình duyệt Chrome, sau khi cài đặt tiện ích người dùng mỗi khi truy cập vào một địa chỉ URL, tiện ích sẽ gửi các yêu cầu HTTP Request đến phía Server để xử lý, sau khi xử lý Server sẽ trả về kết quả phát hiện được cho người dùng.
- Hệ thống phát hiện giả mạo (Server): là thành phần của hệ thống giúp phân tích URL và dự đoán giả mạo từ chúng. Hệ thống dự đoán sẽ gồm 2 phần nhỏ:
 - + Hệ thống trung gian (Intermediary Server): Nhận yêu cầu từ Extension App và trả kết quả dự đoán về cho người dùng. Hệ thống trung gian có chức năng:
 - * Lấy địa chỉ URL từ trang truy cập của người dùng
 - * Tiền xử lý dữ liệu
 - * Gởi yêu cầu dự đoán cho Model Server
 - * Nhận kết quả dự đoán và trả kết quả về cho người dùng
 - * Lưu trữ các website được báo cáo gửi lên server
 - + Hệ thống dự đoán bằng Machine Learning (Model Server): Nhận yêu cầu (HTTP Request) từ phía Intermediary Server. Model Server có chức năng:
 - * Nhận input từ Intermediary Server
 - * Dự đoán xem địa chỉ URL người dùng vừa truy cập có phải từ một trang web giả mạo hay không
 - * Trả kết quả dự đoán về hệ thống trung gian

Các thành phần của hệ thống giao tiếp với nhau thông qua RESTful API.

3.2 Tiện ích mở rộng (Extension)

Nhóm đã tạo ra một tiện ích mở rộng dựa trên Chrome Extension có khả năng phát hiện các trang web lừa đảo và cảnh báo người dùng khi truy cập vào chúng. Chrome Extension về cơ bản là một trang web được lưu lại trong trình duyệt Chrome của Google, được tạo ra bằng HTML, CSS, JavaScript và có thể tương tác với website thông qua api của trình duyệt. Tạo ra 1 extension cho một số trang nhất định như thay đổi thành phần của trang web đã được load hiện tại hoặc show ra một trang HTML khi được click vào (đồng thời sẽ chạy các câu lệnh JS). Cấu trúc thư mục extension của đê tài:



Hình 32: Cấu trúc thư mục Extension

Trong đó file quan trọng nhất là file **manifest.json**, các thuộc tính trong file dùng để cấu hình cho extension:

- **name:** tên của extension
- **version:** version của extension
- **icons:** những icon muốn hiển thị lên trên thanh trình duyệt
- **action:** những hành động xảy ra tác động lên page khi người dùng click vào icon
- **content_script:** dùng để thêm các file js, css vào các trang web
 - + matches: chỉ định những website sẽ được thêm các file.
 - + scripts: đường dẫn chứa các file js sẽ được thêm vào
 - + css: đường dẫn các file css sẽ được thêm vào
- **permissions:** liệt kê các quyền mà extension của chúng muốn sử dụng, khai báo các url, website, API mà chúng ta muốn chạy
- **background:** quản lý trạng thái của các task

Tiện ích này hoạt động bằng cách nhận diện URL của trang web và gửi yêu cầu đến hệ thống trung gian để phân tích các đặc trưng của trang web đó. Sau khi phân tích xong,

hệ thống sẽ gửi yêu cầu đến máy chủ để đưa ra dự đoán kết quả. Kết quả này sẽ được trả về cho tiện ích, và nó sẽ cập nhật cảnh báo cho người dùng nếu trang web được đánh giá là lừa đảo.

Để giảm thời gian xử lý và giảm tải cho hệ thống, nhóm đã thực hiện tính năng lưu trữ thông tin kết quả phân tích vào bộ nhớ cục bộ của người dùng. Điều này giúp tiện ích có thể truy cập các kết quả đã phân tích trước đó một cách nhanh chóng, mà không cần phải thực hiện lại các phân tích đó mỗi khi trang web được truy cập. Tính năng này cũng giúp giảm tải cho hệ thống, vì nó không cần phải xử lý lại các trang web đã được phân tích trước đó.

3.3 Hệ thống máy chủ (Server)

3.3.1 Hệ thống trung gian (Intermediary Server)

Hệ thống đóng vai trò là thành phần trung gian nhận yêu cầu (request) từ tiện ích mở rộng (Extension) và xử lý dữ liệu trả về. Dựa trên các yêu cầu từ tiện ích, hệ thống sẽ chuẩn hóa dữ liệu đầu vào, gửi các yêu cầu cũng như địa chỉ URL từ đầu vào, thông qua đó gửi lên hệ thống dự đoán bằng học máy (Model Server) để lấy được kết quả dự đoán và trả kết quả về cho người dùng.

Nhóm đã triển khai hệ thống lên **Azure Application Service** dựa trên ngôn ngữ Python. Khi triển khai nhóm đã chọn gói **Basic Service Plan B2**, được thiết kế cho những ứng dụng có lưu lượng truy cập thấp và không cần các tính năng quản lý lưu lượng và quy mô tự động nâng cao, với các thiết lập:

- **Web, mobile, or API apps:** Không giới hạn
- **Disk space:** 10GB
- **Maximum instances:** 3
- **Cores:** 2
- **RAM:** 3.5 GB
- **Pay as you go price:** 0.036 USD/hour (26.28 USD/month)

Sau khi đã triển khai thành công, hệ thống trung gian có domain là: <https://phishingdetector.scm.azurewebsites.net/>

3.3.2 Hệ thống dự đoán bằng học máy (Model Server)

Hệ thống máy chủ đóng vai trò là thành phần trung gian nhận yêu cầu (request) từ tiện ích mở rộng và xử lý dữ liệu trả về. Dựa trên các yêu cầu từ tiện ích, hệ thống sẽ

phân tích đặc trưng dựa trên URL nhận vào, gửi yêu cầu dự đoán đến mô hình học máy, và nhận kết quả trả về từ mô hình học máy. Hệ thống được viết bằng framework Flask dựa trên ngôn ngữ Python. Các HTTP request được sử dụng trong hệ thống:

STT	Method	Endpoint	Chức năng
1	GET	/extractFeatures	Lấy các đặc trưng của một URL
2	GET	/getPrediction	Lấy kết quả dự đoán của URL dựa vào các đặc trưng
3	POST	/postReportUrl	Gửi URL mà người dùng báo cáo lên lưu trữ để sau này xử lý

