BAN CƠ YẾU CHÍNH PHỦ

PHÂN HIỆU HỌC VIỆN KTMM TẠI TP. HỒ CHÍ MINH

ĐỒ ÁN TỐT NGHIỆP

PHÁT TRIỂN ỨNG DỤNG PHÁT HIỆN TIN NHẮN LỪA ĐẢO BẰNG PHƯƠNG PHÁP HỌC MÁY

Ngành: An toàn thông tin

Mã số: 7.48.02.02

Sinh viên thực hiện:

Đàm Chí Nguyên

Lớp: AT15H

Giảng viên hướng dẫn:

Nguyễn Xuân Sâm

Giảng viên trường khoa CNTT, HCMUTE

Thành phố Hồ Chí Minh, 2023

BAN CƠ YẾU CHÍNH PHỦ

PHÂN HIỆU HỌC VIỆN KTMM TẠI TP. HỒ CHÍ MINH

ĐỒ ÁN TỐT NGHIỆP

PHÁT TRIỂN ỨNG DỤNG PHÁT HIỆN TIN NHẮN LỪA ĐẢO BẰNG PHƯƠNG PHÁP HỌC MÁY

Ngành: An toàn thông tin

Mã số: 7.48.02.02

Sinh viên thực hiện:

Đàm Chí Nguyên

Lớp: AT15H

Giảng viên hướng dẫn:

Nguyễn Xuân Sâm

Giảng viên trường khoa CNTT, HCMUTE

Thành phố Hồ Chí Minh, 2023

LỜI CẢM ƠN

Trong quá trình thực hiện đồ án tốt nghiệp này, sinh viên đã nhận được sự giúp đỡ tận tình của cán bộ hướng dẫn là thầy Nguyễn Xuân Sâm - Giảng viên trường khoa CNTT, HCMUTE. Trong suốt quá trình thực hiên đề tài, thầy đã giúp đỡ học viên rất nhiều về nội dung và cách áp dụng lý thuyết vào thực hành.

Trong lời đầu tiên, tôi muốn gửi những lời cảm ơn tới tất cả những người đã hỗ trợ, giúp đỡ tôi, tạo những điều kiện tốt nhất để tôi hoàn thành đồ án tốt nghiệp này.

Tuy đã có rất nhiều cố gắng và sự nỗ lực của học viên để hoàn thiện đề tài, nhưng chắc chắn đề tài "Phát triển ứng dụng phát hiện tin nhắn lừa đảo bằng phương pháp học máy" còn nhiều thiếu sót, sinh viên rất mong nhận được sự góp ý, phản hồi từ các thầy cô giáo.

Sinh viên xin chân thành cảm ơn!

Tp. Hồ Chí Minh, ngày 05 tháng 12 năm 2024

Sinh viên thực hiện

Đàm Chí Nguyên

LỜI CAM ĐOAN

Học viên xin cam đoan bản đồ án này do sinh viên tự nghiên cứu dưới sự hướng dẫn của thầy giáo Nguyễn Xuân Sâm.

Để hoàn thành đồ án này, sinh viên chỉ sử dụng những tài liệu đã ghi trong mục tài liệu tham khảo, ngoài ra không sử dụng bất cứ tài liệu nào khác mà không được ghi.

Nếu sai, học viên xin chịu mọi hình thức kỷ luật theo quy định của Học viện.

Tp. Hồ Chí Minh, ngày 05 tháng 12 năm 2024

Sinh viên thực hiện

Đàm Chí Nguyên

MỤC LỤC

DANH MỤC CÁC KÝ HIỆU, CHỮ VIẾT TẮT

Từ viết tắt	**Định nghĩa**
AI	Trí tuệ nhân tạo
CNN	Mạng nơ-ron tích chập
LSTM	Mạng nơ-ron dài ngắn hạn
NLP	Xử lý ngôn ngữ tự nhiên
RNN	Mạng nơ-ron hồi quy
SVM	Máy vector hỗ trợ
TF-IDF	Tần suất từ - Tần suất nghịch tài liệu
URL	Định vị tài nguyên đồng nhất

DANH MỤC HÌNH VẼ, ĐỒ THỊ

LỜI NÓI ĐẦU

Trong môi trường kỹ thuật số ngày càng phát triển, việc đảm bảo an toàn thông tin cá nhân và xác thực danh tính người dùng đã trở thành một yếu tố thiết yếu trong việc bảo vệ không gian mạng. Các cuộc tấn công lừa đảo qua tin nhắn, đặc biệt là thông qua SMS, đang có xu hướng gia tăng cả về số lượng lẫn mức độ phức tạp, đặt ra những thách thức đáng kể cho các chuyên gia bảo mật và các tổ chức trên toàn thế giới. Với mục tiêu đánh cắp thông tin cá nhân nhạy cảm hoặc chiếm đoạt tài sản, các cuộc tấn công này thường dựa vào sự bất cẩn của người dùng và khả năng giả mạo các tổ chức đáng tin cậy.

Trong bối cảnh này, việc nghiên cứu và phát triển các giải pháp phòng ngừa trở nên vô cùng quan trọng. Đồ án này tập trung vào việc phân tích chuyên sâu các phương pháp phát hiện lừa đảo qua tin nhắn, sử dụng các thuật toán học máy tiên tiến nhằm nhận diện các dấu hiệu đặc trưng của tin nhắn lừa đảo. Bên cạnh việc tìm hiểu các yếu tố kỹ thuật như cơ chế hoạt động của các mô hình học máy, em cũng nghiên cứu các tiêu chuẩn bảo mật liên quan và các thách thức trong việc triển khai trên quy mô lớn.

Thông qua việc thu thập và phân tích dữ liệu thực tế, em đã xây dựng và đánh giá hiệu quả của các mô hình nhận diện tin nhắn lừa đảo, từ đó đề xuất các phương pháp cải tiến để giảm thiểu tỷ lệ cảnh báo sai và tăng cường độ chính xác của hệ thống. Em mong rằng những kết quả nghiên cứu này sẽ không chỉ đóng góp vào việc nâng cao nhận thức về an ninh thông tin mà còn cung cấp cơ sở khoa học cho việc phát triển các giải pháp bảo mật hiệu quả hơn trong tương lai. Đồ án này không chỉ là cơ hội để em khám phá sâu hơn về các kỹ thuật bảo mật thông tin mà còn giúp em rèn luyện khả năng nghiên cứu, phân tích, và trình bày một cách chuyên nghiệp và có hệ thống.

CHƯƠNG 1. TỔNG QUAN

1.1. Đặt vấn đề

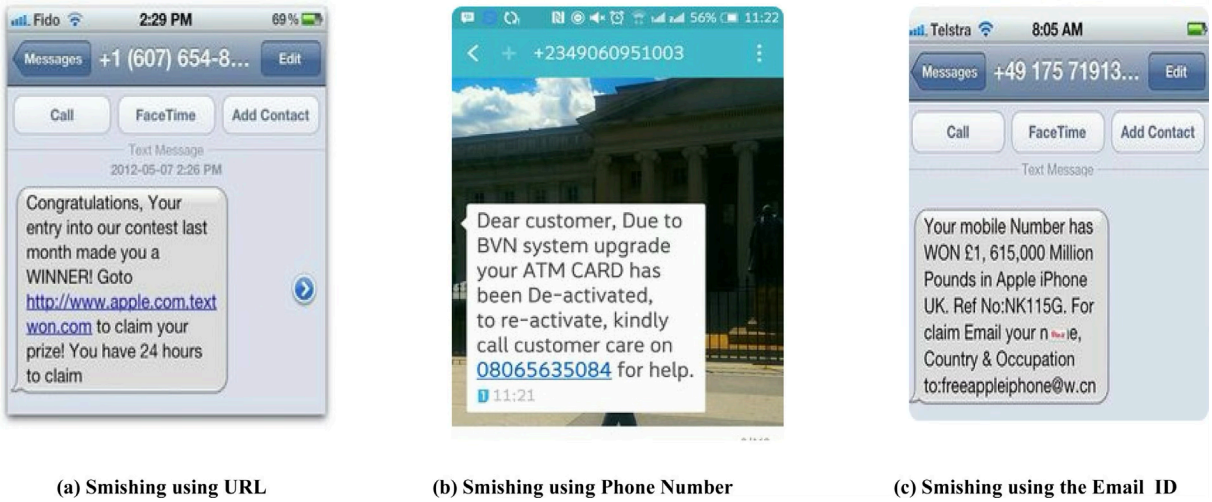
Ngày nay, cùng với sự phát triển không ngừng của công nghệ số đã làm thay đổi cách thức con người giao tiếp và thực hiện các giao dịch hàng ngày. Trong đó, các thiết bị thông minh như máy vi tính, điện thoại thông minh (smart phone),… đóng vai trò quan trọng, trở thành công cụ chính để trao đổi thông tin, thanh toán trực tuyến và lưu trữ các dữ liệu cá nhân quan trọng. Tuy nhiên, cùng với sự nhanh chóng và tiện lợi đó là nguy cơ mất an toàn thông tin, đặc biệt là qua các cuộc tấn công lừa đảo dưới dạng tin nhắn

1.2. Lừa đảo thông qua tin nhắn (Smishing)

Tin nhắn văn bản hoặc SMS (Tin nhắn văn bản ngắn) là một phương thức nhắn tin ngắn gọn giữa các điện thoại di động, thường giới hạn trong khoảng 160 ký tự cho mỗi tin nhắn. SMS phishing còn được gọi là Smishing, một từ ghép giữa SMS và Phishing. David Rayhawk của McAfee đã sử dụng từ "SMISHING" lần đầu tiên vào ngày 25 tháng 8 năm 2006. [4] liên quan các hình thức lừa đảo qua tin nhắn. Trong smishing, kẻ tấn công gửi tin nhắn SMS có chứa liên kết hoặc yêu cầu cung cấp thông tin cá nhân, thường núp dưới danh nghĩa của ngân hàng, công ty dịch vụ tài chính, hoặc các tổ chức uy tín khác. Hình thức tấn công này ngày càng phổ biến, bởi người dùng thường có xu hướng tin tưởng vào những tin nhắn nhận được qua ứng dụng nhắn tin trên điện thoại hơn là qua email.

Ngoài ra, với sự phát triển của các nền tảng mạng xã hội, phishing hiện nay còn mở rộng sang các ứng dụng nhắn tin như Messenger, WhatsApp, và Telegram. Về mặt kỹ thuật không được xem là smishing, nhưng chúng có mối liên hệ chặt chẽ với nhau. Các tài khoản giả mạo của công ty hoặc bạn bè thường được sử dụng để tạo lòng tin, từ đó khiến nạn nhân dễ dàng cung cấp thông tin.

Một tin nhắn Smishing có thể bao gồm các thành phần sau đây:



(a) Smishing using URL **(b) Smishing using Phone Number** **(c) Smishing using the Email ID**

Hình 1: Các thành phần của một tin nhắn Smishing

Liên kết URL nhúng: Trong kỹ thuật này, kẻ tấn công gửi tin nhắn văn bản chứa liên kết URL đến nạn nhân. Khi người dùng đọc tin nhắn và nhấp vào URL, các hoạt động sau sẽ được kích hoạt: Mã độc hại nhúng trong URL sẽ được cài đặt vào điện thoại của nạn nhân.

Một số URL chuyển hướng người dùng đến một trang web phishing độc hại có giao diện giống với một trang web hợp pháp và yêu cầu người dùng điền thông tin đăng nhập hoặc thông tin thẻ tín dụng vào biểu mẫu. Một tệp *.apk được tải xuống điện thoại của nạn nhân, và sau đó gây ra các hoạt động độc hại.

Số điện thoại và/hoặc địa chỉ email: Kẻ tấn công gửi tin nhắn SMS đến nạn nhân, tuyên bố có phiếu giảm giá, quà tặng miễn phí hoặc ưu đãi, và trong tin nhắn đó có đề cập đến số điện thoại hoặc địa chỉ email của kẻ tấn công. Nạn nhân liên hệ với kẻ tấn công qua số điện thoại hoặc email, sau đó kẻ tấn công yêu cầu thông tin đăng nhập của họ.

Tin nhắn tự trả lời: Kẻ tấn công gửi tin nhắn tự trả lời đến nạn nhân yêu cầu họ đăng ký hoặc hủy đăng ký một dịch vụ. Những liên kết này sẽ chuyển hướng người dùng đến các trang web độc hại.

1.3. Tình trạng bảo mật thông tin trên các thiết bị thông minh

Mặc dù nhiều người dùng không nhận thức được mối liên hệ giữa các cuộc tấn công phishing và tin nhắn cá nhân, nhưng thực tế cho thấy kẻ tấn công có thể dễ dàng truy tìm số điện thoại của nạn nhân hơn so với địa chỉ email. Điều này là bởi số điện thoại có một tập hợp giới hạn các sự kết hợp - ví dụ, tại Mỹ, số điện thoại chỉ gồm 10 chữ số. Ngược lại, địa chỉ email không bị giới hạn về độ dài và có thể bao gồm chữ cái, số, cùng với các ký hiệu như !, #, %. Vì vậy, việc tạo ra một chuỗi 10 chữ số ngẫu nhiên để thực hiện tấn công trở nên dễ dàng hơn nhiều so với việc tìm kiếm một địa chỉ email hợp lệ.

Bên cạnh đó, kẻ tấn công có thể dễ dàng gửi tin nhắn đến bất kỳ sự kết hợp nào của các chữ số có độ dài tương ứng với số điện thoại mà không gặp phải rủi ro đáng kể. Theo báo cáo của Gartner, 98% tin nhắn SMS được người dùng đọc và 45% trong số đó nhận được phản hồi, điều này cho thấy SMS là một kênh tấn công lý tưởng đối với kẻ lừa đảo. Trái ngược với đó, chỉ có 6% email nhận được phản hồi từ người nhận.

Trên thực tế 46 triệu người dùng di động của Malaysia đã bị rò rỉ dữ liệu nhạy cảm và quan trọng như địa chỉ, số thẻ định danh, thông tin thẻ, sim điện thoại và thông tin cá nhân khác vào năm 2017; dữ liệu liên quan tới 7,6 triệu chủ tài khoản hiện tại và 65,4 triệu chủ tài khoản trước đây của Công ty viễn thông AT&T - nhà cung cấp dịch vụ không dây bán lẻ lớn thứ 3 của Mỹ đã bị rò rỉ từ trước năm 2019 và mới bị phát hiện vào đầu năm 2024. Theo báo cáo "State of the Phish" năm 2024 của Proofpoint, 75% các tổ chức đã gặp phải các cuộc tấn công smishing trong năm 2023.

Theo thống kê của Techreport, năm 2020, có tới 3 triệu tin nhắn lừa đảo liên quan đến lĩnh vực tài chính - ngân hàng. Thống kê mới nhất của Robokiller - ứng dụng chặn cuộc gọi và tin nhắn rác hàng đầu cho thấy, người Mỹ đã nhận được 225 tỉ tin nhắn rác vào năm 2023 (tăng 157% so với năm 2022) và 19,2 tỉ tin nhắn rác vào tháng 3/2024. Truecaller và Ủy ban Thương mại Liên bang cho biết, 68,4 triệu người Mỹ trở thành nạn nhân của các vụ lừa đảo qua điện thoại với tổng thiệt hại hơn 326 triệu USD. Ở Ailen, chỉ một vụ lừa đảo qua tin nhắn mà thiệt hại lên

đến 800 nghìn USD. Theo báo cáo của Dimensional Enterprise Mobile Security, tấn công Smishing đứng thứ hai trong tất cả các cuộc tấn công vào thiết bị di động.

Tại Việt Nam, vào năm 2021, khách hàng của nhiều ngân hàng thương mại như Ngân hàng Thương mại cổ phần (NHTMCP) Ngoại thương Việt Nam (Vietcombank), NHTMCP Á Châu (ACB), NHTMCP Sài Gòn Thương Tín (Sacombank), NHTMCP Tiên Phong (TPBank) đã nhận được tin nhắn chứa đường dẫn mạo danh ngân hàng (Đại Việt, 2021; N.M, 2021). Những khách hàng này do không biết hoặc chủ quan nên đã nhấp vào và bị chiếm quyền truy cập tài khoản ngân hàng dẫn đến mất số tiền lớn. Năm 2022, Công an tỉnh Tuyên Quang cũng đã phải phát đi cảnh báo thủ đoạn lừa đảo qua SMS. Theo đó, các đối tượng giả lập trạm phát sóng di động của các nhà mạng trong nước rồi phát tán tin nhắn với nội dung thông báo khách hàng đã đăng kí các dịch vụ phát sinh chi phí (từ 3 - 6 triệu đồng) qua tài khoản ngân hàng (Hoàn, 2022).

Mặc dù Smishing là một loại lừa đảo, nhưng nó khác với lừa đảo ở nhiều khía cạnh như lượng thông tin có sẵn trong SMS, chiến lược tấn công, v.v. Do đó, việc phát triển của một ứng dụng có khả năng phát hiện và ngăn chặn tin nhắn lừa đảo trở thành một nhu cầu cấp thiết.