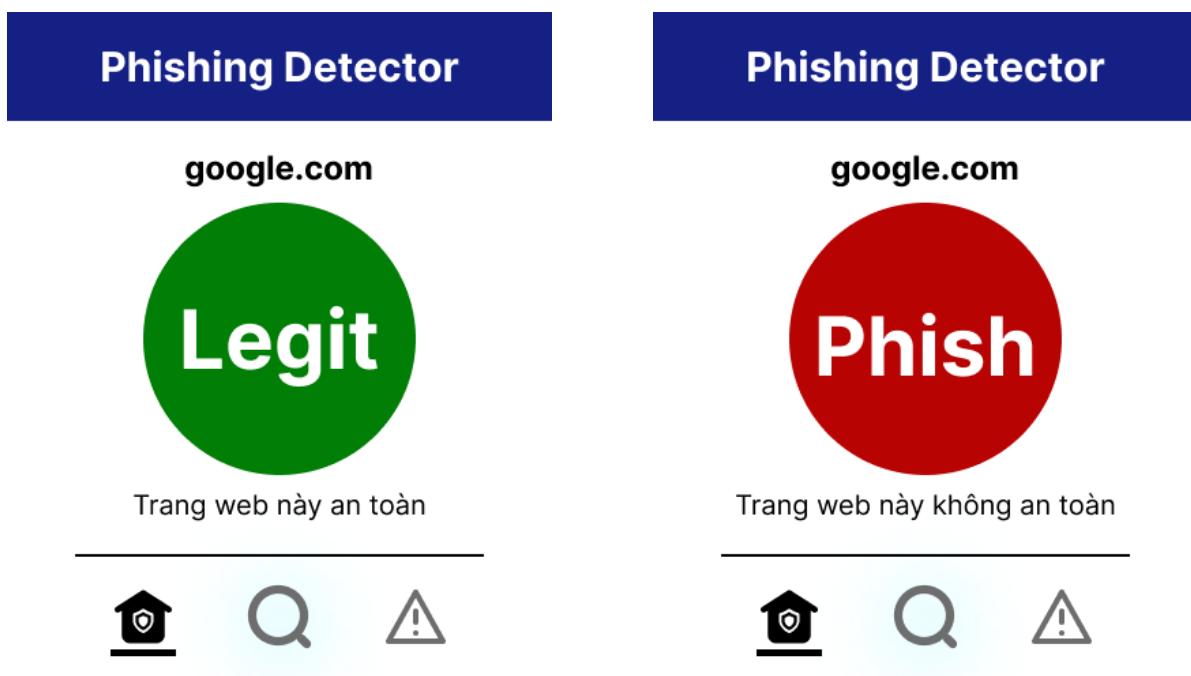
Bảng 8: Các API của hệ thống

Danh sách các tên miền của Whitelist và Blacklist được nhóm lưu vào 2 file text riêng biệt và nó được đọc trước khi đi vào bước phân tích đặc trưng nhằm giảm tải công việc cho máy chủ.

Hiện tại, nhóm đang sử dụng Azure Blob Storage để lưu trữ 1 file text dùng để chứa tất cả các URL đúng cú pháp mà người dùng báo cáo về cho hệ thống ở đường dẫn https://phishingdetectorstorage.blob.core.windows.net/list/report_url.txt nhằm tiện cho việc xem và xử lý. Những đường dẫn đó sẽ được nhóm xem, đánh giá và gán nhãn cho nó theo tần suất nhất định để có thể đóng góp và bổ sung cho tập dataset nhằm cải thiện thêm độ tin cậy của mô hình.

4. Thiết kế UI hệ thống

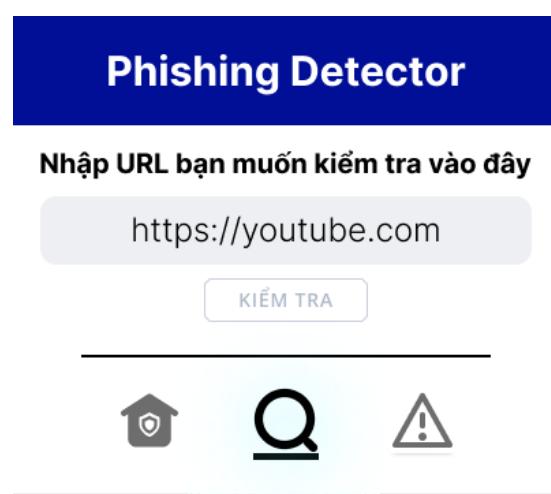
Giao diện cho người dùng được thiết kế đơn giản và dễ sử dụng thông qua HTML và CSS. UI sẽ có một vòng tròn lớn hiển thị trang web đang xem hiện tại có phải là giả mạo hay không. Vòng tròn đó tùy vào trạng thái của trang web mà sẽ hiển thị màu cảnh báo tương ứng đối với điều ra của việc phân loại (xanh lục cho trang web hợp pháp và đỏ cho trang web giả mạo). Bên dưới vòng tròn sẽ hiển thị kết quả phân tích từng đặc trưng của trang web, tùy vào kết quả phân loại đặc trưng sẽ hiển thị màu tương ứng (đỏ - giả mạo, vàng - nghi ngờ, xanh lục - hợp pháp).



Hình 33: Giao diện app hiển thị kết quả

Bên cạnh đó app sẽ còn hiển thị cảnh báo khi người dùng vừa mới truy cập vào một trang web giả mạo để ngăn ngừa người dùng tiếp tục truy cập. Các kết quả kiểm tra chi tiết như độ chính xác hay tốc độ xử lý sẽ hiển thị trên một giao diện riêng.

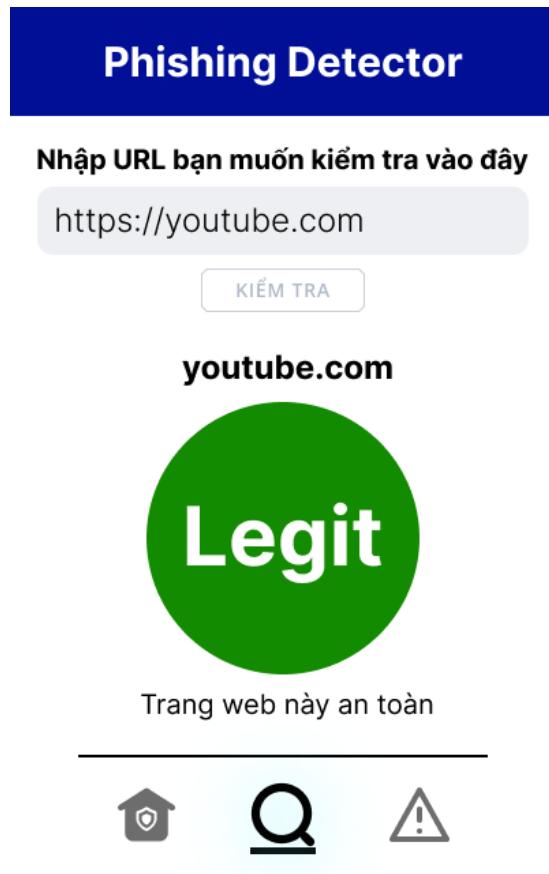
Ngoài ra, người dùng có thể nhập vào extension một đường link bất kỳ để có thể kiểm tra xem trang web đó có uy tín hay lừa đảo không bằng cách bấm vào nút tìm kiếm ở thanh điều hướng.



Hình 34: Giao diện app

Khi bấm nút kiểm tra, ứng dụng sẽ trả về kết quả của trang web đó khi được xử lý

xong.



Hình 35: Giao diện app khi xử lý xong

V. Hiện thực và đánh giá hệ thống

1. Demo ứng dụng

Khi người dùng vừa mở một trang web, tiện ích mở rộng sẽ tiến hành bắt đầu đánh giá độ tin cậy của trang web như hình 36.

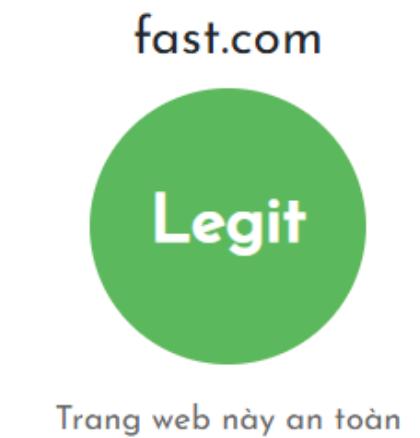
Phishing Detector



Hình 36: Giao diện tiện ích mở rộng khi đang đánh giá

Nếu trang web đó là trang web uy tín, tiện ích sẽ trả về kết quả với khung nền màu xanh nhầm tạo cảm giác tin cậy cho người dùng (hình 37).

Phishing Detector



Hình 37: Giao diện tiện ích mở rộng khi đánh giá là uy tín

Nếu trang web đó có khả năng là trang web lừa đảo, tiện ích sẽ trả về kết quả với khung nền màu đỏ nhầm tạo cảm giác nguy hiểm cho người dùng nếu tiếp tục truy cập vào trang web đó (hình 38).

Phishing Detector

webwave.dev

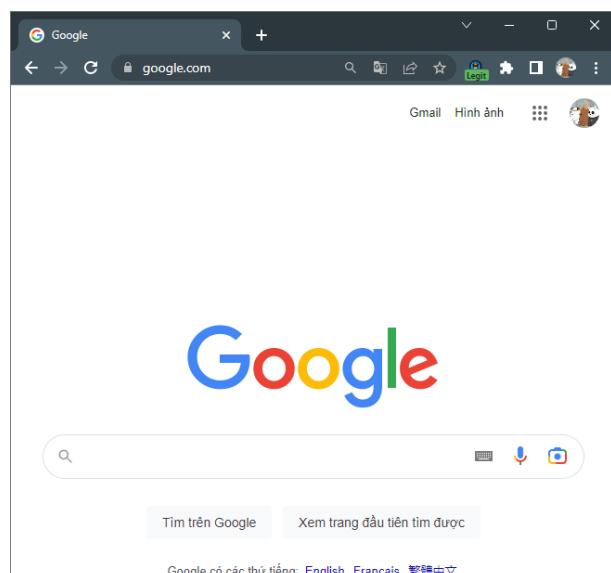


Trang web này không an toàn

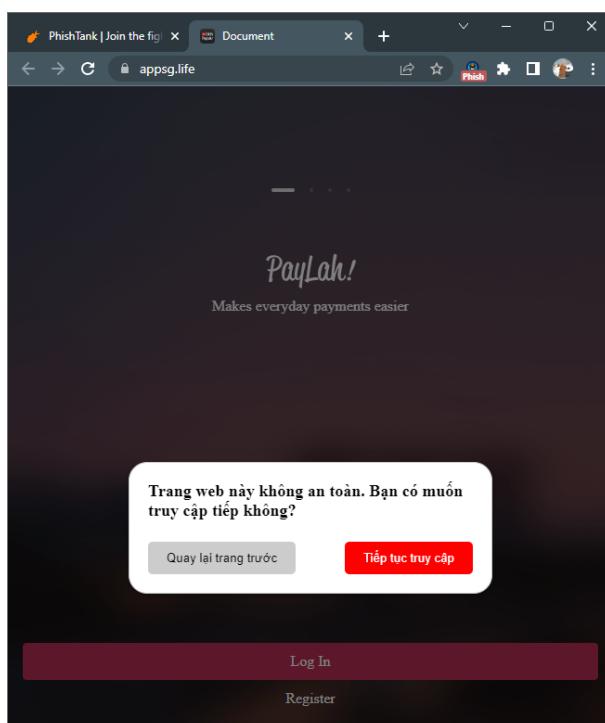


Hình 38: Giao diện tiện ích mở rộng khi đánh giá là lừa đảo

Ngoài ra, khi người dùng truy cập trang web bất kỳ, kết quả dự đoán độ tin cậy của trang web đó sẽ được cập nhật trên biểu tượng của extension nhằm thông báo cho người dùng biết trang web đó là uy tín hay giả mạo mà không cần click vào extension để xem thông tin.

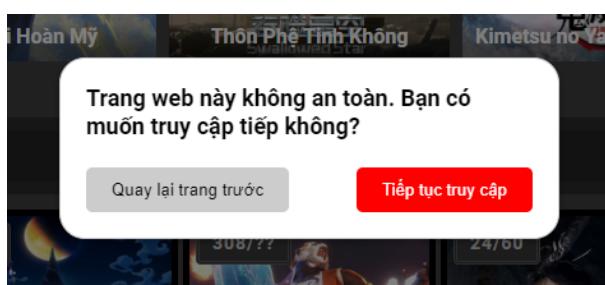


Hình 39: Biểu tượng của extension được cập nhật để hiển thị là uy tín



Hình 40: Biểu tượng của extension được cập nhật để hiển thị là lừa đảo

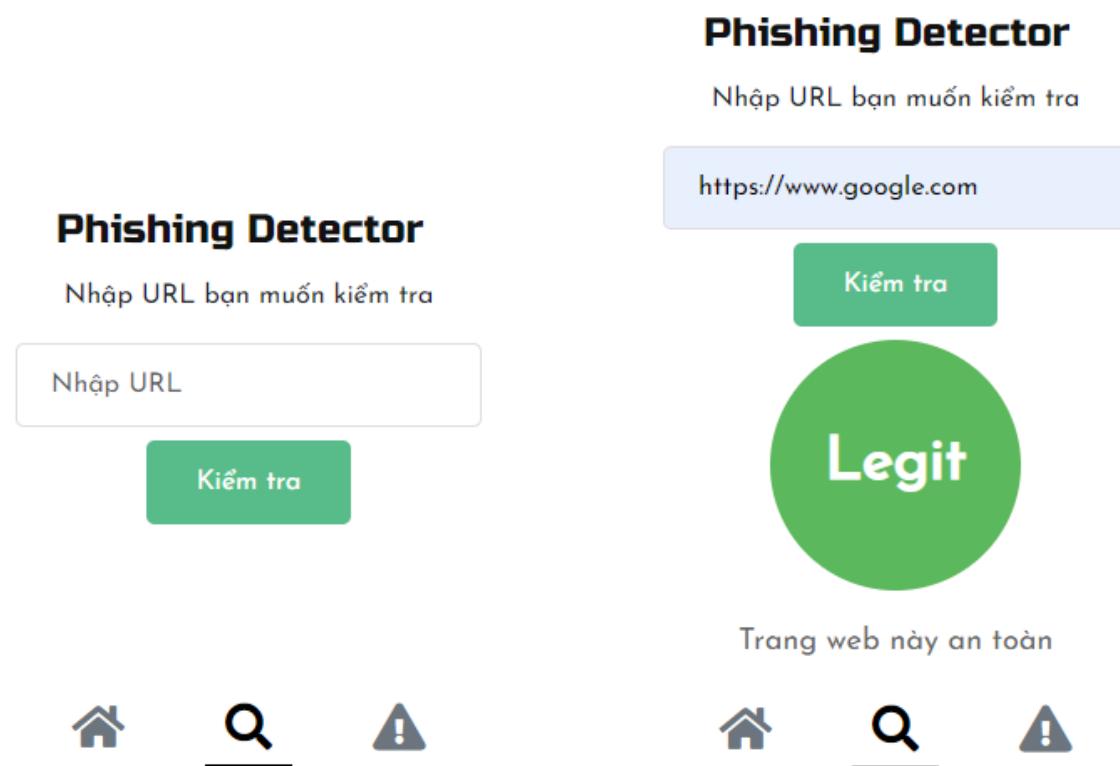
Khi người dùng truy cập vào một trang web không an toàn, extension sẽ hiện lên một pop-up để cảnh báo cho người dùng nếu người dùng không để ý đến biểu tượng.