1.4. Phương pháp học máy

Có hai phương pháp phòng vệ chính được sử dụng để phát hiện tin nhắn SMS giả mạo:

Phương pháp đầu tiên là kỹ thuật dựa trên danh sách đen, ngăn chặn các tin nhắn đến từ các nguồn giả mạo. Tuy nhiên, kỹ thuật này không bao quát được tất cả các nguồn giả mạo, vì tội phạm có thể mua bất kỳ số điện thoại di động nào để gửi tin nhắn giả mạo.

Phương pháp thứ hai là sử dụng các thuật toán học máy, trong đó các đặc trưng khác nhau được trích xuất từ tin nhắn SMS để đưa ra quyết định. Ưu điểm của phương pháp này là có thể phát hiện các tin nhắn giả mạo đến từ bất kỳ nguồn nào. Các phương pháp khai thác dữ liệu giúp trích xuất đặc trưng và tìm mối quan

hệ giữa chúng, từ đó nhận diện kiến thức tiềm ẩn từ các tập dữ liệu dưới dạng các quy tắc và đưa ra quyết định dựa trên các quy tắc đó.

Có hai cách tiếp cận phổ biến trong việc xây dựng các thuật toán để phát hiện tin nhắn giả mạo là phương pháp phát hiện dựa trên luật (Rule-based method) và phương pháp lọc dựa trên nội dung (Xia và Chen, 2021).

1.4.1. Phương pháp dựa trên luật

Phương pháp phát hiện dựa trên luật được những tập đoàn công nghệ lớn như Google, Symantec, McAfee ứng dụng để loại bỏ những tin nhắn, thư điện tử rác (M. Hameed và Hussein Ali, 2021) là một trong những cách tiếp cận truyền thống trong việc phát hiện và lọc tin nhắn rác.

Phương pháp này dựa trên các quy tắc được xác định sẵn để phân loại tin nhắn là rác hoặc không rác. Các quy tắc này được xây dựng dựa trên các mẫu và đặc điểm phổ biến trong các tin nhắn rác đã biết như từ ngữ đặc trưng, cấu trúc câu, hoặc các thông tin cụ thể về người gửi. Các quy tắc sẽ giúp hệ thống phân biệt giữa tin nhắn thông thường và tin nhắn có nội dung quảng cáo hoặc lừa đảo.

Ưu điểm:

- Dễ dàng triển khai: Phương pháp này không yêu cầu tập dữ liệu lớn hoặc các thuật toán phức tạp, do đó dễ triển khai và sử dụng trong các hệ thống đơn giản.
- Có tính tức thời: Hệ thống có thể xác định tin nhắn rác ngay khi nhận được mà không cần phải huấn luyện mô hình phức tạp.
- Khả năng kiểm soát cao: Các quy tắc có thể được thiết lập và điều chỉnh để phù hợp với nhu cầu của tổ chức hoặc cá nhân.

Nhược điểm:

- Tăng kích thước tập luật: Để duy trì hiệu quả, các quy tắc cần được cập nhật liên tục nhằm bắt kịp các kỹ thuật mới của tin nhắn rác. Điều này có thể dẫn đến tập hợp quy tắc trở nên phức tạp và khó quản lý.

• Thiếu tính linh hoạt: Phương pháp này dựa trên quy tắc cố định, nên khi có các mẫu tin rác mới, phương pháp này sẽ không thể phát hiện cho đến khi quy tắc được cập nhật.

• Hiệu quả thấp khi đối mặt với nội dung mới: Tin nhắn độc hại thường xuyên thay đổi mẫu, từ ngữ và cấu trúc để tránh bị phát hiện. Điều này làm cho phương pháp dựa trên luật không còn hiệu quả nếu không có cập nhật thường xuyên.

1.4.2. Phương pháp lọc dựa trên nội dung

Phương pháp lọc tin nhắn độc hại dựa trên nội dung sử dụng nền tảng học máy được quan tâm nghiên cứu nhiều hơn trong những năm trở lại đây (Hsu, 2020; Xia và Chen, 2021) Khác với phương pháp phát hiện dựa trên luật, phương pháp này tập trung vào việc phân tích nội dung thực tế của tin nhắn.

Các kỹ thuật và thuật toán phổ biến có thể kể đến như:

• Naive Bayes: Là một trong những thuật toán đơn giản và hiệu quả trong việc phân loại văn bản. Naive Bayes giả định rằng các đặc trưng là độc lập với nhau, điều này giúp giảm độ phức tạp tính toán.

• Support Vector Machines (SVM): Là một thuật toán mạnh mẽ cho các bài toán phân loại, SVM tìm kiếm siêu phẳng (hyperplane) tối ưu để phân loại dữ liệu.

• Deep Learning: Các mô hình mạng nơ-ron, đặc biệt là RNN (Recurrent Neural Network) và LSTM (Long Short-Term Memory), đang được sử dụng ngày càng nhiều trong việc phân loại văn bản, nhờ vào khả năng nắm bắt mối quan hệ trong chuỗi văn bản.

Ưu điểm: Phương pháp lọc dựa trên nội dung mang lại độ chính xác cao, có khả năng phát hiện các tin nhắn rác mới mà không cần cập nhật quy tắc, đồng thời mô hình học máy có thể cải thiện qua thời gian khi được cung cấp thêm dữ liệu mới.

Nhược điểm: Để đạt được độ chính xác cao, phương pháp này yêu cầu một lượng lớn dữ liệu đã gán nhãn để huấn luyện mô hình, và một số thuật toán, đặc biệt là Deep Learning, có thể cần tài nguyên tính toán lớn.

Ngày nay với nhiều nghiên cứu khảo sát có thể thấy, hầu hết những giải pháp hiện có được đề xuất cho bài toán phát hiện tin nhắn rác đạt kết quả chính xác rất cao, thậm chí tiệm cận tới 100% có thể kể đến như:

- Nghiên cứu xác định phát hiện tin nhắn lừa đảo của Sandhya Mishra and Devpriya Soni bằng cách sử dụng mô hình bảo mật thông qua phân tích nội dung tin nhắn và phân tích hành vi URL đã đạt được độ hiệu quả tổng thể lên đến 96.29%.
- Thực hiện thử nghiệm với hai phương pháp phân lớp SVM và Naive Bayes trên kho SMS của Almeida và cộng sự (2011), Sonowal, Kuppusamy và cộng sự (2018) đã phát triển hai mô hình phân loại tin nhắn rác có độ chính xác 94,2%. Cùng sử dụng bộ dữ liệu tương tự, Jain và Gupta (2019) cũng đạt được độ chính xác 94,2% khi áp dụng cây quyết định và SVM vào bài toán phát hiện tin nhắn rác.
- Sjarif và cộng sự (2019) cùng Mishra và Soni (2020), sử dụng kỹ thuật phân lớp Naive Bayes trên cùng bộ dữ liệu của Almeida và cộng sự (2011), đã đạt được các mô hình phát hiện tin nhắn rác với độ chính xác lần lượt là 97,5% và 96,29%.
- Xia và Chen đã liên tục cải thiện mô hình Markov ẩn trên kho SMS của Almeida và cộng sự (2011) để phát triển công cụ phân loại tin nhắn rác, đạt độ chính xác 95,9% trong nghiên cứu (Xia và Chen, 2020) và 96,9% trong nghiên cứu (Xia và Chen, 2021).
- Roy và cộng sự (2020) đã phát triển một loạt mô hình phát hiện tin nhắn rác dựa trên các kỹ thuật học sâu như LSTM, CNN và mô hình kết hợp LSTM-CNN, sử dụng bộ dữ liệu SMS của Almeida và cộng sự (2011). Mô hình kết hợp LSTM-CNN đạt độ chính xác cao nhất là 99,44%.
- Cùng sử dụng phương pháp kết hợp LSTM và CNN, Ghourabi và cộng sự (2020) đã đề xuất một công cụ lọc tin nhắn rác dựa trên cùng bộ dữ liệu, đạt độ chính xác 98,37%. Kết quả này tương đương với các mô hình phân loại tin nhắn

rác sử dụng thuật toán KNN của Sousa và cộng sự (2021), mô hình tuần tự trong học sâu của Liu và cộng sự (2021), CNN của Giri và cộng sự (2023), và mô hình kết hợp giữa CNN và Bi-LSTM của Mambina và cộng sự (2024).

- Một số nghiên cứu đã phát triển mô hình phát hiện tin nhắn rác gần như chính xác tuyệt đối khi sử dụng kho SMS của Almeida và cộng sự (2011), như công trình của Ghourabi và Alohaly (2023) kết hợp các kỹ thuật SVM, KNN, LightGBM và CNN; nghiên cứu của Maqsood và cộng sự (2023) áp dụng kỹ thuật SVM; hay công trình của Hussein và cộng sự (2023) kết hợp LSTM và CNN.

Ngoài các mô hình phát hiện tin nhắn rác tiếng Anh, đã có một số nghiên cứu phát hiện tin nhắn rác tiếng Indonesia. Hikmaturokhman và cộng sự (2022) sử dụng các kỹ thuật mạng nơ-ron dày đặc, LSTM và Bi-LSTM với độ chính xác lần lượt là 95,63%, 94,76% và 94,75%. Tuấn và cộng sự (2022) đã phát triển mô hình phát hiện tin nhắn rác tiếng Việt dựa trên sự kết hợp DNN và PhoBERT với độ chính xác 99,53%. Mambina và cộng sự (2024) xây dựng công cụ phát hiện tin nhắn rác tiếng Tanzania bằng cách kết hợp CNN và LSTM, đạt độ chính xác 99,98%. Cuối cùng, Ayaz và cộng sự (2024) sử dụng kho tin nhắn ngôn ngữ Latinh riêng tư để huấn luyện mô hình SVM và Naive Bayes, đạt được độ chính xác lần lượt là 97,33% và 99,42%.

Tuy nhiên, những số liệu thống kê lại phản ánh thực tế vấn nạn tin nhắn rác và mức độ thiệt hại của người dùng thiết bị di động không có dấu hiệu thuyên giảm trong những năm gần đây Những nghiên cứu điển hình được phân tích ở trên sử dụng hầu hết các bộ dữ liệu tin nhắn bị hạn chế về mặt số lượng tin nhắn và dữ liệu tin nhắn không được cập nhật thường xuyên (Salman và cộng sự, 2024), trong khi đó tin tặc lại sử dụng nhiều thủ đoạn khác nhau nhằm thay đổi liên tục mẫu tin nhắn rác để tránh bị phát hiện.

1.5. Mục tiêu đề tài

Mục tiêu chính của đề tài này là xây dựng một ứng dụng phát hiện tin nhắn lừa đảo bằng cách áp dụng các thuật toán học máy dựa trên bộ dữ liệu tin nhắn

phức hợp được cập nhật và tích hợp vào các nền tảng SMS hoặc công cụ chatbot tiện dụng. Cụ thể, đề tài sẽ tập trung vào những mục tiêu chính như:

- Phát hiện hiệu quả các tin nhắn SMS spam và lừa đảo: Phân tích và ứng dụng các mô hình học máy như Naive Bayes, Random Forest, LSTM, và Transformer để phát hiện và phân loại tin nhắn lừa đảo dựa trên nội dung và hành vi URL. Mục đích giúp tăng khả năng phát hiện các tin nhắn chứa đường link giả mạo hoặc yêu cầu cung cấp thông tin cá nhân.
- Tối ưu hóa độ chính xác và giảm tỷ lệ báo động sai: Mục tiêu tiếp theo là cải thiện độ chính xác của hệ thống phát hiện lừa đảo, giảm thiểu tỷ lệ báo động sai (false-positive) thông qua việc sử dụng các mô hình học sâu (Deep Learning) và tích hợp các thuật toán kiểm tra hành vi URL. Điều này đảm bảo rằng chỉ những tin nhắn thực sự nguy hiểm mới bị chặn.
- Tích hợp hệ thống phát hiện lừa đảo với các nền tảng SMS hiện có: Phát triển một hệ thống có khả năng tích hợp trực tiếp vào các nền tảng SMS hiện tại của các tổ chức tài chính hoặc nhà cung cấp dịch vụ viễn thông. Điều này sẽ giúp phát hiện và ngăn chặn các tin nhắn lừa đảo trước khi chúng đến tay người dùng, đồng thời bảo vệ dữ liệu cá nhân và tài chính của người dùng.
- Nâng cao nhận thức và bảo mật cho người dùng di động: Hướng đến việc tạo ra một hệ thống không chỉ phát hiện mà còn cung cấp các cảnh báo và giáo dục người dùng về cách nhận diện tin nhắn lừa đảo. Mục tiêu này giúp người dùng nắm vững kiến thức cơ bản về bảo mật và nâng cao khả năng tự bảo vệ trước các cuộc tấn công.

1.6. Các đóng góp chính của đề tài

1.6.1. Mô tả kịch bản tấn công Smishing

Phân loại các kiểu tấn công Smishing: Đề tài phân tích chi tiết các kịch bản lừa đảo thường gặp, bao gồm những tin nhắn giả mạo ngân hàng, cơ quan chính phủ, dịch vụ tài chính, và các tin nhắn yêu cầu người dùng cung cấp thông tin cá

nhân hoặc truy cập vào các liên kết giả mạo. Mỗi kịch bản được mô tả cụ thể, nhằm hiểu rõ cơ chế tấn công và hành vi của kẻ xấu.

Nhận diện các yếu tố nhận dạng Smishing: Đề tài nghiên cứu và mô tả các yếu tố đặc trưng của tin nhắn lừa đảo, bao gồm nội dung chứa URL độc hại, từ ngữ khẩn cấp như "khóa tài khoản", "trúng thưởng", "liên hệ ngay", nhằm đánh lừa và kích thích người dùng tương tác.

1.6.2. Thu thập và xử lý dữ liệu

Dữ liệu là yếu tố trung tâm quyết định chất lượng của hệ thống học máy. Phần này tập trung vào việc thu thập dữ liệu tin nhắn lừa đảo và hợp lệ, cũng như quy trình tiền xử lý để đảm bảo rằng mô hình học máy hoạt động hiệu quả nhất.

1.6.2.1. Các nguồn dữ liệu tin nhắn lừa đảo và hợp lệ

Để phát hiện chính xác tin nhắn lừa đảo, cần phải có một tập dữ liệu đa dạng và phong phú bao gồm cả tin nhắn hợp lệ và tin nhắn lừa đảo.

Dữ liệu từ các nguồn công khai và học thuật:

- Kaggle datasets: Đây là một nguồn quan trọng cung cấp các tập dữ liệu về tin nhắn SMS bao gồm cả lừa đảo và hợp lệ. Kaggle có nhiều tập dữ liệu liên quan đến tin nhắn lừa đảo (spam) được thu thập từ nhiều quốc gia khác nhau, bao gồm cả ngôn ngữ Anh, Việt, và nhiều ngôn ngữ khác.
- SpamAssassin Public Corpus: Là một trong những tập dữ liệu nổi tiếng về spam và lừa đảo, chứa các tin nhắn từ nhiều nguồn email khác nhau. Tuy dữ liệu ban đầu là email, nhưng nhiều mô hình xử lý văn bản có thể tái sử dụng các phương pháp này cho tin nhắn SMS.
- Nghiên cứu học thuật và báo cáo từ các tổ chức an ninh: Các nghiên cứu từ các tổ chức lớn như Symantec, McAfee, và nhiều trường đại học đã cung cấp các tập dữ liệu về Smishing (SMS Phishing), trong đó chứa đựng các mẫu tin nhắn lừa đảo.

Dữ liệu tự tạo hoặc thu thập thủ công:

• Một số tổ chức có thể tự xây dựng tập dữ liệu bằng cách giả lập các cuộc tấn công lừa đảo qua tin nhắn, nhằm hiểu rõ cách thức tấn công và tạo ra các mẫu lừa đảo chuẩn mực.

• Thu thập từ các diễn đàn, trang web và mạng xã hội: Nhiều kẻ tấn công sử dụng các diễn đàn, website hoặc mạng xã hội để chia sẻ các mẫu tin nhắn lừa đảo. Việc theo dõi và thu thập các mẫu này từ những nguồn trên cũng là một cách để xây dựng tập dữ liệu.

1.6.2.2. Quy trình tiền xử lý dữ liệu

Tiền xử lý dữ liệu là bước rất quan trọng trong quá trình phát triển mô hình học máy, đặc biệt là với dữ liệu văn bản như tin nhắn SMS. Việc tiền xử lý sẽ giúp làm sạch dữ liệu, loại bỏ nhiễu, và trích xuất các đặc trưng quan trọng phục vụ cho việc học máy. Một số các bước trong quy trình tiền xử lý bao gồm:

Làm sạch văn bản (Text Cleaning):

• Loại bỏ ký tự đặc biệt: Tin nhắn SMS thường chứa các ký tự đặc biệt, chẳng hạn như biểu tượng, số điện thoại, hoặc các ký tự đặc biệt mà không cần thiết cho mô hình học máy. Loại bỏ các ký tự này giúp giảm nhiễu và tập trung vào nội dung chính của tin nhắn.

• Xử lý viết tắt và từ viết sai chính tả: Tin nhắn SMS thường chứa các từ viết tắt hoặc viết sai chính tả. Các kỹ thuật như lemmatization hoặc stemming sẽ giúp chuyển đổi các từ này về dạng chuẩn để giúp mô hình hiểu rõ hơn nội dung.

Loại bỏ từ dừng (Stopwords Removal): Các từ dừng là những từ phổ biến trong văn bản nhưng không mang nhiều ý nghĩa cho việc phân loại, chẳng hạn như: “và”, “là”, “nhưng”, “hoặc” (trong tiếng Việt), hay “is”, “and”, “but”, “or” (trong tiếng Anh). Việc loại bỏ từ dừng giúp giảm số lượng từ cần xử lý mà vẫn giữ lại ý nghĩa chính của tin nhắn.

Chuyển đổi văn bản thành số:

• Bag of Words (BoW): Mô hình BoW tạo ra ma trận từ điển trong đó mỗi từ trong tập dữ liệu được biểu diễn bằng một cột, và mỗi tin nhắn được biểu diễn

bằng số lần xuất hiện của các từ này. Tuy nhiên, BoW có hạn chế là không xem xét được thứ tự của các từ.

• TF-IDF (Term Frequency-Inverse Document Frequency): Đây là một trong những phương pháp phổ biến để trích xuất đặc trưng từ văn bản. TF-IDF không chỉ tính toán tần suất của từ trong tin nhắn mà còn điều chỉnh mức độ quan trọng của từ đó dựa trên tần suất xuất hiện của nó trong toàn bộ tập dữ liệu. TF-IDF giúp tăng trọng số cho các từ có ý nghĩa trong việc phân loại tin nhắn lừa đảo.

Chuẩn hóa và xử lý biến số: Sau khi chuyển văn bản thành số liệu, việc chuẩn hóa dữ liệu là cần thiết để đảm bảo rằng tất cả các đặc trưng (features) có cùng tỷ lệ và trọng số. Điều này rất quan trọng đối với các mô hình học máy như SVM và Naive Bayes để đảm bảo rằng các đặc trưng không bị mất cân bằng ảnh hưởng đến kết quả.

Xử lý dữ liệu bị thiếu (Missing Data): Dữ liệu bị thiếu hoặc không đầy đủ có thể dẫn đến sai lệch trong việc huấn luyện mô hình. Quá trình này bao gồm việc kiểm tra và xử lý các trường hợp dữ liệu trống hoặc không hợp lệ.

1.6.3. Phân tích và đề xuất mô hình

1.6.3.1. Phân tích đặc trưng của tin nhắn

Việc phân tích đặc trưng của tin nhắn là cần thiết để nhận biết các dấu hiệu giúp phát hiện tin nhắn lừa đảo:

• Phân tích từ khóa: Các tin nhắn lừa đảo thường chứa các từ khóa hoặc cụm từ gây hoang mang hoặc thúc giục người dùng hành động ngay lập tức, chẳng hạn như “trúng thưởng”, “khóa tài khoản”, “liên hệ ngay”. Những từ khóa này là tín hiệu rõ ràng của hành vi lừa đảo.

• URL trong tin nhắn: Các tin nhắn lừa đảo thường chứa các liên kết (URL) giả mạo, dẫn đến các trang web giả để lừa người dùng cung cấp thông tin cá nhân. Việc phân tích cấu trúc và độ tin cậy của các URL trong tin nhắn giúp phát hiện các tin nhắn lừa đảo hiệu quả hơn.

• Phân tích ngữ nghĩa: Việc hiểu rõ ngữ cảnh và nội dung của tin nhắn cũng là một bước quan trọng. Ví dụ, các tin nhắn yêu cầu cung cấp mã OTP, mật khẩu, hoặc số thẻ tín dụng thường là dấu hiệu rõ ràng của lừa đảo.

• Độ dài và cấu trúc của tin nhắn: Các tin nhắn lừa đảo thường có cấu trúc khác biệt so với tin nhắn hợp lệ. Chúng thường ngắn gọn, sử dụng từ ngữ khẩn cấp hoặc đôi khi dài dòng nhưng chứa nội dung không rõ ràng.

1.6.3.2. Ứng dụng thực tiễn

• Phát triển ứng dụng phát hiện lừa đảo: Đề tài phát triển một ứng dụng thực tiễn, dễ sử dụng để người dùng có thể nhập tin nhắn và kiểm tra xem tin nhắn đó có phải lừa đảo hay không. Giao diện người dùng được thiết kế đơn giản, thân thiện, và có khả năng xử lý nhanh chóng các tin nhắn đầu vào.

• Cảnh báo bảo mật và tích hợp: Ứng dụng tích hợp các tính năng cảnh báo bảo mật cho người dùng, giúp họ phòng ngừa các mối đe dọa tiềm tàng từ các tin nhắn lừa đảo.

• Lựa chọn mô hình tối ưu: Đề tài đánh giá và so sánh giữa các mô hình học máy truyền thống và học sâu, nhằm tìm ra mô hình tối ưu nhất dựa trên độ chính xác, tốc độ và khả năng mở rộng. Sau khi đánh giá, mô hình có độ chính xác cao nhất và phù hợp với yêu cầu của đề tài sẽ được sử dụng để triển khai ứng dụng thực tiễn.

CHƯƠNG 2. TỔNG QUAN VỀ CÔNG NGHỆ HỌC MÁY

2.1. Giới thiệu

Học máy (Machine Learning) là một lĩnh vực con của trí tuệ nhân tạo (AI) tập trung vào việc phát triển các thuật toán cho phép máy tính học từ dữ liệu mà không cần được lập trình cụ thể cho từng nhiệm vụ. Học máy cho phép hệ thống tự động cải thiện hiệu suất khi có thêm dữ liệu, từ đó giúp đưa ra dự đoán hoặc quyết định một cách chính xác.

Vòng đời học máy là một quy trình tuần hoàn nhằm xây dựng một dự án học máy hiệu quả. Mục đích chính của vòng đời này là tìm ra giải pháp cho vấn đề hoặc dự án:

1. Thu thập Dữ liệu: Thu thập và xác định các vấn đề liên quan đến dữ liệu.
2. Chuẩn bị Dữ liệu: Sắp xếp dữ liệu để sử dụng trong huấn luyện.
3. Dọn dẹp Dữ liệu: Làm sạch và chuyển đổi dữ liệu thành định dạng có thể sử dụng.
4. Phân tích Dữ liệu: Sử dụng các kỹ thuật phân tích để xây dựng mô hình.
5. Huấn luyện Mô hình: Cải thiện hiệu suất của mô hình.
6. Kiểm tra Mô hình: Đánh giá độ chính xác của mô hình bằng cách sử dụng tập dữ liệu kiểm tra.
7. Triển khai: Đưa mô hình vào hệ thống thực tế.

2.2. Phân loại

Học máy được chia thành các loại chính sau:

2.2.1. Học có giám sát (Supervised Learning)

Học giám sát là một phương pháp học máy trong đó mô hình được huấn luyện dựa trên một tập dữ liệu đã được gán nhãn. Mỗi mẫu trong tập dữ liệu bao gồm

đầu vào và đầu ra tương ứng, cho phép mô hình học cách dự đoán đầu ra từ các đầu vào mới.

Quá trình học giám sát thường được chia thành các bước chính:

- Chuẩn bị dữ liệu: Tập hợp và làm sạch dữ liệu, bao gồm gán nhãn cho các mẫu dữ liệu.
- Phân chia dữ liệu: Tách dữ liệu thành tập huấn luyện và tập kiểm tra để đánh giá hiệu suất mô hình.
- Lựa chọn mô hình: Chọn một thuật toán học máy phù hợp, như cây quyết định, hồi quy logistic, hay mạng nơ-ron.
- Huấn luyện mô hình: Sử dụng tập huấn luyện để điều chỉnh trọng số của mô hình nhằm tối ưu hóa khả năng dự đoán.
- Đánh giá mô hình: Sử dụng tập kiểm tra để đo lường độ chính xác và khả năng tổng quát của mô hình.

Các thuật toán học giám sát phổ biến:

- Cây quyết định: Thuật toán đơn giản nhưng hiệu quả trong việc phân loại và hồi quy.
- Hồi quy logistic: Thích hợp cho bài toán phân loại nhị phân.
- Mạng nơ-ron: Mô hình mạnh mẽ cho cả phân loại và hồi quy, đặc biệt là trong các bài toán phức tạp như nhận diện hình ảnh và ngôn ngữ.

Ưu điểm:

- Độ chính xác cao khi có dữ liệu chất lượng.
- Dễ dàng giải thích và áp dụng cho nhiều lĩnh vực khác nhau.

Nhược điểm:

- Cần một lượng lớn dữ liệu đã được gán nhãn, tốn thời gian và công sức.
- Khả năng tổng quát có thể bị hạn chế nếu mô hình không được huấn luyện đầy đủ.

Học giám sát là phương pháp chủ yếu trong việc phát hiện tin nhắn lừa đảo. Trong phương pháp này, các mô hình được huấn luyện trên tập dữ liệu có gán nhãn, cho phép chúng học từ các ví dụ thực tế. Thuật toán có thể sử dụng:

• Naive Bayes: Một thuật toán đơn giản nhưng hiệu quả, Naive Bayes dựa trên lý thuyết xác suất để phân loại tin nhắn. Nó thích hợp cho các bài toán phân loại văn bản nhờ vào khả năng xử lý các đặc trưng từ văn bản một cách nhanh chóng.

• Support Vector Machines (SVM): SVM tìm kiếm siêu phẳng tốt nhất để phân loại dữ liệu, có hiệu quả cao trong không gian đặc trưng cao và giúp tách biệt các tin nhắn lừa đảo khỏi các tin nhắn hợp lệ.

• Decision Trees: Là một phương pháp trực quan, Decision Trees xây dựng một cây phân quyết để phân loại các tin nhắn dựa trên các đặc trưng của chúng. Mặc dù dễ hiểu, nó cũng có thể gặp phải vấn đề quá khớp.

• Random Forest: Là phiên bản nâng cao của Decision Trees, Random Forest sử dụng nhiều cây quyết định để cải thiện độ chính xác và giảm thiểu hiện tượng quá khớp.

2.2.2. Học không giám sát (Unsupervised Learning)

Học không giám sát là một phương pháp học máy trong đó mô hình được huấn luyện trên một tập dữ liệu không có nhãn. Mục tiêu của phương pháp này là tìm ra cấu trúc hoặc mẫu trong dữ liệu mà không cần bất kỳ hướng dẫn nào từ nhãn.

Quá trình học không giám sát có thể được chia thành các bước chính:

• Chuẩn bị dữ liệu: Tập hợp và làm sạch dữ liệu, đảm bảo tính đầy đủ và nhất quán.

• Phân tích dữ liệu: Sử dụng các kỹ thuật như phân tích thành phần chính (PCA) để giảm số chiều dữ liệu và tìm ra các đặc trưng quan trọng.

• Chọn thuật toán: Lựa chọn một thuật toán phù hợp như clustering hoặc các kỹ thuật học sâu.

• Huấn luyện mô hình: Áp dụng thuật toán để nhóm hoặc phân loại dữ liệu mà không cần nhãn.

• Đánh giá kết quả: Sử dụng các chỉ số như silhouette score hoặc Davies-Bouldin index để đánh giá chất lượng của các nhóm được tạo ra.

Các thuật toán học không giám sát phổ biến:

• K-means: Một thuật toán phân cụm đơn giản nhưng hiệu quả, sử dụng khoảng cách Euclid để xác định độ tương đồng giữa các điểm dữ liệu.

• DBSCAN: Được sử dụng để phát hiện các cụm có hình dạng phức tạp và có khả năng phát hiện nhiễu (noise).

• Phân cụm theo chiều sâu: Cho phép tạo ra các cấu trúc phân cụm ở nhiều cấp độ khác nhau.

Ưu điểm:

• Không cần dữ liệu đã gán nhãn, tiết kiệm thời gian và công sức trong việc chuẩn bị dữ liệu.

• Có khả năng phát hiện các mẫu và cấu trúc ẩn mà không bị giới hạn bởi nhãn.

Nhược điểm:

• Khó khăn trong việc đánh giá chất lượng của các kết quả, vì không có nhãn để đối chiếu.

• Kết quả có thể nhạy cảm với lựa chọn tham số và thuật toán, đôi khi cần thử nghiệm nhiều phương pháp để đạt được kết quả tốt nhất.

2.2.3. Học bán giám sát (Semi-Supervised Learning)

Kết hợp giữa học có giám sát và học không giám sát, sử dụng cả dữ liệu có nhãn và không có nhãn để cải thiện hiệu suất mô hình.

2.2.4. Học tăng cường (Reinforcement Learning)

• Một tác nhân học cách hành động trong một môi trường thông qua việc nhận phần thưởng hoặc phạt từ các hành động của nó.

• Ứng dụng: Chơi game, robot tự động.

• Thuật toán phổ biến: Q-learning, Deep Q-Networks (DQN), Policy Gradients.

2.2.4.1. Học sâu (Deep Learning)

Học sâu là một nhánh của học máy, trong đó sử dụng các mạng nơ-ron nhân tạo để mô hình hóa và giải quyết các bài toán phức tạp. Khác với các phương pháp học máy truyền thống, học sâu có khả năng tự động trích xuất các đặc trưng từ dữ liệu mà không cần sự can thiệp của con người, từ đó cải thiện độ chính xác trong các nhiệm vụ phân loại, nhận dạng và dự đoán.

Mạng nơ-ron là thành phần cơ bản trong học sâu. Cấu trúc mạng nơ-ron bao gồm:

- Đầu vào (Input Layer): Nhận dữ liệu đầu vào từ bộ dữ liệu.
- Các lớp ẩn (Hidden Layers): Thực hiện các phép toán để trích xuất và chuyển đổi đặc trưng từ dữ liệu. Số lượng lớp và số lượng nơ-ron trong mỗi lớp có thể thay đổi tùy thuộc vào bài toán.
- Đầu ra (Output Layer): Cung cấp kết quả cuối cùng, như phân loại tin nhắn là lừa đảo hoặc không lừa đảo.

Học sâu bao gồm nhiều loại mạng nơ-ron, mỗi loại phù hợp với những bài toán khác nhau:

- Mạng nơ-ron tích chập (Convolutional Neural Networks - CNN): Thường được sử dụng trong nhận diện hình ảnh và phân loại, CNN có khả năng tự động phát hiện và học các đặc trưng từ dữ liệu đầu vào thông qua các phép toán chập.
- Mạng nơ-ron hồi tiếp (Recurrent Neural Networks - RNN): Phù hợp cho các dữ liệu có tính tuần tự, như chuỗi thời gian và văn bản. RNN có khả năng ghi nhớ thông tin từ các bước trước đó, giúp xử lý các bài toán như dịch máy và phân tích ngữ nghĩa.
- Mạng nơ-ron dài ngắn hạn (Long Short-Term Memory - LSTM): Là một dạng cải tiến của RNN, LSTM giúp giải quyết vấn đề mất mát thông tin trong các chuỗi dài. Nó cho phép mạng ghi nhớ thông tin trong khoảng thời gian dài hơn, rất hữu ích trong các bài toán xử lý ngôn ngữ tự nhiên.

Thuật toán học sâu phổ biến

• Xử lý ngôn ngữ tự nhiên (NLP): Các mô hình như BERT (Bidirectional Encoder Representations from Transformers) đã cách mạng hóa cách thức xử lý ngôn ngữ tự nhiên, cho phép phát hiện các mối quan hệ phức tạp trong dữ liệu văn bản.

Ưu điểm:

• Học sâu có khả năng xử lý và học từ dữ liệu lớn, giúp phát hiện mẫu lừa đảo một cách hiệu quả.

• Tự động trích xuất đặc trưng, giảm thiểu sự cần thiết phải can thiệp thủ công từ con người.

Nhược điểm:

• Cần một lượng lớn dữ liệu để huấn luyện và có thể tốn kém về tài nguyên tính toán.

• Mô hình phức tạp có thể khó giải thích, dẫn đến việc khó xác định lý do vì sao một tin nhắn cụ thể được phân loại là lừa đảo.

Gần đây, các mô hình học sâu đã trở thành công cụ mạnh mẽ trong phát hiện tin nhắn lừa đảo nhờ khả năng học từ dữ liệu lớn và tự động trích xuất các đặc trưng phức tạp.

• Recurrent Neural Networks (RNN): RNN được thiết kế để xử lý dữ liệu chuỗi, giúp nhận diện các mẫu trong văn bản theo ngữ cảnh. Tuy nhiên, RNN có thể gặp khó khăn với các chuỗi dài do vấn đề gradient biến mất.

• Long Short-Term Memory (LSTM): Là một biến thể của RNN, LSTM khắc phục vấn đề gradient biến mất, cho phép mô hình ghi nhớ thông tin lâu hơn và xử lý hiệu quả các chuỗi dài trong văn bản.

• Convolutional Neural Networks (CNN): Mặc dù thường được sử dụng trong xử lý hình ảnh, CNN cũng cho thấy hiệu quả trong việc phân tích văn bản nhờ khả năng phát hiện các đặc trưng cục bộ.

• BERT (Bidirectional Encoder Representations from Transformers): Là một trong những mô hình tiên tiến nhất trong NLP, BERT sử dụng cơ chế attention để học ngữ cảnh từ cả hai chiều. Mô hình này đã đạt được độ chính

xác cao trong nhiều bài toán phân loại văn bản, bao gồm cả phát hiện tin nhắn lừa đảo.