Hình 41: Pop-up hiện lên để cảnh báo người dùng

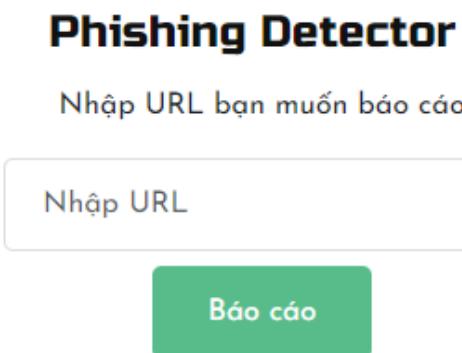
Người dùng chỉ có thể tạm thời tương tác với pop-up với 2 lựa chọn. Nếu người dùng ấn tiếp tục truy cập, thì pop-up sẽ biến mất và người dùng vẫn có thể vào trang web đó một cách bình thường. Và ngược lại, nếu người dùng ấn quay lại trang trước, người dùng sẽ được trả về URL đã truy cập trước đó dựa vào lịch sử duyệt web của mình.

Ngoài ra, người dùng cũng có thể kiểm tra một URL bất kì trước khi muốn truy cập vào trang web đó bằng công cụ kiểm tra ở dưới thanh điều hướng.



Hình 42: Giao diện của extension khi kiểm tra website

Người dùng cũng có thể báo cáo, góp ý trang web mà mình nghi ngờ nó là không an toàn.



Hình 43: Giao diện của extension báo cáo

2. Kiểm thử và đánh giá hệ thống

Để đảm bảo tiện ích chạy không có lỗi và đúng với yêu cầu đặt ra, cần có việc kiểm thử. Do tiện ích không có nhiều tính năng quá phức tạp cần kiểm thử chuyên sâu, nhóm chỉ thực hiện kiểm thử hệ thống. Nhóm cũng tiến hành các đánh giá khác nhau để xác định chất lượng của tiện ích.

2.1 Kiểm thử hệ thống

Kiểm thử hệ thống tập trung vào kiểm thử các chức năng. Việc kiểm thử đảm bảo các chức năng, giao diện và các hành vi của hệ thống thực hiện hoàn chỉnh, đúng yêu cầu.

STT	Yêu cầu	Đánh giá
1	Truy cập một website bất kỳ nằm trong whitelist mà không báo giả mạo	Đạt
2	Hệ thống cảnh báo người dùng khi truy cập một website giả mạo nằm trong white-black list	Đạt
3	Truy cập một website bất kỳ không nằm trong white-black list mà hệ thống vẫn chạy bình thường	Đạt
4	Hệ thống không cảnh báo khi truy cập một website hợp lệ không nằm trong white-black list	Đạt
5	Hệ thống cảnh báo người dùng khi truy cập một website giả mạo không nằm trong white-black list	Đạt
6	Hệ thống không cảnh báo khi truy cập một website giả mạo đã chết	Đạt
7	Tiếp tục truy cập website giả mạo sau khi tắt bảng cảnh báo	Đạt
8	Quay trở lại trang trước thông qua phím tắt trên bảng thông báo	Đạt
9	Hiển thị kết quả giả mạo hoặc hợp lệ lên trên bảng pop up extension	Đạt

Bảng 9: Kiểm thử tính năng cảnh báo người dùng

STT	Yêu cầu	Đánh giá
1	Kiểm tra một website đúng cú pháp vào thanh kiểm tra và trả về kết quả	Đạt
2	Kiểm tra một website không đúng cú pháp vào thanh kiểm tra và không trả về kết quả	Đạt
3	Extension trả về kết quả hợp lệ với các website hợp lệ khi kiểm tra	Đạt
4	Extension trả về kết quả giả mạo với các website giả mạo khi kiểm tra	Đạt

Bảng 10: Kiểm thử tính năng kiểm tra website giả mạo hay không

STT	Yêu cầu	Đánh giá
1	Hệ thống báo lỗi khi điền một thanh địa chỉ không đúng cú pháp	Đạt
2	Hệ thống cập nhật website lên blacklist local của người dùng	Đạt
3	Hệ thống cảnh báo khi truy cập vào website vừa báo cáo	Đạt
4	URL vừa báo cáo được đẩy lên danh sách cần xử lý ở Server	Đạt

Bảng 11: Kiểm thử tính năng báo cáo website giả mạo

2.2 Đánh giá hiệu năng hệ thống

Nhóm tiến hành đánh giá hiệu năng toàn bộ hệ thống dựa vào các tính năng đã nêu. Hai tính năng trên ảnh hưởng trực tiếp đến trải nghiệm người dùng, còn việc báo cáo website giả mạo việc cập nhật blacklist còn phụ thuộc vào thời gian cập nhật của server nên không thể đánh giá chính xác hoàn toàn.

STT	Trường hợp phản ứng	Thời gian phản hồi
1	Người dùng truy cập một website nằm trong white-black list	0.32s
2	Người dùng truy cập một website giả mạo nằm ngoài white-black list	4.12s
3	Người dùng truy cập một website hợp lệ nằm ngoài white-black list	5.84s
4	Kiểm tra một website không đúng cú pháp	0.14s
5	Kiểm tra một website đúng cú pháp nằm trong white-black list	0.24s
6	Kiểm tra một website giả mạo không nằm trong white-black list	7.55s
7	Kiểm tra một website hợp lệ không nằm trong white-black list	9.03s
8	Báo cáo một website bất kỳ không đúng cú pháp	0.56s
9	Báo cáo một website bất kỳ đúng cú pháp	0.37s

Bảng 12: Kiểm tra tốc độ xử lý của hệ thống

Tiến hành đánh giá thời gian phản hồi của hệ thống khi nhiều người dùng sử dụng cùng một lúc mà những URL không thuộc whitelist và blacklist thông qua chức năng cảnh báo người dùng.