2.3. Nguyên lý thuật toán học máy

2.3.1. Mô hình Naive Bayes

Mô hình phân lớp Naive Bayes là một phương pháp phân loại xác suất dựa trên Định lý Bayes và giả định tính độc lập của các thuộc tính. Đây là một phương pháp đơn giản nhưng mạnh mẽ, thường được sử dụng trong các bài toán phân loại văn bản như phân loại email spam, phân tích cảm xúc và phát hiện tin nhắn lừa đảo.

Định lý Bayes cung cấp một cách tính toán xác suất của một sự kiện dựa trên kiến thức có sẵn về các sự kiện liên quan, được biểu diễn bằng công thức, trong đó:

- P(y|X) gọi là posterior probability: xác suất của mục tiêu y với điều kiện có đặc trưng X
- P(X|y) gọi là likelihood: xác suất của đặc trưng X khi đã biết mục tiêu y
- P(y) gọi là prior probability của mục tiêu y
- P(X) gọi là prior probability của đặc trưng X

Trong mô hình Naive Bayes, có hai giả thiết được đặt ra:

- Các đặc trưng đưa vào mô hình là độc lập với nhau. Tức là sự thay đổi giá trị của một đặc trưng không ảnh hưởng đến các đặc trưng còn lại.
- Các đặc trưng đưa vào mô hình có ảnh hưởng ngang nhau đối với đầu ra mục tiêu.

Cả hai giả thiết gần như không tồn tại trong thực tế trên, mô hình này mới được gọi là naive (ngây thơ). Tuy nhiên, chính sự đơn giản của nó với việc dự đoán rất nhanh kết quả đầu ra khiến nó được sử dụng rất nhiều trong thực tế trên những bộ dữ liệu lớn, đem lại kết quả khả quan.

Một số kiểu mô hình Naive Bayes:

• Multinomial Naive Bayes: Mô hình này chủ yếu được sử dụng trong phân loại văn bản. Đặc trưng đầu vào ở đây chính là tần suất xuất hiện của từ trong văn bản đó.

• Bernoulli Naive Bayes: Mô hình này được sử dụng khi các đặc trưng đầu vào chỉ nhận giá trị nhị phân 0 hoặc 1 (phân bố Bernoulli).

• Gaussian Naive Bayes: Khi các đặc trưng nhận giá trị liên tục, ta giả sử các đặc trưng đó có phân phối Gaussian. Khi đó, likelihood sẽ có dạng:

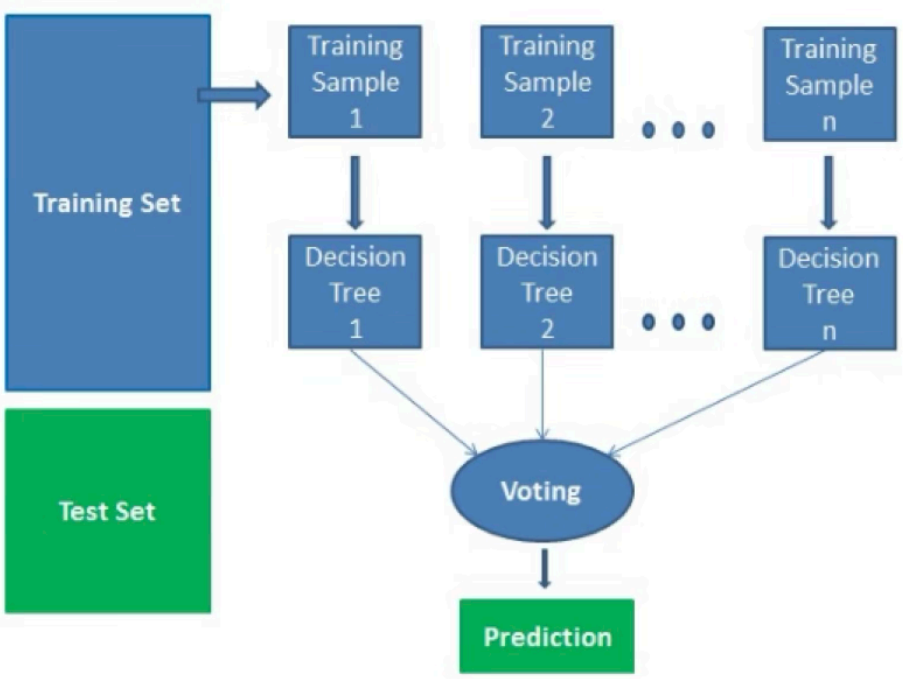
Mô hình Naive Bayes là mô hình phân lớp đơn giản dễ cài đặt, có tốc độ xử lý nhanh. Tuy nhiên có nhược điểm lớn là yêu cầu các đặc trưng đầu vào phải độc lập, mà điều này khó xảy ra trong thực tế làm giảm chất lượng của mô hình.

2.3.2. Mô hình Random Forest

Thuật toán Random Forest là một trong những phương pháp học máy mạnh mẽ và phổ biến, đặc biệt trong các bài toán phân loại và hồi quy. Được xây dựng dựa trên ý tưởng của thuật toán "tập hợp" (ensemble learning), Random Forest phát huy hiệu quả bằng cách kết hợp nhiều cây quyết định (Decision Trees) nhằm cải thiện độ chính xác và giảm thiểu rủi ro của việc quá khớp (overfitting).

Thuật toán hoạt động tuần tự theo bốn bước:

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



Hình 2: Mô hình Random Forest

Random Forests được xem như một phương pháp mạnh mẽ và có độ chính xác cao nhờ sự tham gia của nhiều cây quyết định trong quá trình phân tích.

2.3.3. Mô hình Cây Quyết Định

Thuật toán Decision Tree (Cây Quyết Định) là một trong những phương pháp học máy đơn giản nhưng rất hiệu quả, thường được sử dụng trong các bài toán phân loại và hồi quy. Cấu trúc của mô hình này giống như một cây phân cấp, nơi mỗi nút biểu thị một đặc tính của dữ liệu, mỗi nhánh biểu thị kết quả của một phép thử, và mỗi lá biểu thị một kết quả hoặc quyết định cuối cùng.

Quy trình hoạt động của thuật toán Decision Tree có thể được chia thành các bước chính như sau:

Bước 1: Chọn Thuộc Tính Tốt Nhất: Ở mỗi nút, mô hình chọn đặc tính tốt nhất để chia dữ liệu. Tiêu chí chọn thường dựa trên mức độ giảm độ bất định (impurity) sau khi chia dữ liệu. Hai tiêu chí phổ biến là:

• Information Gain (tăng thông tin), sử dụng trong cây phân loại dựa trên lý thuyết thông tin (entropy).

• Gini Index, sử dụng trong thuật toán CART (Classification and Regression Tree).

Bước 2: Chia Dữ Liệu: Mỗi nút chia dữ liệu thành hai hoặc nhiều nhánh con dựa trên đặc tính được chọn.

Bước 3: Lặp Lại Quy Trình: Quá trình lặp lại này tiếp tục cho đến khi một điều kiện dừng được thỏa mãn, chẳng hạn như:

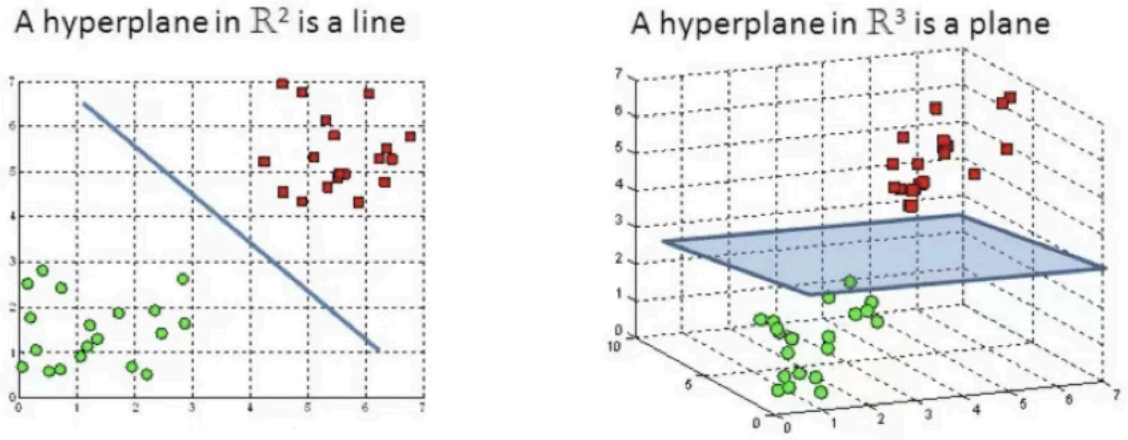
• Tất cả các điểm dữ liệu trong một nút thuộc cùng một lớp.

• Không còn thuộc tính nào để chia hoặc độ sâu tối đa của cây đã đạt.

Bước 4: Tạo Quyết Định: Các lá của cây là những dự đoán cuối cùng, giúp phân loại dữ liệu đầu vào mới.

2.3.4. Mô hình Support Vector Machine

Support Vector Machine (SVM) là một thuật toán học máy được sử dụng chủ yếu cho các bài toán phân loại và đôi khi cho các bài toán hồi quy.



Hình 3: Mô hình Support Vector Machine trong không gian hai chiều và ba chiều

SVM giải bài toán tối ưu hóa để tìm siêu phẳng có dạng:

$w \cdot x - b = 0$

trong đó:

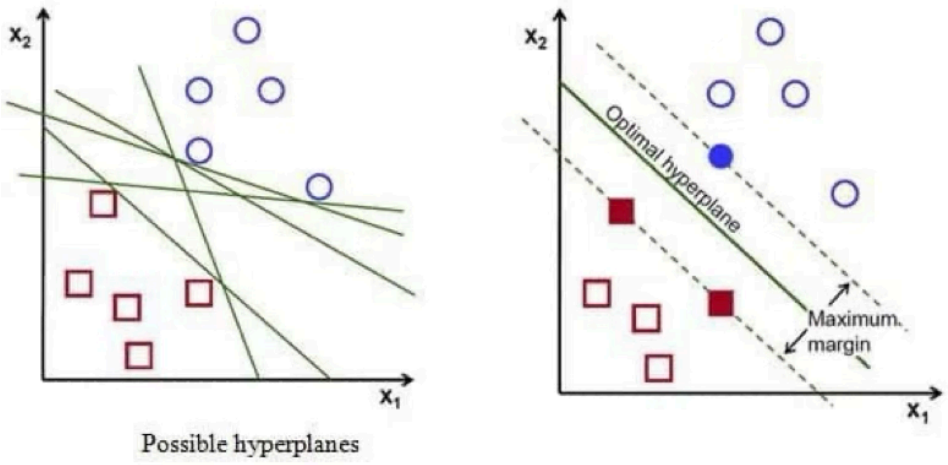
- w là vector trọng số (định hướng của siêu phẳng).
- b là bias (độ lệch của siêu phẳng). Bài toán tối ưu hóa được xây dựng sao cho:

Dữ liệu thuộc lớp +1 thỏa mãn: $w \cdot x_i - b \geq 1$

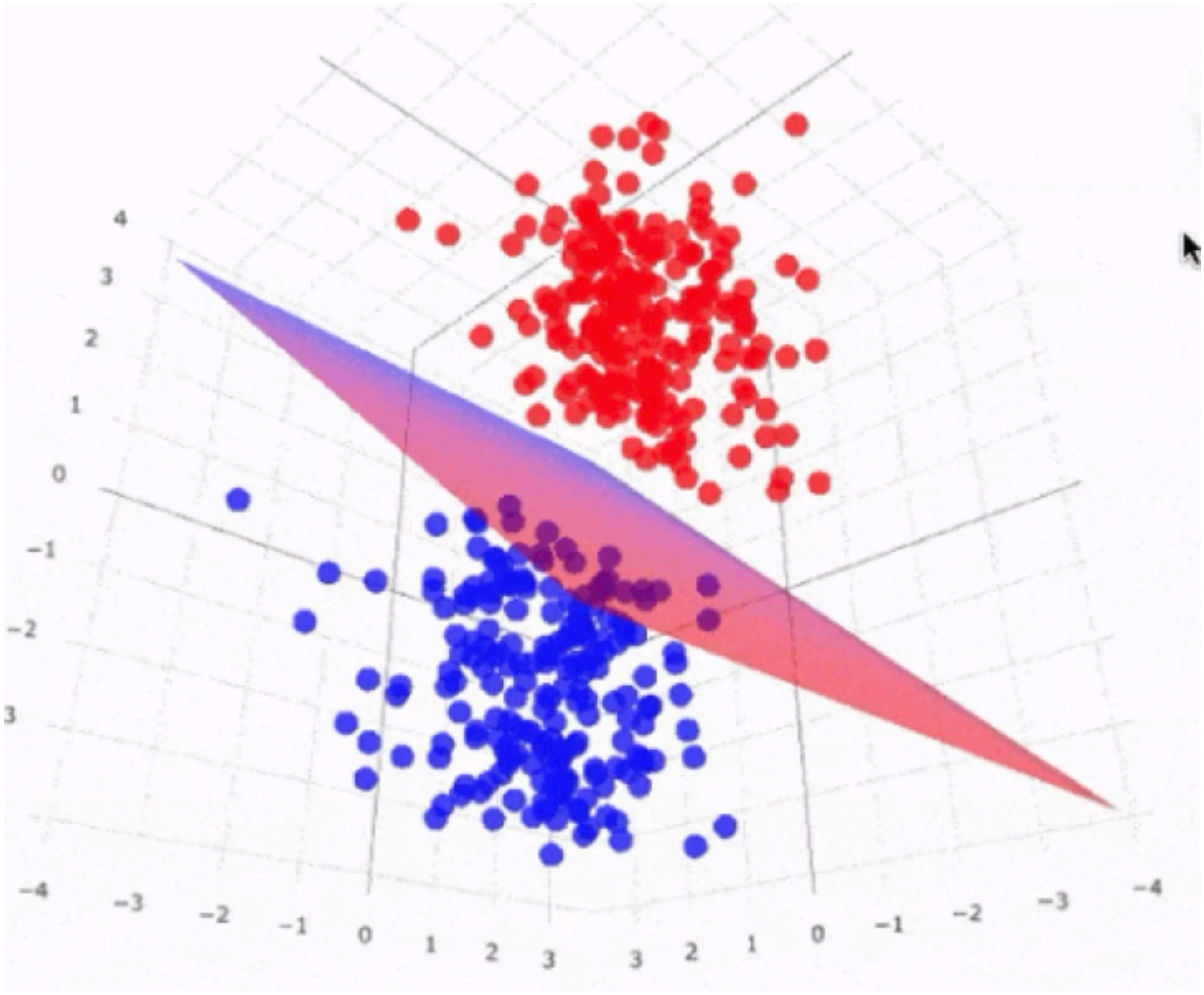
Dữ liệu thuộc lớp -1 thỏa mãn: $w \cdot x_i - b \leq -1$

Mục tiêu là tối thiểu hóa $| w |^2$, tức là làm cho siêu phẳng càng "mỏng" và margin càng lớn càng tốt.

Mục tiêu chính của SVM là tìm một siêu phẳng (hyperplane) tối ưu nhất để phân chia các lớp dữ liệu khác nhau trong không gian nhiều chiều. Các siêu phẳng này giúp tối đa hóa khoảng cách biên (maximal margin) giữa các lớp dữ liệu, tạo nên một mô hình mạnh mẽ và hiệu quả cho việc phân loại.