Số lượng URL cần phân tích cùng lúc	Thời gian phản hồi lâu nhất
1	5.34s
2	5.69s
3	10.43s
4	15.39s
5	18.73s
6	21.34s
7	25.62s
8	29.83s

Bởi vì số lượng vCPU của Azure App Services khá ít (chỉ 2 vCPU với gói Basic Service Plan B2), cho nên ứng dụng chỉ hoạt động tốt với số lượng URL cần phân tích cùng lúc khoảng 3-4 URL (không ảnh hưởng đến các URL nằm trong whitelist hay blacklist). Vì vậy, nhóm quyết định lưu những URL kèm kết quả đã phân tích mà tất cả các người dùng đã truy cập, nhằm tăng khả năng chịu tải cho phía Server, và giúp người dùng sẽ nhận được kết quả nhanh hơn.

2.3 Đánh giá và nhận xét từ người dùng tiện ích

Nhóm đã tiến hành triển khai đưa cho một số bạn bè sử dụng và đánh giá thông qua Google Form



Đánh giá và nhận xét tiện ích PhishingDetector

Hiện nay việc tấn công lừa đảo qua mạng là một hình thức tấn công phổ biến bằng việc xây dựng những hệ thống lừa đảo nhằm việc đánh cắp các thông tin nhạy cảm như tên đăng nhập, mật khẩu, hay thông tin về các loại thẻ tín dụng của người dùng.

Chính vì vậy nhằm nâng cao cảnh giác, nhóm đã xây dựng hệ thống ứng dụng giúp phát hiện những trang web giả mạo giúp cảnh báo người dùng khi truy cập những website có thông tin đáng ngờ.

Để có cái nhìn khách quan về người dùng, bạn hãy sử dụng và đánh giá tiện ích của nhóm từ đó giúp nhóm khắc phục những hạn chế tồn đọng và cải thiện trong tương lai.

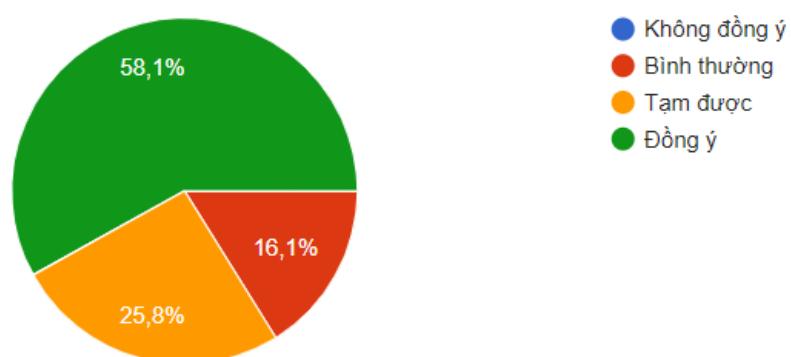
Link hướng dẫn tải và cài đặt Phishing Detector App: <https://github.com/zelvor/PhishingDetectorExtension>

Hình 44: Form đánh giá và nhận xét

Nhóm đã thu nhập được 31 lượt phản hồi từ những người tham gia đánh giá form. Các tiêu chí đánh giá bao gồm giao diện trực quan, dễ nhìn, dễ sử dụng, thời gian phản hồi của ứng dụng, chức năng cảnh báo người dùng, chức năng kiểm tra website giả mạo, chức năng cảnh báo và đánh giá tính hữu dụng của ứng dụng.

Giao diện trực quan, dễ nhìn, dễ sử dụng

31 câu trả lời

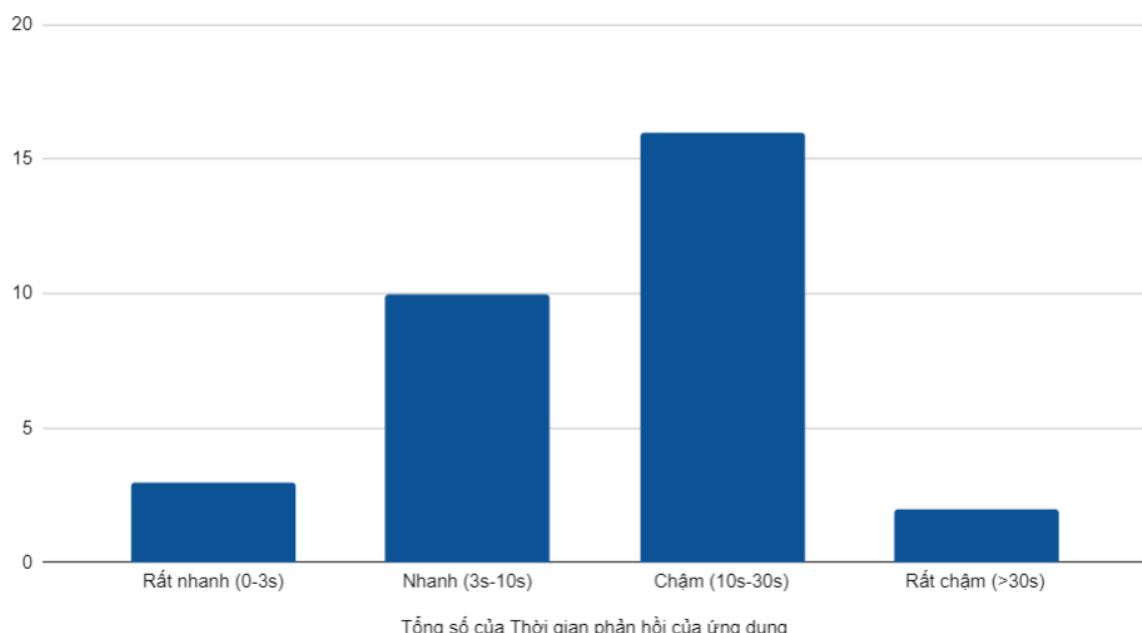


Hình 45: Kết quả đánh giá về UI của ứng dụng

Thông qua bảng đánh giá tính trực quan, dễ nhìn, dễ sử dụng, biểu đồ thống kê đa số nhận phản hồi tốt về tính trực quan của ứng dụng. Tuy không có các phản hồi "không đồng ý" về giao diện của hệ thống nhưng đến gần phân nửa phản hồi vẫn cho rằng giao diện của hệ thống là "bình thường" và "tạm được". Có thể thấy rằng ứng dụng đã đáp ứng được trải nghiệm cơ bản nhưng vẫn cần cải thiện nhiều về cách bố trí và phối màu để nâng cao tính trải nghiệm của người dùng hơn nữa.

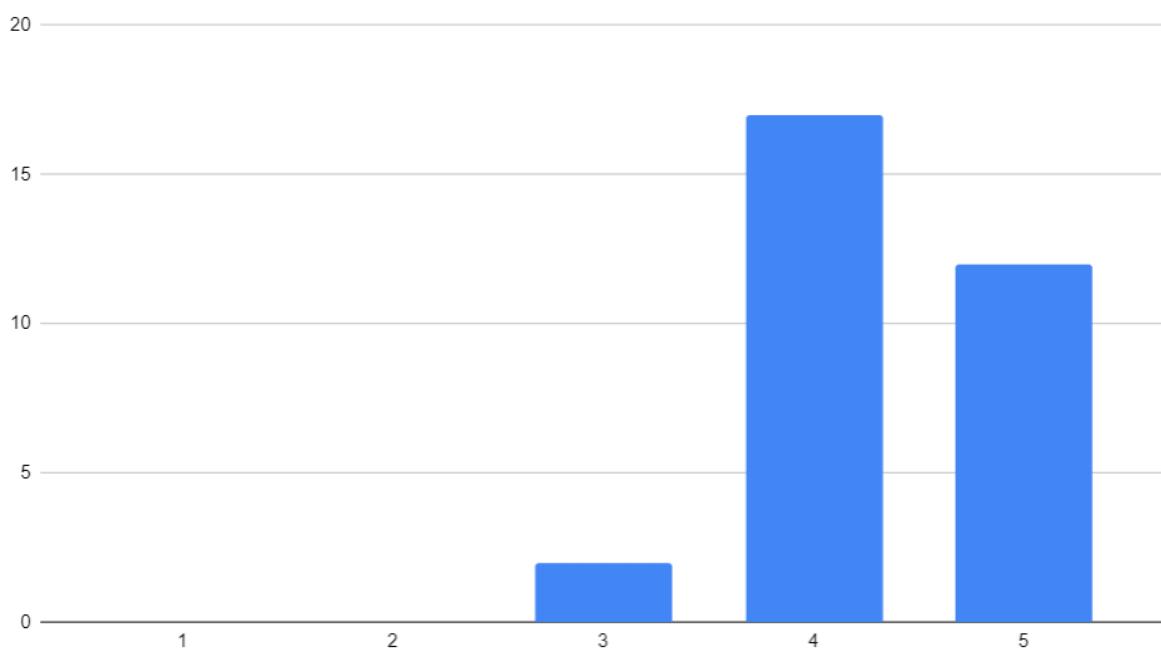
Tuy nhiên về thời gian phản hồi của ứng dụng, trong tổng số 31 phản hồi thì chưa có phản hồi nào gặp trục trặc hay lỗi trong khi sử dụng các tính năng của extension. Có thể thấy được trung bình thời gian mà ứng dụng phản hồi rơi từ tầm 3s-30s và việc phản hồi rất chậm (>30s) chỉ chiếm 6.5% mặc cho lượng người dùng luôn dao động, dù vậy nhóm thấy người dùng vẫn khá hài lòng với tốc độ này.

Tổng số của thời gian phản hồi của ứng dụng



Hình 46: Kết quả đánh giá của người dùng về thời gian phản hồi

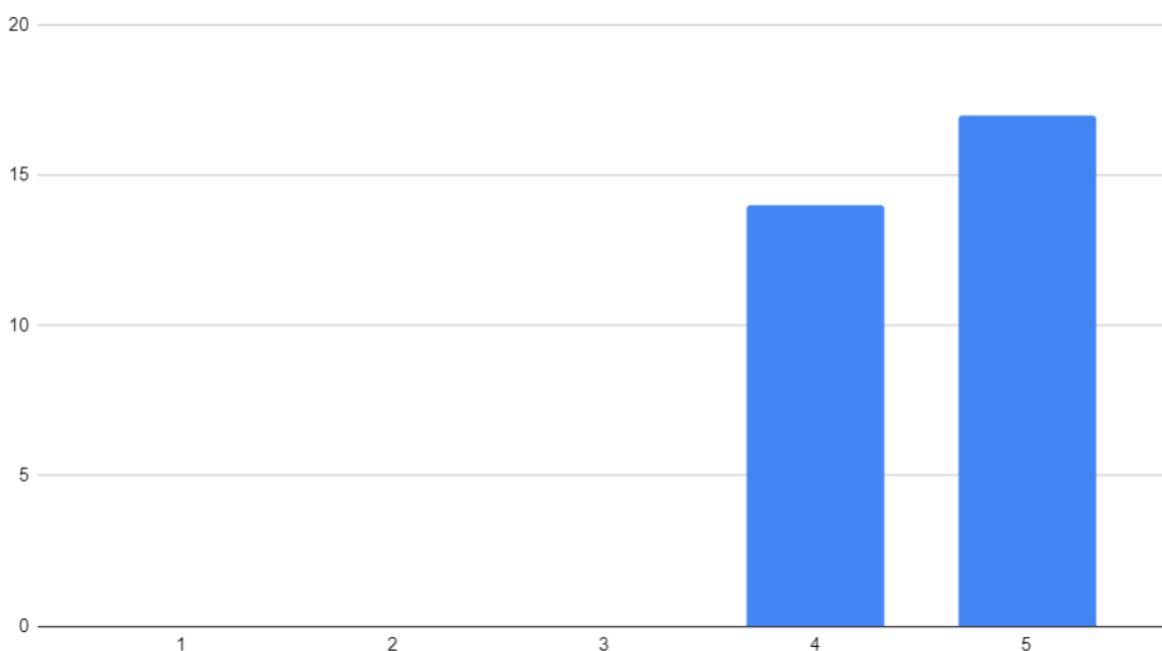
Chức năng cảnh báo người dùng



Hình 47: Kết quả đánh giá tính năng cảnh báo người dùng

Thông qua bảng đánh giá đến tính năng cảnh báo người dùng, đa số trải nghiệm của người sử dụng vẫn tương đối tốt, hệ thống vẫn đáp ứng được khi lượng người sử dụng không quá đông. Vẫn còn một số chậm chạp trong việc phản hồi tuy nhiên đa số vẫn đánh giá tính năng này từ 4-5 điểm đạt tới 93.5%.

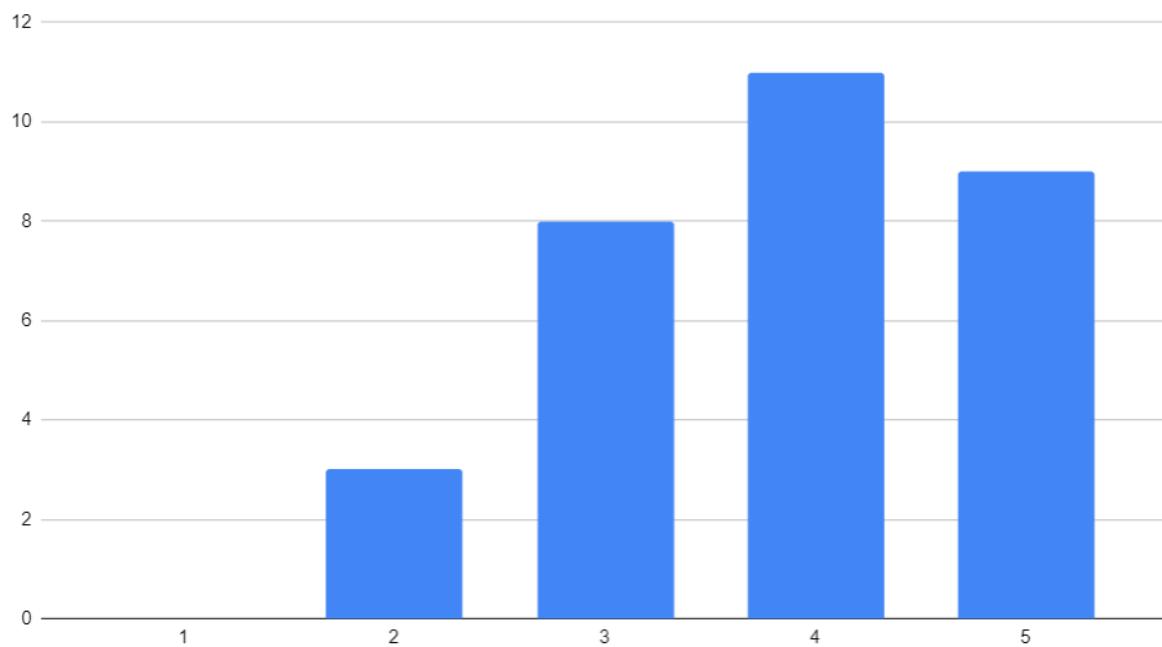
Chức năng kiểm tra website có phải giả mạo hay không