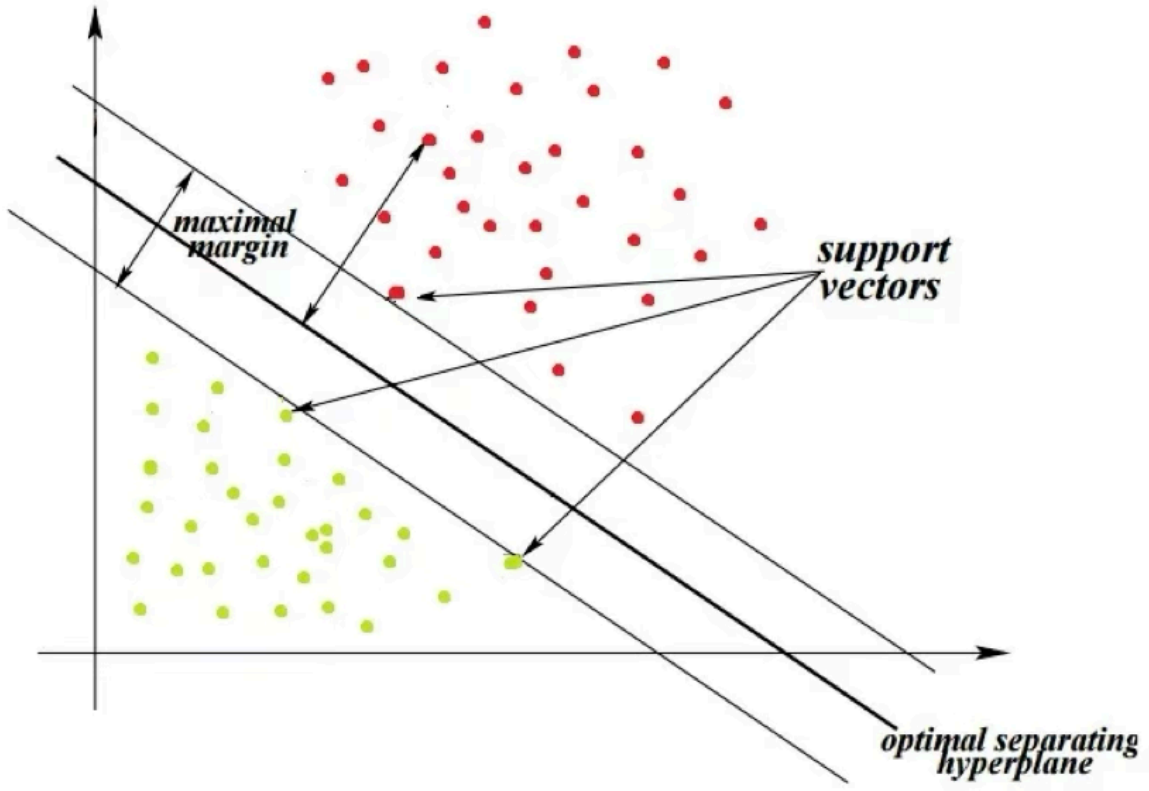


Hình 4: Siêu phẳng tối ưu có lề cực đại



Hình 5: Siêu mặt phẳng cực đại trong không gian 3D

Một điểm trong không gian véc tơ có thể được coi là một véc tơ từ gốc tọa độ tới điểm đó. Các điểm dữ liệu nằm trên hoặc gần nhất với siêu phẳng được gọi là véc tơ hỗ trợ, chúng ảnh hưởng đến vị trí và hướng của siêu phẳng. Các véc tơ này được sử dụng để tối ưu hóa lề và nếu xóa các điểm này, vị trí của siêu phẳng sẽ thay đổi. Một điểm lưu ý nữa đó là các véc tơ hỗ trợ phải cách đều siêu phẳng.



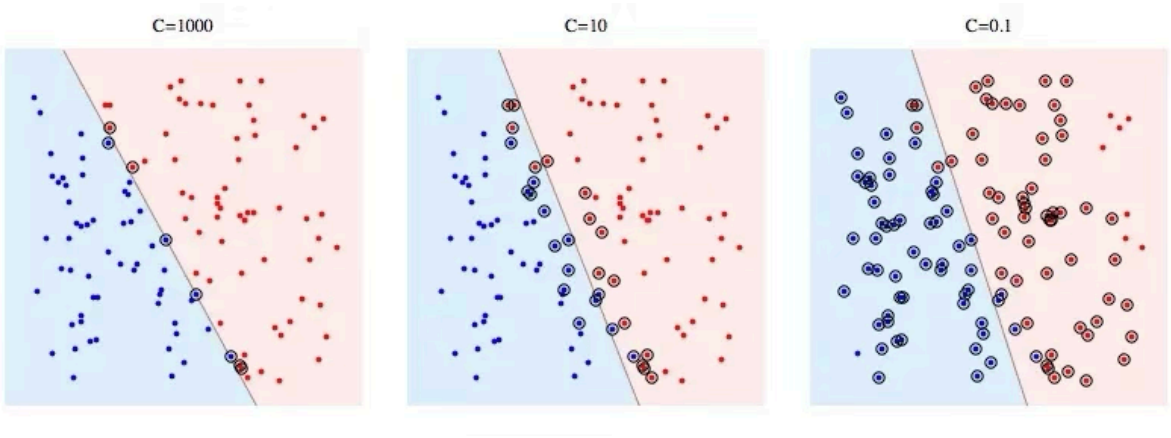
Hình 6: Siêu mặt phẳng cực đại trong không gian 2D

Khi dữ liệu không thể phân tách bằng một siêu phẳng duy nhất (ví dụ, trong trường hợp không tuyến tính), SVM sử dụng hai kỹ thuật:

Một là lề mềm (Soft margin): Thuật toán này cho phép SVM mắc một số lỗi nhất định và giữ cho lề càng rộng càng tốt để các điểm khác vẫn có thể được phân loại chính xác. Có hai kiểu phân loại sai có thể xảy ra:

- Dữ liệu nằm ở đúng bên nhưng phạm vào lề
- Dữ liệu nằm ở sai bên

Mức độ chấp nhận lỗi càng lớn có nghĩa là SVM càng bị phạt nặng khi thực hiện phân loại sai. Do đó, lề càng hẹp và càng ít vectơ hỗ trợ được sử dụng. Khi lập trình với sklearn, mức độ chấp nhận lỗi được coi như một tham số phạt (C).



Hình 7: Hình thể hiện SVM với các giá trị C khác nhau

Hai gọi là kernel trick. Kỹ thuật này cho phép chuyển đổi không gian dữ liệu về một không gian cao hơn, nơi mà dữ liệu có thể được phân tách một cách hiệu quả. Có nhiều loại hàm kernel, như linear, polynomial và radial basis function (RBF), mỗi loại có những ưu điểm riêng tùy thuộc vào tính chất của dữ liệu.

2.3.5. Mô hình Logistic Regression

Logistic Regression là một mô hình phân loại phổ biến trong học máy, thường được sử dụng để dự đoán khả năng một đối tượng thuộc vào một trong hai nhóm hoặc lớp. Mặc dù có tên gọi "hồi quy", Logistic Regression chủ yếu được sử dụng cho các bài toán phân loại.

Logistic Regression sử dụng một hàm sigmoid (hay logistic) để biến đổi đầu ra tuyến tính của mô hình thành một xác suất:

$p(y = 1|x) = \sigma(w^T \cdot x) = \frac{1}{1+\exp(-w^T x)}$

trong đó:

- w là vector trọng số.
- x là vector đặc trưng của mẫu.
- $\sigma(z)$ là hàm sigmoid, chuyển giá trị z thành một xác suất từ 0 đến 1.

2.3.6. Hàm mất mát

Hàm mất mát trong Logistic Regression là hàm log-loss, được định nghĩa như sau:

$L(w) = -\frac{1}{N}\sum_{i=1}^{N}[y_i \log p(y_i|x_i) + (1-y_i)\log(1-p(y_i|x_i))]$

trong đó:

- N là số lượng mẫu trong tập huấn luyện.
- y_i là nhãn thực tế của mẫu thứ i (có giá trị 0 hoặc 1).
- $p(y_i|x_i)$ là xác suất dự đoán mà mô hình ước lượng.

Để tối ưu hóa hàm mất mát L(w), ta sử dụng thuật toán Gradient Descent. Gradient Descent cập nhật trọng số w theo hướng ngược lại của gradient của hàm mất mát:

$w \leftarrow w - \eta\nabla L(w) w \leftarrow w - \eta\nabla L(w)$

trong đó:

- η là tốc độ học (learning rate).
- $\nabla L(w)$ là gradient của hàm mất mát $L(w)$.

Gradient $\eta\nabla L(w)$ được tính theo công thức:

$\eta L(w) = -\frac{1}{N}\sum_{i=1}^{N}(y_i - \sigma(w^T x_i))x_i$

Khi áp dụng công thức này, ta có thể cập nhật w như sau:

$w \leftarrow w + \eta\frac{1}{N}\sum_{i=1}^{N}(y_i - \sigma(w^T x_i))x_i$

Với $y_i - \sigma(w^T x_i)$ là sự chênh lệch giữa nhãn thực tế y_i và xác suất dự đoán $\sigma(w^T x_i)$. Giá trị này cho biết hướng và mức độ điều chỉnh cần thiết cho trọng số w.

Bằng cách điều chỉnh w theo gradient, mô hình dần dần học cách tối thiểu hóa hàm mất mát, dẫn đến việc cải thiện dự đoán.

2.3.7. Mạng nơ-ron hồi quy

Mạng nơ-ron hồi quy (Recurrent Neural Network, RNN) là một loại mạng nơ-ron chuyên biệt, được thiết kế để xử lý các dữ liệu dạng chuỗi, như chuỗi thời gian, ngôn ngữ tự nhiên, hoặc bất kỳ dạng dữ liệu nào có thứ tự liên tiếp. Điểm khác biệt chính của RNN so với các mạng nơ-ron truyền thống là khả năng ghi nhớ thông tin từ các bước thời gian trước đó, giúp mô hình hóa các mối quan hệ và phụ thuộc theo thời gian.

RNN có một chuỗi tính toán lặp lại qua từng bước thời gian, khiến mỗi đầu vào và đầu ra đều phụ thuộc vào ngữ cảnh của các bước trước. Các phép tính lặp này cho phép RNN ghi nhớ các thông tin ngắn hạn trong chuỗi. Tuy nhiên, RNN gặp vấn đề với "quên dần" khi phải ghi nhớ các chuỗi dài hơn do vấn đề lan truyền ngược qua thời gian, dẫn đến hiện tượng mất mát hoặc quá lớn của gradient.

RNN xử lý mỗi đầu vào theo trình tự, và trạng thái ẩn h_t ở mỗi bước thời gian t phụ thuộc vào đầu vào x_t và trạng thái ẩn của bước trước đó h_{t-1}. Công thức tính trạng thái ẩn được biểu diễn như sau:

$h_t = f(W_h \cdot h_{t-1} + W_x \cdot x_t + b)$

Trong đó:

- h_t: Trạng thái ẩn tại thời điểm t.
- x_t: Đầu vào hiện tại tại bước t.
- W_h: Ma trận trọng số áp dụng lên trạng thái ẩn trước đó.
- W_x: Ma trận trọng số của đầu vào hiện tại.
- b: Hệ số điều chỉnh (bias).
- f: Hàm kích hoạt (thường là hàm tanh hoặc ReLU) để tạo ra đầu ra phi tuyến.

RNN sử dụng thuật toán lan truyền ngược để cập nhật trọng số, nhưng trong quá trình này, các gradient có thể trở nên rất nhỏ (gradient tiêu hao) hoặc rất lớn (gradient bùng nổ), dẫn đến sai số lớn trong quá trình học. Điều này gây khó khăn cho RNN trong việc ghi nhớ các thông tin quan trọng từ các bước xa trong chuỗi.

Để khắc phục vấn đề này, các mô hình RNN nâng cao như LSTM (Long Short-Term Memory) và GRU (Gated Recurrent Unit) được phát triển, cho phép mô hình ghi nhớ các thông tin dài hạn tốt hơn bằng cách sử dụng các cổng điều khiển trạng thái.

2.3.8. Mô hình Long Short-Term Memory

Long Short-Term Memory (LSTM) là một dạng mạng nơ-ron hồi tiếp (Recurrent Neural Network - RNN) được thiết kế để xử lý và dự đoán dữ liệu chuỗi thời gian bằng cách giải quyết những hạn chế của RNN truyền thống, đặc biệt là trong việc lưu trữ và truy xuất thông tin dài hạn. Mô hình LSTM được phát triển nhằm giảm thiểu vấn đề về suy giảm đạo hàm (vanishing gradient problem) trong quá trình huấn luyện RNN với chuỗi dữ liệu dài.

Thuật toán LSTM hoạt động bằng cách sử dụng một cấu trúc ô nhớ (memory cell), nơi các trạng thái nhớ được kiểm soát qua các cổng: cổng quên, cổng nhập, và cổng đầu ra.

1. Các Cổng Chính trong LSTM

Cổng Quên (Forget Gate): Quyết định giữ lại bao nhiêu thông tin từ trạng thái trước đó. Công thức của cổng quên là:

- $f_t = \sigma\big(W_f \cdot [h_{t-1}, x_t] + b_f\big)$
    - W_f: Trọng số cho cổng quên.
    - h_{t-1}: Đầu ra từ trạng thái trước.
    - x_t: Dữ liệu hiện tại.
    - σ: Hàm sigmoid, giới hạn đầu ra trong khoảng $(0, 1)$, xác định tỷ lệ thông tin được giữ lại.

Cổng Nhập (Input Gate): Quyết định thông tin mới sẽ được thêm vào trạng thái ô nhớ. Cổng nhập có hai bước:

- Xác định giá trị cần cập nhật: $i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i)$
    - Tạo một vector ứng viên $\tilde{C}_t$ để thêm vào ô nhớ: $\tilde{C}_t = \tanh(W_C \cdot [h_{t-1}, x_t] + b_C)$

◦ Trạng thái ô nhớ mới: $C_t = f_t * C_{t-1} + i_t * \tilde{C}_t$

Cổng Đầu Ra (Output Gate): Quyết định phần nào của trạng thái ô nhớ sẽ được đưa ra. Công thức tính toán là:

- $o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o)$
- Sau đó, đầu ra cuối cùng: $h_t = o_t * \tanh(C_t)$

Mỗi bước trong chuỗi dữ liệu, LSTM sẽ thực hiện các bước sau:

Bước 1: Xác định thông tin nào cần bỏ qua từ trạng thái nhớ trước qua cổng quên. Bước 2: Thêm thông tin mới vào ô nhớ thông qua cổng nhập. Bước 3: Tính toán đầu ra dựa trên cổng đầu ra và trạng thái mới của ô nhớ.

2.3.9. Mô hình Bidirectional Encoder Representation from Transformer

BERT là viết tắt của cụm từ Bidirectional Encoder Representation from Transformer có nghĩa là mô hình biểu diễn từ theo 2 chiều ứng dụng kỹ thuật Transformer. BERT được thiết kế để huấn luyện trước các biểu diễn từ (pre-train word embedding). Điểm đặc biệt ở BERT đó là nó có thể điều hòa cân bằng bối cảnh theo cả 2 chiều trái và phải.

Cơ chế attention của Transformer sẽ truyền toàn bộ các từ trong câu văn đồng thời vào mô hình một lúc mà không cần quan tâm đến chiều của câu. Do đó Transformer được xem như là huấn luyện hai chiều (bidirectional) mặc dù trên thực tế chính xác hơn chúng ta có thể nói rằng đó là huấn luyện không chiều (non-directional). Đặc điểm này cho phép mô hình học được bối cảnh của từ dựa trên toàn bộ các từ xung quanh nó bao gồm cả từ bên trái và từ bên phải.

CHƯƠNG 3. URL-BASED VÀ SMS SYSTEMS

Hệ thống dựa trên URL và SMS đã trở thành một phần không thể thiếu trong cuộc sống kỹ thuật số hiện đại, được sử dụng rộng rãi cho nhiều mục đích, từ thương mại điện tử và tiếp thị đến giao tiếp cá nhân và dịch vụ công. Tuy nhiên, sự

phổ biến này cũng đi kèm với những rủi ro bảo mật, đặc biệt là sự gia tăng của các hoạt động lừa đảo. Phần này sẽ tập trung vào việc phân tích các hệ thống dựa trên URL và SMS, làm nổi bật các phương pháp phát hiện lừa đảo và các biện pháp bảo mật cần thiết.

3.1. Sự phát triển của các hệ thống dựa trên URL và SMS

Sự phát triển của internet và điện thoại di động đã thúc đẩy sự phổ biến của các hệ thống dựa trên URL và SMS. URL (Uniform Resource Locator) là một trong các tiêu chuẩn để định vị tài nguyên trên internet, trong khi SMS (Short Message Service) cho phép gửi tin nhắn văn bản ngắn giữa các thiết bị di động. Sự kết hợp của hai công nghệ này đã tạo ra một nền tảng mạnh mẽ cho nhiều ứng dụng, bao gồm:

- Thương mại điện tử: URL được sử dụng để dẫn người dùng đến các trang web mua sắm, trong khi SMS được sử dụng để gửi xác nhận đơn hàng, khuyến mãi và cập nhật giao hàng.
- Tiếp thị: URL trong tin nhắn SMS cho phép người dùng truy cập nhanh vào các trang web quảng cáo, chương trình khuyến mãi và thông tin sản phẩm.
- Dịch vụ tài chính: SMS được sử dụng để xác thực giao dịch, gửi thông báo số dư tài khoản và cảnh báo gian lận.
- Giao tiếp cá nhân: SMS vẫn là một phương thức giao tiếp phổ biến, cho phép người dùng gửi tin nhắn văn bản nhanh chóng và dễ dàng.
- Dịch vụ công: SMS được sử dụng để gửi thông báo khẩn cấp, thông tin y tế và các dịch vụ công cộng khác.

3.2. Rủi ro bảo mật và lừa đảo

Mặc dù mang lại nhiều lợi ích, các hệ thống dựa trên URL và SMS cũng tiềm ẩn nhiều rủi ro bảo mật. Sự gia tăng của các hoạt động lừa đảo là một mối quan tâm đáng kể. Các hình thức lừa đảo phổ biến bao gồm:

- Phishing: Tin nhắn SMS hoặc email chứa URL dẫn đến các trang web giả mạo, yêu cầu người dùng cung cấp thông tin cá nhân như tên đăng nhập, mật khẩu và thông tin tài khoản ngân hàng.
- Smishing: Tương tự như phishing, smishing sử dụng SMS để lừa đảo người dùng.
- Malware: URL độc hại có thể dẫn đến việc tải xuống phần mềm độc hại lên thiết bị của người dùng.
- Spam: Tin nhắn SMS không mong muốn, thường chứa quảng cáo hoặc nội dung lừa đảo.