Hình 48: Kết quả đánh giá tính năng kiểm tra trang web giả mạo

Thông qua bảng đánh giá tổng quan về tính năng kiểm tra trang website giả mạo, nhóm nhận thấy người dùng khá hài lòng về chức năng này và có phần nhỉnh hơn so với các tính năng còn lại. Hầu như mọi phản hồi từ tính năng này đều là 4 hoặc 5 điểm, không có bất cứ phản hồi tiêu cực nào về tính năng này

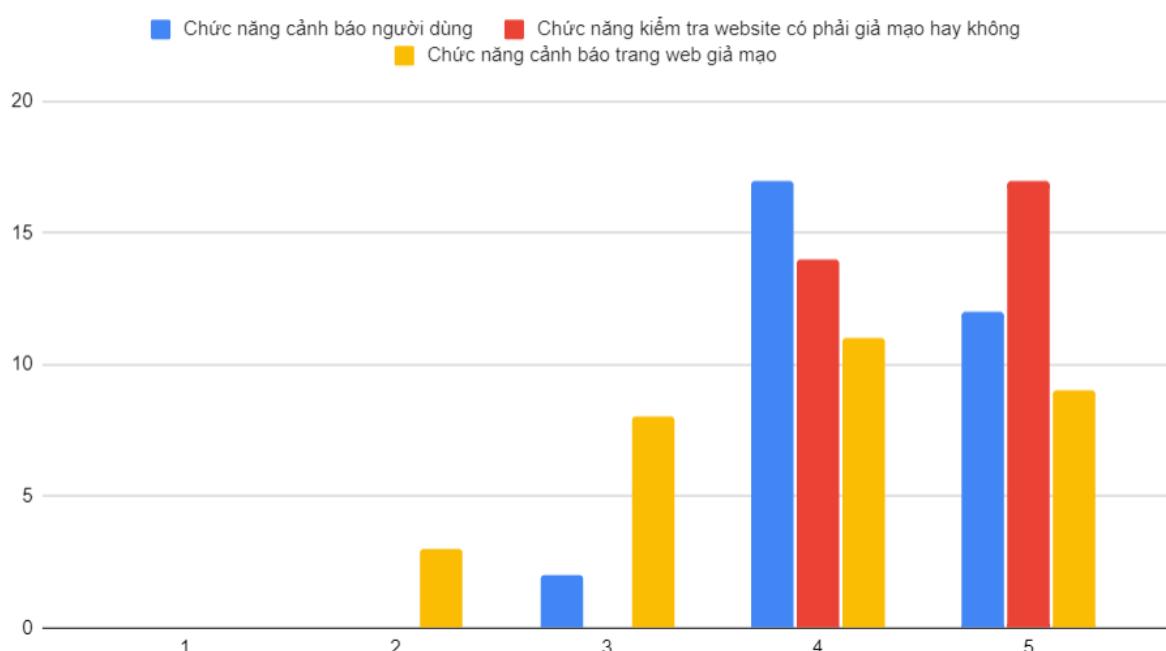
Chức năng cảnh báo trang web giả mạo



Hình 49: Kết quả đánh giá tính năng báo cáo website giả mạo

Đối với chức năng cảnh báo người dùng, nhóm đã nhận nhiều phản ứng trái chiều khi nhiều người đánh giá rất thấp tính năng này. Mặc dù về điểm số, tính năng này vẫn được chấm nhiều 4-5 điểm nhưng so với các tính năng khác thì 2-3 điểm chiếm số lượng nhiều nhất. Đây cũng là tính năng nhóm còn nhiều thiếu sót khi vẫn chưa có chức năng tự cập nhật những URL được báo cáo vào blacklist mà vẫn đang cập nhật bằng thủ công vì những URL mới cập nhật có thể có những đặc tính khác so với những URL lừa đảo cũ. Việc xác định URL đó có lừa đảo hay giả mạo hay không phải phụ thuộc những người có chuyên môn gán nhãn.

So sánh đánh giá các chức năng

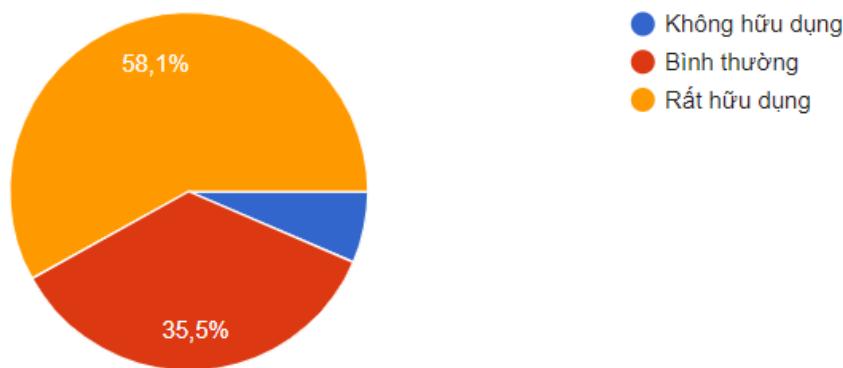


Hình 50: So sánh các tính năng với nhau

Dựa vào các bảng so sánh số phản hồi liên quan đến chức năng cảnh báo người dùng và kiểm tra trang web giả mạo mang lại nhiều đánh giá tích cực nhất, từ hình 50 thông qua 31 người đánh giá không ai đánh giá 1 điểm từ đó có thể thấy được nhiều người dùng đều hài lòng cũng như có trải nghiệm tốt với ứng dụng. Bên cạnh đó, nhóm cũng nhận phản hồi từ độ hữu dụng của tiện ích.

Tính hữu dụng của ứng dụng

31 câu trả lời



Hình 51: Kết quả người dùng đánh giá tính hữu ích của ứng dụng

Về mặt tính hữu dụng của tiện ích mở rộng 51, ứng dụng nhận được nhiều phản hồi tốt lên đến 58.1%. Chỉ có khoảng 6.5% người dùng cảm thấy ứng dụng chưa thật sự hữu dụng, điều này dễ hiểu vì nhiều người dùng chưa bị lừa đảo qua mạng nên chưa nhận thức được tầm ảnh hưởng của nó. Vì vậy nhóm cần phải cải thiện ứng dụng này nhiều hơn và đưa đến cho nhiều người dùng hơn nữa để nâng cao nhận thức cũng như lan truyền sự nguy hiểm của việc giả mạo trên các trang web.