CHƯƠNG 4. ĐỀ XUẤT THIẾT KẾ HỆ THỐNG

Bài này trình bày đề xuất thiết kế một hệ thống phát hiện lừa đảo trong tin nhắn SMS và URL, sử dụng các kỹ thuật học máy và xử lý ngôn ngữ tự nhiên. Hệ thống này nhằm mục đích bảo vệ người dùng khỏi các mối đe dọa trực tuyến bằng cách phân tích nội dung, ngữ cảnh và các đặc điểm khác của tin nhắn và URL để xác định các dấu hiệu lừa đảo.

4.1. Kiến trúc tổng thể

Hệ thống được thiết kế theo kiến trúc modular, bao gồm các thành phần chính sau:

- Mô-đun phân tích và trích xuất đặc trưng: Phân tích nội dung của SMS và các URL được tìm thấy trong tin nhắn SMS để phát hiện các đặc điểm đáng ngờ, chẳng hạn như tên miền đáng ngờ, URL rút gọn và các tham số bất thường. Từ đó xây dựng mô hình phù hợp cho việc phân loại.
- Mô-đun học máy và xử lý ngôn ngữ tự nhiên: Sử dụng các mô hình học máy có giám sát và các kỹ thuật NLP để phân tích nội dung tin nhắn SMS. Nhằm phân loại tin nhắn SMS và URL là hợp pháp hoặc lừa đảo. Mô-đun này được huấn luyện trên một tập dữ liệu lớn các tin nhắn và URL đã được gắn nhãn.
- Mô-đun báo cáo: Cung cấp giao diện để hiển thị kết quả phân tích và báo cáo các tin nhắn và URL đáng ngờ cho người dùng.

4.2. Quy trình thu thập và xử lý dữ liệu

4.2.1. Các phương pháp phân tích URL trong phát hiện lừa đảo

Phân tích URL là một phần quan trọng trong việc phát hiện lừa đảo, vì các URL thường được sử dụng để dẫn người dùng đến các trang web giả mạo hoặc độc hại. Các phương pháp phân tích URL bao gồm:

4.2.1.1. Phân tích cấu trúc URL

Phân tích cấu trúc URL tập trung vào việc kiểm tra các thành phần của URL để phát hiện các dấu hiệu lừa đảo. Các đặc điểm thường được xem xét bao gồm:

- Kiểm tra xem URL có sử dụng địa chỉ IP thay vì tên miền hay không.
- URL quá dài có thể là dấu hiệu của một URL độc hại.
- URL rút gọn có thể che giấu đích đến thực sự của liên kết.
- Kiểm tra xem URL có chứa ký tự @ hay không.
- Kiểm tra xem URL có chuyển hướng người dùng đến trang web khác hay không.
- Kiểm tra xem URL có chứa dấu gạch ngang (-) giữa tên miền hay không.
- Kiểm tra số lượng tên miền phụ trong URL.

4.2.1.2. Phân tích bảo mật URL

Phân tích bảo mật URL liên quan đến việc kiểm tra các yếu tố bảo mật của URL. Các đặc điểm thường được xem xét bao gồm:

- Kiểm tra xem URL có sử dụng HTTPS hay không.
- Kiểm tra xem URL có sử dụng port chuẩn hay không.
- Kiểm tra xem tên miền có khớp với chứng chỉ HTTPS hay không.

4.2.1.3. Phân tích nội dung trang web

Phân tích nội dung trang web liên quan đến việc kiểm tra nội dung của trang web mà URL dẫn đến. Các đặc điểm thường được xem xét bao gồm:

- Kiểm tra xem URL có yêu cầu tài nguyên từ các tên miền khác hay không.
- Kiểm tra các liên kết neo trong trang web.
- Kiểm tra các liên kết trong thẻ script.
- Kiểm tra xem biểu mẫu trên trang web có gửi dữ liệu đến tên miền khác hay không.
- Kiểm tra xem trang web có chứa địa chỉ email thông tin hay không.

4.2.1.4. Phân tích hành vi

Phân tích hành vi tập trung vào việc theo dõi và phân tích hành vi của người dùng khi tương tác với URL. Các đặc điểm thường được xem xét bao gồm:

- Kiểm tra xem trang web có chuyển hướng nhiều lần hay không.
- Kiểm tra xem thanh trạng thái có bị tùy chỉnh hay không.
- Kiểm tra xem trang web có vô hiệu hóa chuột phải hay không.
- Kiểm tra xem trang web có sử dụng cửa sổ popup hay không.
- Kiểm tra xem trang web có sử dụng iframe để chuyển hướng hay không.

4.2.1.5. Phân tích lịch sử và uy tín trang web

Phân tích lịch sử và uy tín trang web liên quan đến việc kiểm tra các yếu tố như tuổi đời và lưu lượng truy cập của trang web. Các đặc điểm thường được xem xét bao gồm:

- Kiểm tra tuổi đời của tên miền.
- Kiểm tra bản ghi DNS của tên miền.
- Kiểm tra lưu lượng truy cập của trang web.
- Kiểm tra xếp hạng trang của trang web.
- Kiểm tra xem trang web có được Google lập chỉ mục hay không.
- Kiểm tra số lượng liên kết trỏ đến trang web.
- Kiểm tra báo cáo thống kê của trang web.

4.2.1.6. Kết hợp các phương pháp phân tích

Việc kết hợp nhiều phương pháp phân tích URL có thể giúp cải thiện độ chính xác và hiệu quả của hệ thống phát hiện lừa đảo. Bằng cách sử dụng cả phân tích cấu trúc, bảo mật, nội dung, hành vi và lịch sử, hệ thống có thể đưa ra các đánh giá toàn diện và chính xác hơn về mức độ an toàn của URL.

4.2.2. Các phương pháp phân tích tin nhắn SMS

Phân tích tin nhắn SMS để phát hiện lừa đảo là một ứng dụng quan trọng của học máy và học sâu trong xử lý ngôn ngữ tự nhiên (NLP). Một số phương pháp phân tích tin nhắn SMS bao gồm:

4.2.3. Tiền xử lý văn bản

- Làm sạch dữ liệu: Loại bỏ các ký tự không cần thiết như dấu câu, ký tự đặc biệt, hoặc các từ dư thừa. Tin nhắn thường được chuyển thành chữ thường để đảm bảo tính đồng nhất.
- Tokenization: Phân tách văn bản thành các từ hoặc cụm từ nhỏ hơn (tokens) để phân tích.
- Loại bỏ từ dừng (Stop Words Removal): Loại bỏ các từ phổ biến không mang nhiều ý nghĩa như “là”, “của”, “và” trong các ngôn ngữ khác nhau.
- Stemming và Lemmatization: Rút gọn từ về gốc để giảm bớt sự khác biệt giữa các biến thể của từ.

4.2.4. Khai thác đặc trưng (Feature Extraction)

- Bag of Words (BoW): Chuyển đổi văn bản thành ma trận tần suất các từ xuất hiện, không quan tâm đến thứ tự của chúng.
- TF-IDF (Term Frequency-Inverse Document Frequency): Đánh trọng số từ dựa trên tần suất xuất hiện trong một tin nhắn so với toàn bộ tập dữ liệu.
- Word Embeddings: Sử dụng các phương pháp như Word2Vec, GloVe, hoặc FastText để chuyển đổi từ ngữ thành các vector không gian liên tục, biểu diễn ngữ nghĩa tốt hơn.

4.2.5. Phát hiện bất thường (Anomaly Detection)

Sử dụng các kỹ thuật học không giám sát (unsupervised learning) để phát hiện các mẫu bất thường trong tin nhắn, chẳng hạn như tần suất từ khóa, cấu trúc câu khác lạ, hoặc hành vi không phổ biến.

4.2.6. Kỹ thuật nâng cao

- Transfer Learning: Tận dụng các mô hình ngôn ngữ được huấn luyện trước, như BERT, để giảm thời gian huấn luyện và cải thiện độ chính xác.
- Attention Mechanisms: Áp dụng các cơ chế attention để giúp mô hình tập trung vào các phần quan trọng trong tin nhắn.

4.2.7. Một số thách thức trong việc phân tích

- Dữ liệu không cân bằng: Số lượng tin nhắn lừa đảo thường ít hơn so với tin nhắn thông thường, gây ra sự mất cân bằng dữ liệu. Các kỹ thuật như oversampling, undersampling hoặc sử dụng các thuật toán đặc biệt như XGBoost có thể giải quyết vấn đề này.
- Thay đổi hành vi lừa đảo: Các kẻ tấn công liên tục thay đổi chiến thuật, nên mô hình cần được cập nhật thường xuyên bằng cách huấn luyện lại với dữ liệu mới.

4.3. Đánh giá mô hình

4.3.1. Support Vector Machine (SVM)

- Hiệu quả với dữ liệu có số chiều cao, giúp phân biệt tốt giữa các lớp nếu dữ liệu phân tách được. Sử dụng tốt khi có một lượng dữ liệu nhỏ hoặc trung bình, thường đạt độ chính xác cao trong phân loại văn bản. Có thể sử dụng hạt nhân (kernel) để xử lý dữ liệu phi tuyến tính.

• Nhược điểm: Không hiệu quả khi làm việc với các tập dữ liệu lớn vì yêu cầu thời gian tính toán cao. Hiệu suất giảm nếu dữ liệu không phân tách tuyến tính tốt, và việc chọn kernel phù hợp có thể phức tạp.

• Thích hợp khi dùng với các phương pháp khai thác đặc trưng như TF-IDF hoặc BoW.

4.3.2. Naive Bayes

• Ưu điểm: Nhanh, dễ triển khai, và yêu cầu ít tài nguyên tính toán, lý tưởng cho các bài toán phân loại văn bản. Hiệu quả đặc biệt trong bài toán phân loại văn bản hoặc với dữ liệu có nhiều đặc trưng không liên quan. Hoạt động tốt với dữ liệu có sự phân phối xác suất rõ ràng.

• Nhược điểm: Giả định độc lập giữa các đặc trưng (giả định của Naive Bayes) có thể không đúng trong thực tế, làm giảm độ chính xác. Không hiệu quả khi xử lý các đặc trưng có mối quan hệ phức tạp.

• Các biến thể phổ biến như Multinomial Naive Bayes hoặc Bernoulli Naive Bayes được dùng nhiều cho phân loại văn bản.

4.3.3. Random Forest

• Ưu điểm: Khả năng tổng quát hóa tốt và hiệu quả với các tập dữ liệu có nhiều đặc trưng. Chống overfitting do sử dụng nhiều cây quyết định với kết quả trung bình. Có thể xử lý dữ liệu không cân bằng và cung cấp thông tin về tầm quan trọng của từng đặc trưng.

• Nhược điểm: Yêu cầu nhiều tài nguyên tính toán và bộ nhớ khi làm việc với các tập dữ liệu lớn. Khó diễn giải các quyết định của mô hình, vì Random Forest là một “hộp đen”.

• Thích hợp để thử nghiệm ban đầu vì có hiệu suất tốt mà không cần nhiều tinh chỉnh.

4.3.4. Logistic Regression

• Ưu điểm: Dễ hiểu, dễ triển khai, và diễn giải, với hiệu quả tốt trong bài toán phân loại nhị phân. Hoạt động tốt khi các đặc trưng có mối quan hệ tuyến tính với đầu ra. Yêu cầu ít tài nguyên tính toán, thích hợp với các bộ dữ liệu cỡ vừa và nhỏ.

• Nhược điểm: Không xử lý tốt dữ liệu phi tuyến tính hoặc có mối quan hệ phức tạp giữa các đặc trưng. Nhạy cảm với dữ liệu không cân bằng, cần sử dụng thêm các kỹ thuật như điều chỉnh trọng số hoặc oversampling.

• Có thể mở rộng để xử lý đa lớp bằng phương pháp One-vs-Rest.

4.3.5. Long Short-Term Memory (LSTM)

• Ưu điểm: Khả năng ghi nhớ thông tin dài hạn, giúp xử lý tốt các chuỗi văn bản dài và có ngữ cảnh phức tạp. Phù hợp để phát hiện các mẫu ngữ nghĩa trong dữ liệu tuần tự. Thích hợp cho dữ liệu mà ngữ cảnh trước đó có ảnh hưởng quan trọng đến quyết định.

• Nhược điểm: Yêu cầu nhiều tài nguyên tính toán và thời gian huấn luyện lâu. Cần một lượng lớn dữ liệu để hoạt động hiệu quả.

• LSTM thường được sử dụng trong các bài toán xử lý ngôn ngữ tự nhiên phức tạp hơn, như phân tích cảm xúc hoặc nhận diện thực thể.

4.3.6. Bidirectional Encoder Representations from Transformers (BERT)

• Ưu điểm: Mô hình mạnh mẽ với khả năng hiểu ngữ cảnh hai chiều, giúp nắm bắt ý nghĩa của từ trong văn bản tốt hơn so với các mô hình trước đó. Hiệu quả cao trong các bài toán xử lý ngôn ngữ tự nhiên phức tạp, có thể tinh chỉnh (fine-tune) cho các nhiệm vụ cụ thể như phân loại spam. Được tiền huấn luyện trên tập dữ liệu lớn, giúp giảm thiểu thời gian và tài nguyên cần thiết để huấn luyện từ đầu.

• Nhược điểm: Yêu cầu rất nhiều tài nguyên phần cứng, như GPU hoặc TPU, để huấn luyện và suy luận. Triển khai phức tạp và cần nhiều điều chỉnh để có kết quả tối ưu.

• Thích hợp cho các bài toán mà ngữ cảnh sâu của từ hoặc cụm từ cần được hiểu rõ.

4.3.7. Recurrent Neural Network (RNN)

• Ưu điểm: Phù hợp để xử lý dữ liệu tuần tự, như văn bản, với khả năng mô hình hóa ngữ cảnh ngắn hạn. Dễ dàng xử lý các chuỗi dữ liệu có độ dài khác nhau.

• Nhược điểm: RNN truyền thống có vấn đề với việc ghi nhớ thông tin dài hạn, do hiện tượng "vấn đề biến mất hoặc bùng nổ của gradient". Hiệu quả không cao so với các mô hình học sâu tiên tiến như LSTM hoặc BERT.

• Thường được thay thế bằng LSTM hoặc GRU (Gated Recurrent Unit) trong các bài toán yêu cầu ngữ cảnh dài hạn.

4.3.8. Đánh giá lựa chọn mô hình

• Mô hình đơn giản: Naive Bayes, Logistic Regression, và SVM là lựa chọn tốt nếu dữ liệu không quá phức tạp và muốn mô hình dễ hiểu.

• Mô hình trung cấp: Random Forest thích hợp khi cần một mô hình mạnh mẽ với khả năng xử lý dữ liệu phức tạp mà không phải mất nhiều thời gian tinh chỉnh.

• Mô hình nâng cao: LSTM và BERT là những lựa chọn tốt khi ngữ cảnh của văn bản đóng vai trò quan trọng, và có đủ tài liệu và tài nguyên tính toán.

4.4. Thiết kế giao diện người dùng

Giao diện người dùng được thiết kế đơn giản và dễ sử dụng, cho phép người dùng dễ dàng tương tác với hệ thống:

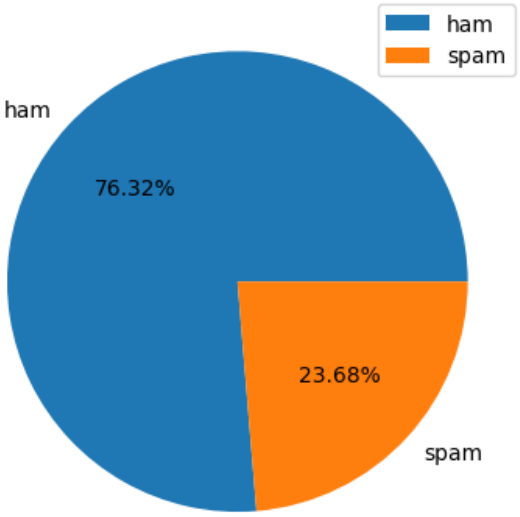
- Hiển thị form nhập văn bản để người dùng nhập tin nhắn SMS hoặc URL cần phân tích.
- Hiển thị kết quả phân tích, bao gồm xác định tin nhắn hoặc URL là hợp pháp hoặc lừa đảo, và các thông tin chi tiết về các đặc điểm đáng ngờ.

CHƯƠNG 5. HUẤN LUYỆN VÀ TRIỂN KHAI MÔ HÌNH PHÂN LOẠI

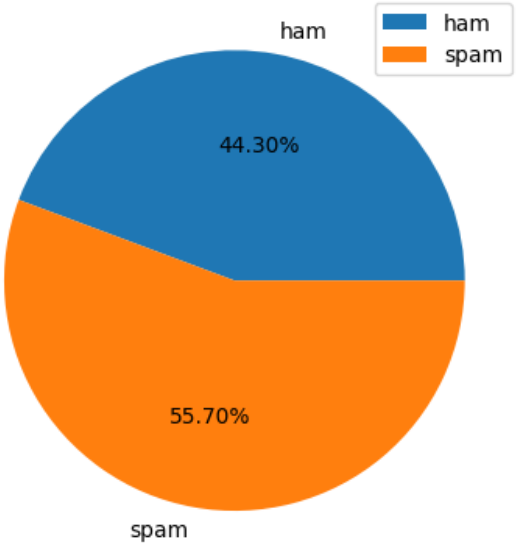
5.1. Thu thập và xử lý dữ liệu huấn luyện

Dữ liệu trong bài viết này tôi sử dụng nguồn từ Kaggle và Corpus. Bao gồm:

- 425 tin nhắn SMS rác đã được trích xuất thủ công từ Grumbletext. [1]
- 3,375 tin nhắn SMS hợp lệ được chọn ngẫu nhiên từ NUS SMS Corpus (NSC), là một tập dữ liệu gồm khoảng 10.000 tin nhắn hợp pháp được thu thập để nghiên cứu tại Khoa Khoa học Máy tính của Đại học Quốc gia Singapore. Phần lớn các tin nhắn này đến từ người Singapore, chủ yếu là sinh viên của trường. [2]
- 450 tin nhắn SMS hợp lệ được thu thập từ Luận án Tiến sĩ của Caroline Tag. [3]
- 1.002 tin nhắn SMS hợp lệ và 322 tin nhắn rác của tập dữ liệu SMS Spam Corpus v.0.1 Big. [4]
- Tập dữ liệu URL của hơn 11.000 trang web. Mỗi mẫu có 30 tham số và một tham số xác định trang web đó có phải là lừa đảo / spam hay không. [5]



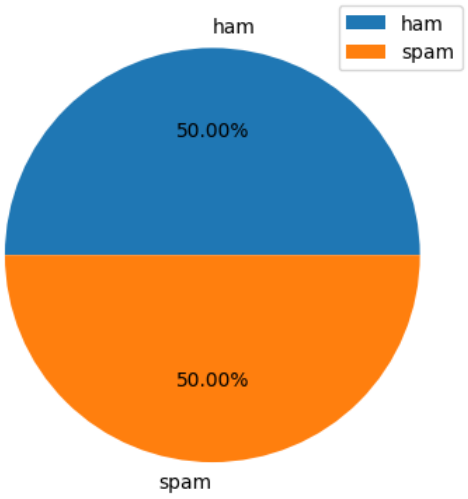
Hình 8: Phân phối tin nhắn SMS hợp pháp (ham) và rác (spam)



Hình 9: Phân phối URL hợp pháp (ham) và lừa đảo / spam (spam)

5.1.1. Tiền xử lý dữ liệu và trích xuất đặc trưng

Do số lượng mẫu trong mỗi lớp của dữ liệu SMS bị chênh lệch khá lớn (23.68% ham / 76.32% spam) nên tôi sử dụng phương pháp undersampling để cân bằng dữ liệu. Phương pháp này thực hiện bằng cách chọn ngẫu nhiên một số mẫu từ lớp chiếm ưu thế sao cho số lượng mẫu của hai lớp trở nên cân bằng hoặc gần cân bằng. Ngoài ra, ta cũng có thể sử dụng kỹ thuật tăng cường dữ liệu văn bản (data augmentation) để tạo ra thêm các mẫu huấn luyện.

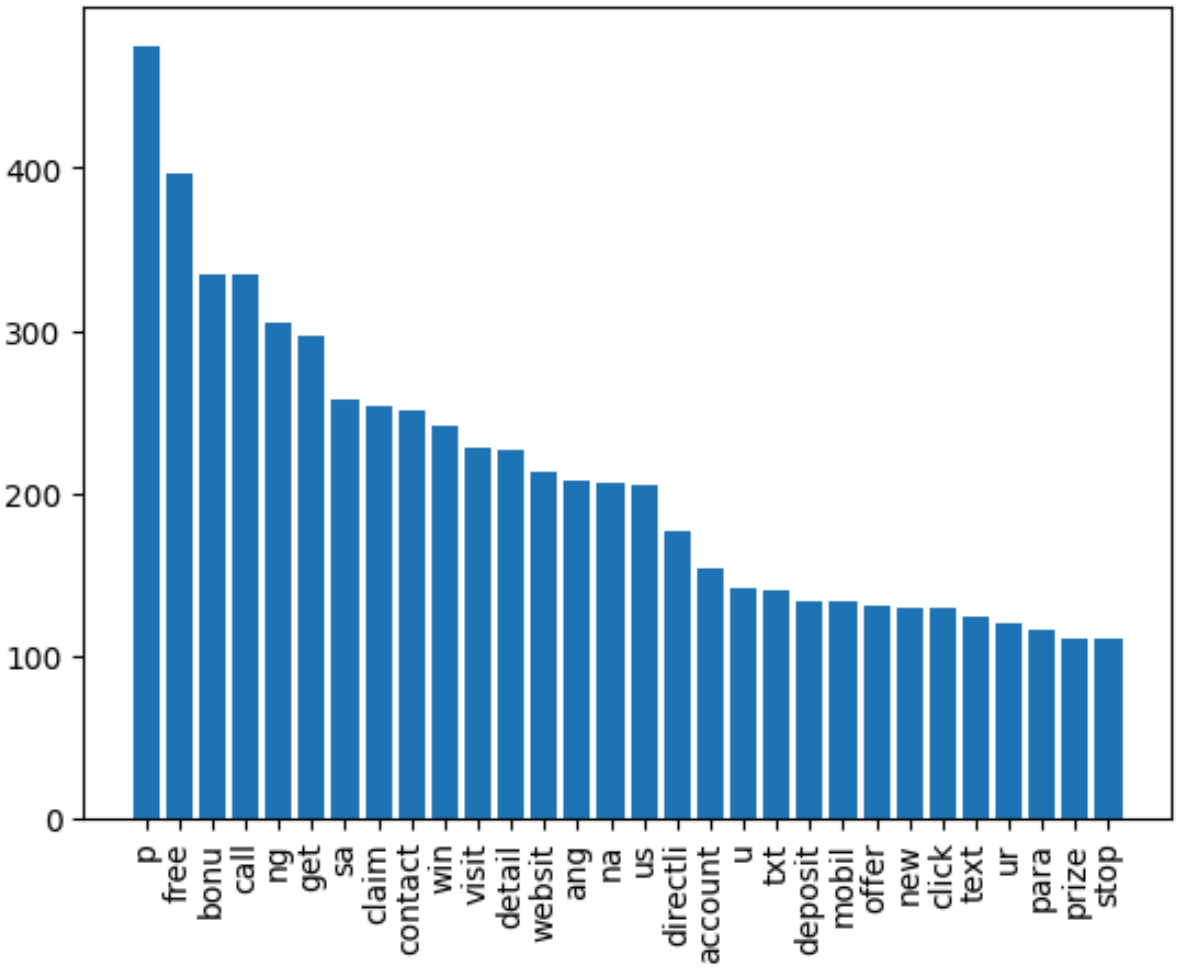


Hình 10: Phân phối tin nhắn SMS hợp pháp (ham) và rác (spam) sau khi cân bằng

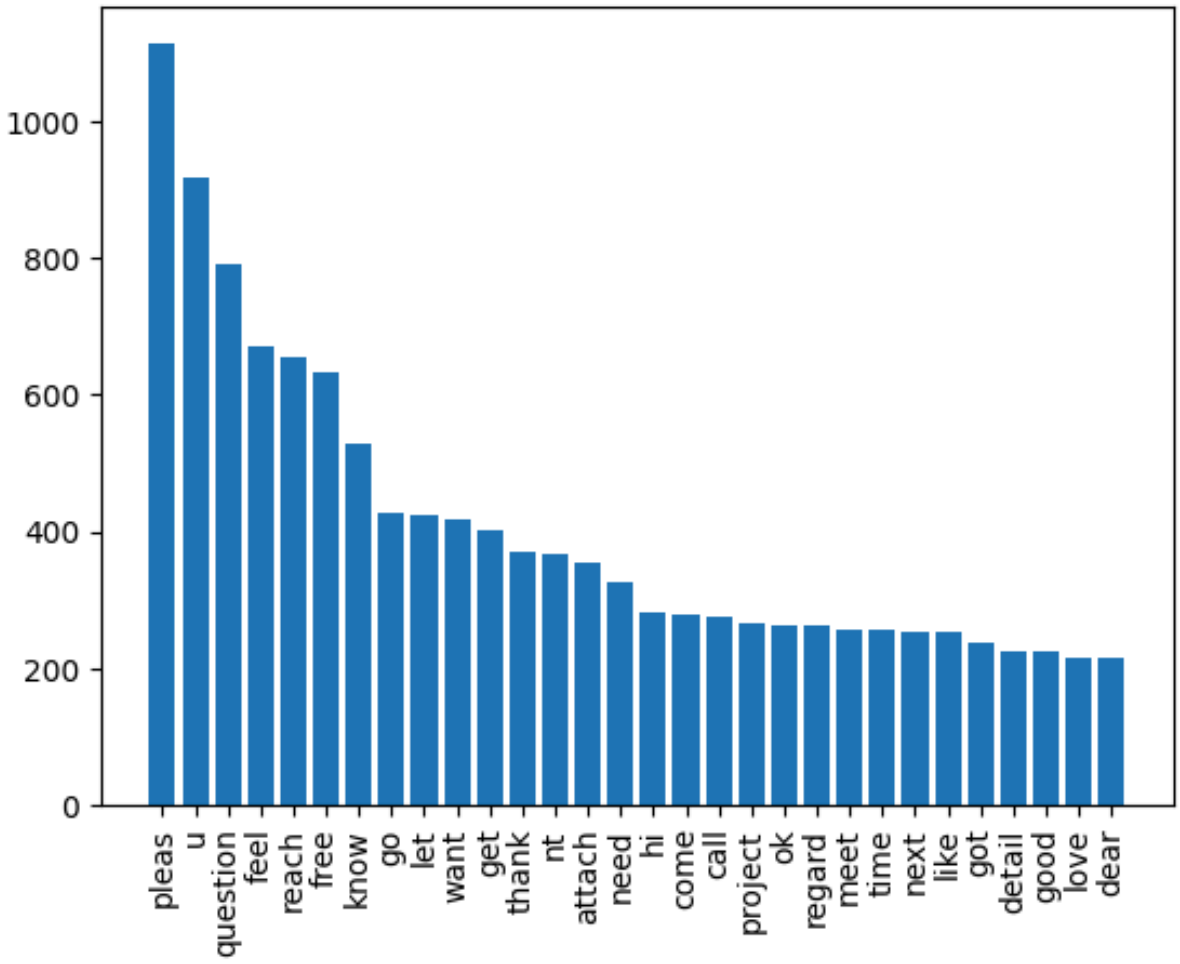
Dữ liệu SMS sau đó được tiền xử lý để loại bỏ các thông tin nhiễu khác như stopwords, các ký tự đặc biệt, dấu câu không cần thiết, chuyển đổi văn bản thành chữ thường,…

	content	clean_content
0	Sorry chikku, my cell got some problem thts y ...	sorri chikku cell got problem tht nt abl repli...
1	Yes ammae....life takes lot of turns you can o...	ye amma life take lot turn sit tri hold steer
2	Maglaro sa pinakamalaking platform at makuha a...	maglaro sa pinakamalak platform makuha ang p n...
3	I'm used to it. I just hope my agents don't dr...	use hope agent nt drop sinc book thing year wh...
4	Have you seen who's back at Holby?!	seen back holbi

Hình 11: Dữ liệu SMS trước và sau khi được làm sạch

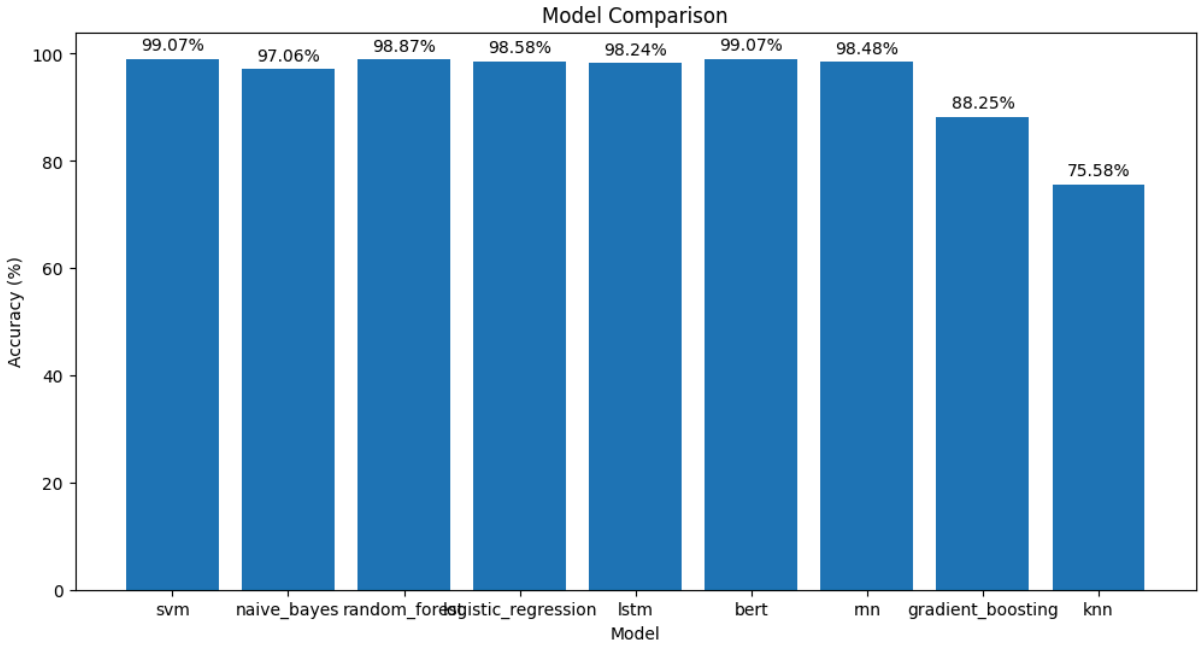


Hình 12: Các keyword phổ biến trong tin nhắn SMS rác



Hình 13: Các keyword phổ biến trong tin nhắn SMS hợp lệ

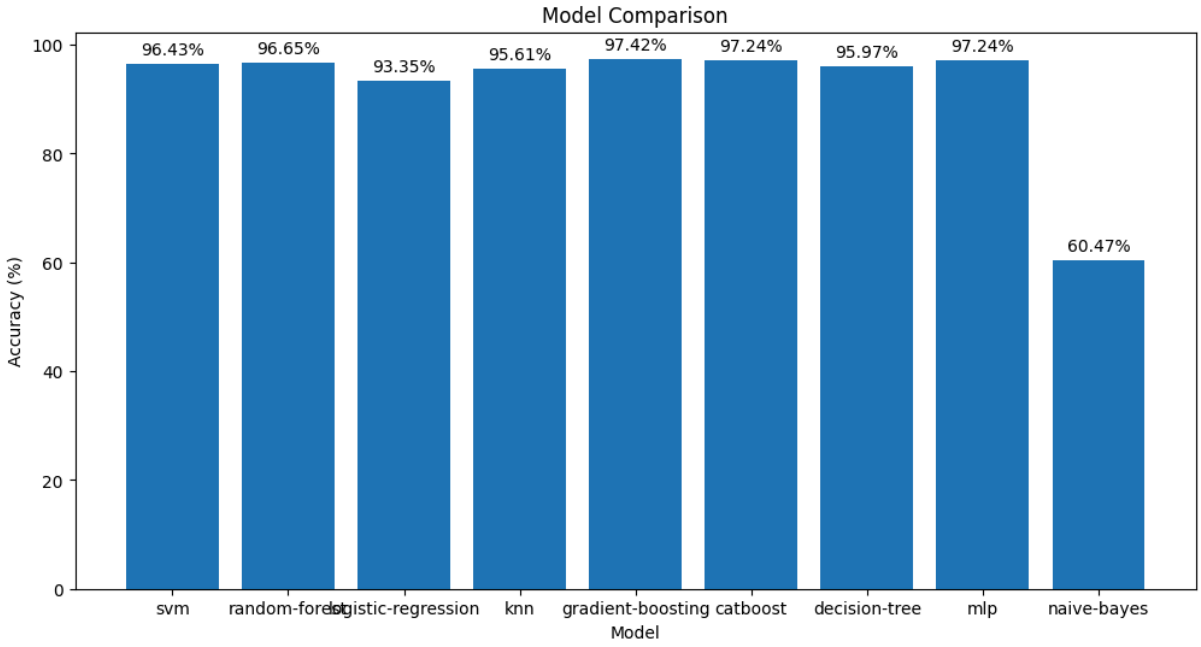
5.2. Huấn luyện mô hình



Hình 14: So sánh hiệu suất của các mô hình học máy trên dữ liệu SMS

	Model	Accuracy (%)	F1 Score (%)	Recall (%)	Precision (%)
0	svm	99.069995	99.069960	99.069995	99.077950
1	naive_bayes	97.063142	97.063134	97.063142	97.063870
2	random_forest	98.874205	98.874068	98.874205	98.899007
3	logistic_regression	98.580519	98.580403	98.580519	98.597344
4	lstm	98.237885	98.237883	98.237885	98.238069
5	bert	99.069995	99.069975	99.069995	99.073801
6	rnn	98.482624	98.482591	98.482624	98.486381
7	gradient_boosting	88.252570	88.091648	88.252570	90.449179
8	knn	75.575135	74.029065	75.575135	83.593561

Hình 15: Kết quả chi tiết huấn luyện mô hình phân loại tin nhắn SMS



Hình 16: So sánh hiệu suất của các mô hình học máy trên dữ liệu URL

	Model	Accuracy (%)	F1 Score (%)	Recall (%)	Precision (%)
0	svm	96.426956	96.420746	96.426956	96.448365
1	random-forest	96.653098	96.651614	96.653098	96.652698
2	logistic-regression	93.351425	93.338984	93.351425	93.363938
3	knn	95.612845	95.612129	95.612845	95.611771
4	gradient-boosting	97.421981	97.417500	97.421981	97.447211
5	catboost	97.241067	97.238377	97.241067	97.247380
6	decision-tree	95.974672	95.974455	95.974672	95.974278
7	mlp	97.241067	97.236993	97.241067	97.258254
8	naive-bayes	60.470375	55.816592	60.470375	78.842023

Hình 17: Kết quả chi tiết huấn luyện mô hình phân loại URL

5.2.1. Đánh giá hiệu quả của các mô hình

- SVM: Hiệu suất rất cao với cả Accuracy (99.07%), F1 Score (99.07%), Recall (99.07%), và Precision (99.08%). Điều này cho thấy mô hình SVM hoạt động rất tốt trong phân loại SMS và đạt sự cân bằng giữa Recall và Precision.