VI. Tổng kết

1. Kết quả đạt được

Qua quá trình thực hiện đề cương và luận văn tốt nghiệp, nhóm đã đạt được các kết quả sau:

- Tổng hợp nội dung và cung cấp kiến thức về thực trạng của việc giả mạo trên mạng hiện nay
- Tổng hợp các đặc trưng trích xuất được từ URL và so sánh tìm ra các đặc trưng quan trọng hỗ trợ cho việc phân loại
- Tổ chức tổng hợp và đánh nhãn giả mạo từ nhiều nguồn website khác nhau.
- Trích xuất được những đặc trưng của từng URL trong tập dữ liệu cho việc huấn luyện và phân loại mô hình
- Xây dựng hệ thống hướng dẫn phân loại website giả mạo và tổng hợp các loại đo lường giúp so sánh các mô hình dễ dàng hơn.

- Xây dựng mô hình giúp phân loại các trang web giả mạo.
- Kết hợp các mô hình lại sử dụng Ensemble Learning.
- So sánh và đánh giá các mô hình phân loại để tìm ra mô hình phù hợp nhất.
- Xây dựng tiện ích mở rộng trên Chrome giúp cảnh báo người dùng khi truy cập website giả mạo, kiểm tra một website bất kỳ và báo cáo website đó lên Server.
- Triển khai mô hình học máy lên môi trường sử dụng, kết hợp với tiện ích mở rộng và hệ thống trung gian tạo thành một hệ thống hoàn chỉnh.

2. Những hạn chế tồn tại

Hệ thống của nhóm khắc phục cũng như cải thiện so với Đồ án chuyên ngành, song vẫn còn tồn tại một số hạn chế cần khắc phục trong tương lai như:

- Thời gian xử lý cho việc trích xuất các đặc trưng chưa được tối ưu nên còn tốn khá nhiều thời gian.
- Do còn phụ thuộc vào chi phí chi trả cho Azure App Service nên chưa tối ưu được thời gian phản hồi cho nhiều người dùng một lúc.
- Hệ thống chưa triển khai được trên Docker cũng như kiến thức về triển khai trên Docker còn hạn hẹp nên còn phụ thuộc nhiều vào các Service bên ngoài.
- Tiện ích mở rộng mới chỉ hỗ trợ được trên các trình duyệt có nhân Chromium, do đó một bộ phận người dùng sẽ không tiếp cận được với công cụ này
- Do riêng tạo riêng Database để lưu trữ dữ liệu nên khi truy xuất các data còn khá nhiều nhọc nhằn.
- Vẫn chưa mang ứng dụng lên cửa hàng tiện ích, nên chưa nhận được ý kiến khách quan từ người dùng.

3. Hướng phát triển trong tương lai

Trong tương lai, nhóm mong muốn cải thiện một số vấn đề và phát triển hệ thống như:

- Mở rộng thêm nhiều lựa chọn, giúp người dùng thoải mái tùy chỉnh extension.
- Cải thiện tốc độ xử lý và khả năng chịu tải của hệ thống.



- Bổ sung tập dữ liệu mới nhất từ các trang web cung cấp website giả mạo để gia tăng độ hiệu quả khi phát hiện lừa đảo.
- Việc triển khai các mô hình tham khảo thêm nhiều chuyên gia có kinh nghiệm trong lĩnh vực an ninh mạng để đạt độ chính xác và tính ứng dụng thực tiễn cao.
- Mang ứng dụng lên cửa hàng tiện ích mở rộng Chrome để triển khai thu thập đánh giá của nhiều người dùng. Từ đó bổ sung các tính năng cho hệ thống cũng như mở rộng quy mô ứng dụng.
- Mở rộng tiện ích trên nhiều nền tảng khác nhau để đa dạng hóa và thu hút thêm người sử dụng.

TÀI LIỆU THAM KHẢO

- [1] Abdul Basit, Maham Zafar, Xuan Liu, Abdul Rehman Javed, Zunera Jalil, Kashif Kifayat, "A comprehensive survey of AI-enabled phishing attacks detection techniques", 2020.
- [2] Anti-Phishing Working Group (APWG), "Phishing Activity Trends Report, 1st Quarter 2022", retrieve from: https://docs.apwg.org/reports/apwg_trends_report_q1_2022.pdf, accessed 10.10.2022.
- [3] Kang Leng Chiewa , Choon Lin Tan, KokSheik Wong , Kelvin S.C. Yong, Wei King Tiong, "A new hybrid ensemble feature selection framework for machine learning-based phishing detection system", 2019.
- [4] Lizhen Tang , Qusay H. Mahmoud, "A Survey of Machine Learning-Based Solutions for Phishing Website Detection", 2021.
- [5] Vishakha Prashant Ratnaparkhi, Sahil Siddharth Jambhulkar, "FRAMEWORK FOR DETECTION AND PREVENTION OF PHISHING WEBSITE USING MACHINE LEARNING APPROACH", 2020.
- [6] Shafaizal Shabuddin, Nor S Sani, Mohd Aliff, "Feature Selection for Phishing Website Classification", 2020.
- [7] Ammara Zamir, Hikmat Ullah Khan and Tassawar Iqbal, Nazish Yousaf, Farah Aslam, Almas Anjum, Maryam Hamdani, "Phishing web site detection using diverse machine learning algorithms", 2020.
- [8] V. H. Tiệp, "Machine learning cơ bản," 2019.
- [9] Pham Minh Hoang , "Ensemble learning và các biến thể (P1)", Retrieve from: <https://viblo.asia/p/ensemble-learning-va-cac-bien-the-p1-WAyK80AkKxX>, 2020
- [10] Ting, K.M. & Witten, I.H, "Stacked generalization: when does it work?", 1997.
- [11] Joseph Rocca, "Ensemble methods: bagging, boosting and stacking", retrieve from: <https://towardsdatascience.com/ensemble-methods-bagging-boosting-and-stacking-c9214a10a205>, 2019.

- [12] Sai Nikhilesh Kasturi, "XGBOOST vs LightGBM: Which algorithm wins the race !!!", retrieve from: <https://towardsdatascience.com/ensemble-methods-bagging-boosting-and-stacking-c9214a10a205>, 2019.
- [13] Jian Mao, Jingdong Bian, Wenqian Tian, Shishi Zhu, Tao Wei, Aili Li, Zhenkai Liang, "Phishing page detection via learning classifiers from page layout feature", 2019.
- [14] Ariyadasa, Subhash; Fernando, Shantha; Fernando, Subha, "Phishing Websites Dataset", Mendeley Data, V1, doi: 10.17632/n96ncsr5g4.1, 2021.
- [15] Abdelhakim Hannousse, Salima Yahiouche, "TOWARDS BENCHMARK DATASETS FOR MACHINE LEARNING BASED WEBSITE PHISHING DETECTION: AN EXPERIMENTAL STUDY", 2020.
- [16] Rami M. Mohammad, Fadi Thabtah, Lee McCluskey, "Phishing Websites Features", 2015.