• Naive Bayes: Cũng có hiệu suất cao, với Accuracy, F1 Score, Recall, và Precision đều xấp xỉ 97.06%. Naive Bayes là một mô hình đơn giản nhưng hoạt động rất tốt với dữ liệu văn bản, vì vậy kết quả này là phù hợp.

• Random Forest: Cũng có kết quả tốt với Accuracy 98.87% và F1 Score 98.87%. Mô hình này có thể hơi phức tạp hơn nhưng cung cấp hiệu suất mạnh mẽ và khả năng tổng quát hóa cao.

• Logistic Regression: Có Accuracy và F1 Score khoảng 98.58%, đây là một mô hình tuyến tính hiệu quả và dễ diễn giải, nhưng hiệu suất thấp hơn một chút so với SVM và Random Forest.

• LSTM: Đạt Accuracy và F1 Score là 98.24%, cho thấy LSTM hoạt động tốt khi xử lý chuỗi dữ liệu văn bản. Tuy nhiên, hiệu suất này không cao hơn nhiều so với các mô hình truyền thống như SVM hoặc Random Forest.

• BERT: Hiệu suất của BERT (99.07%) rất cao, tương đương với SVM, cho thấy mô hình này nắm bắt tốt ngữ cảnh trong dữ liệu văn bản. Tuy nhiên, việc sử dụng BERT yêu cầu nhiều tài nguyên tính toán.

• RNN: Hiệu suất của RNN là 98.48%, thấp hơn so với LSTM và BERT, điều này là do RNN gặp vấn đề khi xử lý các chuỗi dài và mất ngữ cảnh trong một số trường hợp.

• Gradient Boosting: Hiệu suất thấp hơn đáng kể, với Accuracy 88.25% và F1 Score 88.09%. Đây có thể là do mô hình không phù hợp lắm với kiểu dữ liệu văn bản hoặc chưa được tối ưu đúng cách.

• KNN: Kết quả thấp nhất, với Accuracy 75.58% và F1 Score 74.03%, cho thấy mô hình KNN không phù hợp với bài toán này, đặc biệt khi dữ liệu văn bản có không gian đặc trưng cao.

5.2.2. Đề xuất cải tiến mô hình

Với SVM, ta có thể thử điều chỉnh các siêu tham số như loại kernel (Radial Basis Function, Polynomial) hoặc hệ số phạt (C) để tối ưu hóa thêm. Thực tế, sau khi sử dụng mô hình GridSearchCV để tinh chỉnh siêu tham số nhằm tìm ra tham số tối ưu, mô hình SVM có thể đạt hiệu suất cao hơn so với trước khi tinh chỉnh:

```
Fitting 5 folds for each of 30 candidates, totalling 150 fits
[CV] END .................C=0.01, gamma=scale, kernel=linear; total time=  19.5s
[CV] END .................C=0.01, gamma=scale, kernel=linear; total time=  19.3s
[CV] END .................C=0.01, gamma=scale, kernel=linear; total time=  20.1s
[CV] END .................C=0.01, gamma=scale, kernel=linear; total time=  18.8s
[CV] END .................C=0.01, gamma=scale, kernel=linear; total time=  19.4s
[CV] END .....................C=0.01, gamma=scale, kernel=rbf; total time=  21.5s
[CV] END .....................C=0.01, gamma=scale, kernel=rbf; total time=  20.7s
[CV] END .....................C=0.01, gamma=scale, kernel=rbf; total time=  20.4s
[CV] END .....................C=0.01, gamma=scale, kernel=rbf; total time=  20.3s
[CV] END .....................C=0.01, gamma=scale, kernel=rbf; total time=  20.5s
[CV] END ...................C=0.01, gamma=scale, kernel=poly; total time=  20.8s
[CV] END ...................C=0.01, gamma=scale, kernel=poly; total time=  21.6s
[CV] END ...................C=0.01, gamma=scale, kernel=poly; total time=  18.9s
[CV] END ...................C=0.01, gamma=scale, kernel=poly; total time=  18.8s
[CV] END ...................C=0.01, gamma=scale, kernel=poly; total time=  18.7s
[CV] END ..................C=0.01, gamma=auto, kernel=linear; total time=  18.7s
[CV] END ..................C=0.01, gamma=auto, kernel=linear; total time=  19.4s
[CV] END ..................C=0.01, gamma=auto, kernel=linear; total time=  18.6s
[CV] END ..................C=0.01, gamma=auto, kernel=linear; total time=  18.4s
[CV] END ..................C=0.01, gamma=auto, kernel=linear; total time=  19.2s
[CV] END ......................C=0.01, gamma=auto, kernel=rbf; total time=  20.8s
[CV] END ......................C=0.01, gamma=auto, kernel=rbf; total time=  20.6s
[CV] END ......................C=0.01, gamma=auto, kernel=rbf; total time=  20.7s
[CV] END ......................C=0.01, gamma=auto, kernel=rbf; total time=  21.2s
...
[CV] END ....................C=100, gamma=auto, kernel=poly; total time=  19.2s
[CV] END ....................C=100, gamma=auto, kernel=poly; total time=  19.4s
[CV] END ....................C=100, gamma=auto, kernel=poly; total time=  19.4s
Best parameters found:  {'C': 10, 'gamma': 'scale', 'kernel': 'rbf'}
```

Hình 18: Tinh chỉnh siêu tham số của mô hình SVM với GridSearchCV

```
Evaluating svm...
Accuracy: 0.9951052373959863
F1 Score: 0.9951052233232975
Recall: 0.9951052373959863
Precision: 0.9951128385971205
```

	Model	Accuracy (%)	F1 Score (%)	Recall (%)	Precision (%)
0	svm	99.510524	99.510522	99.510524	99.511284

Hình 19: Hiệu suất mô hình SVM tăng mạnh sau khi tinh chỉnh siêu tham số

Với BERT, ta có thể thử điều chỉnh quá trình fine-tuning hoặc dùng phiên bản nhỏ hơn của BERT như DistilBERT để giảm thời gian huấn luyện.

Cải tiến kỹ thuật tiền xử lý:

- Mô hình cần cải thiện tiền xử lý văn bản, chẳng hạn như sử dụng stemming, lemmatization, hoặc loại bỏ các từ dư thừa để giúp mô hình phân biệt rõ hơn giữa spam và ham.
- Cần thử nghiệm với các biểu diễn văn bản khác, như sử dụng từ nhúng (word embeddings) như Word2Vec hoặc GloVe.

Ta cũng có thể kết hợp các mô hình khác nhau (như SVM và Naive Bayes) để tận dụng ưu điểm của từng mô hình và cải thiện độ chính xác.

Ngoài ra, do hạn chế của nguồn dữ liệu và số lượng mẫu, việc tăng cường dữ liệu bằng cách thu thập thêm dữ liệu hoặc tạo dữ liệu tổng hợp có thể giúp cải thiện hiệu suất của mô hình.

5.2.3. Kết luận

- Hiệu suất hàng đầu: SVM và BERT có hiệu suất cao nhất. Tuy nhiên, SVM đơn giản hơn, có kết quả tốt hơn và yêu cầu ít tài nguyên hơn so với BERT. Do đó tôi chọn SVM là mô hình chính cho hệ thống phân loại tin nhắn SMS và URL.
- Mô hình học sâu: LSTM và RNN hoạt động tốt nhưng không vượt trội so với các mô hình truyền thống như SVM hoặc Random Forest. Điều này có thể là do kích thước tập dữ liệu chưa đủ lớn để phát huy toàn bộ tiềm năng của các mô hình học sâu.

5.3. Xây dựng chương trình phân loại

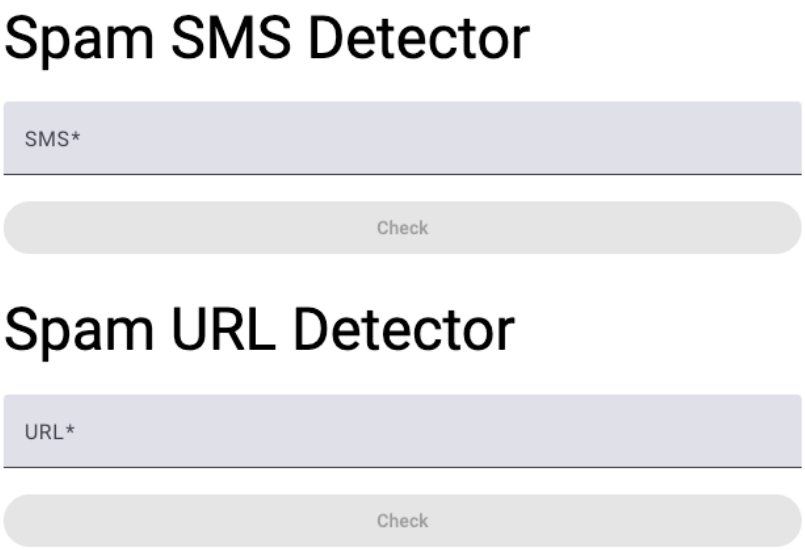
Sử dụng thư viện scikit-learn để xây dựng mô hình học máy SVM:

```
class SMSMLClassifier(SMSClassifier):
  def train(self, X: np.ndarray, Y: np.ndarray) -> None:
    # Nếu vectorizer chưa được khởi tạo, khởi tạo nó bằng TfidfVectorizerFactory
    if self.vectorizer is None:
      self.vectorizer = TfidfVectorizerFactory().vectorizer
```

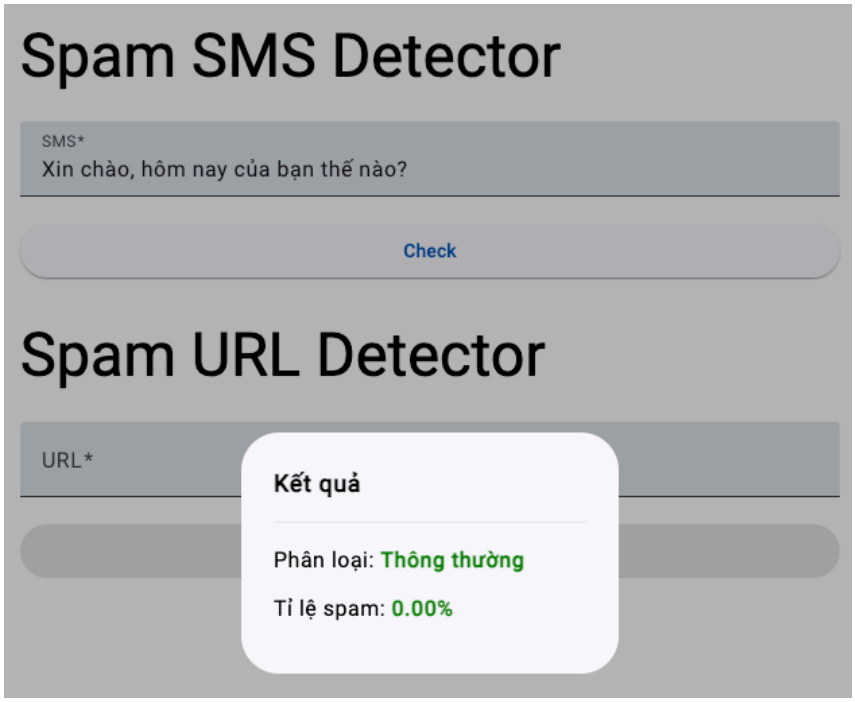
```
    # Huấn luyện vectorizer trên dữ liệu X
    self.vectorizer.fit(X.copy())
    # Chuyển đổi dữ liệu X thành dạng TF-IDF
    X_tfidf_transformed = self.vectorizer.transform(X.copy())

    # Huấn luyện mô hình trên dữ liệu đã được chuyển đổi
    self.model.fit(X_tfidf_transformed, Y)

  def predict(self, X: np.ndarray) -> np.ndarray[int]:
    # Chuyển đổi dữ liệu X thành dạng TF-IDF
    X_tfidf_transformed = self.vectorizer.transform(X.copy())
    # Dự đoán nhãn cho dữ liệu đã được chuyển đổi
    return self.model.predict(X_tfidf_transformed)

class SMSMLSVMClassifier(SMSMLClassifier):
    def __init__(self, model_dir):
        super().__init__("svm", model_dir)
        self.model = SVC(kernel="rbf", C=10, gamma="scale", random_state=42,
probability=True)
```

5.4. Phát triển giao diện người dùng

Giao diện người dùng được phát triển bằng framework Angular và FastAPI cho phép người dùng tương tác với hệ thống, xem kết quả phân tích, và báo cáo tin nhắn đáng ngờ.

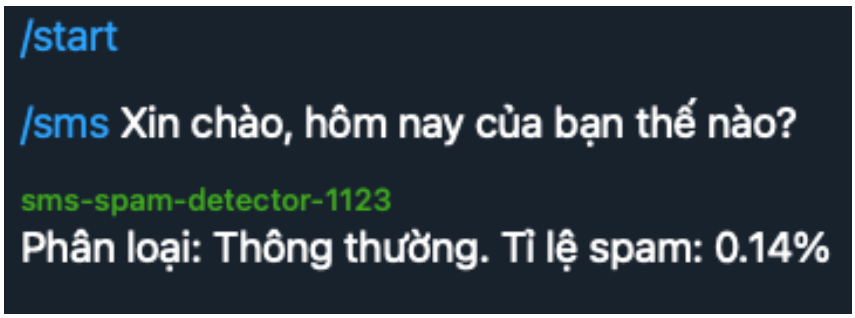


Hình 20: Giao diện người dùng của ứng dụng



Hình 21: Thông báo kết quả phân loại tin nhắn SMS

Ngoài ra, tôi còn phát triển thêm chatbot trên nền tảng Telegram, cho phép người dùng gửi tin nhắn và nhận kết quả phân loại trực tiếp từ bot:



Hình 22: Chatbot phân loại tin nhắn SMS trên Telegram

CHƯƠNG 6. KẾT LUẬN

6.1. Các thành tựu và những điểm nổi bật

Trong quá trình thực hiện dự án này, tôi đã đạt được nhiều thành tựu quan trọng. Hệ thống phát hiện lừa đảo trong tin nhắn SMS và URL đã được phát triển và triển khai thành công, sử dụng các kỹ thuật học máy và xử lý ngôn ngữ tự nhiên tiên tiến. Các điểm nổi bật của hệ thống bao gồm:

- Hiệu suất cao: Hệ thống đạt được độ chính xác cao trong việc phân loại tin nhắn SMS và URL là hợp pháp hoặc lừa đảo, với mô hình SVM và BERT đạt hiệu suất hàng đầu.
- Khả năng mở rộng: Hệ thống được thiết kế theo kiến trúc modular, cho phép dễ dàng mở rộng và tích hợp các mô hình và thuật toán mới trong tương lai.
- Giao diện người dùng thân thiện: Giao diện người dùng được phát triển đơn giản và dễ sử dụng, cho phép người dùng dễ dàng tương tác với hệ thống và xem kết quả phân tích.

6.2. Thách thức và hướng phát triển

Việc phát hiện lừa đảo trong tin nhắn SMS vẫn còn nhiều thách thức, bao gồm:

- Sự tinh vi của các kỹ thuật lừa đảo: Những kẻ lừa đảo liên tục phát triển các kỹ thuật mới để vượt qua các biện pháp bảo mật, đòi hỏi hệ thống phải được cập nhật thường xuyên.
- Sự đa dạng của ngôn ngữ và nội dung: Tin nhắn SMS có thể được viết bằng nhiều ngôn ngữ khác nhau và có nhiều nội dung khác nhau, khiến việc phân tích trở nên khó khăn và phức tạp.
- Bảo vệ quyền riêng tư của người dùng: Việc phân tích tin nhắn SMS cần phải được thực hiện theo cách bảo vệ quyền riêng tư của người dùng, đảm bảo rằng dữ liệu cá nhân không bị lạm dụng.

Các hướng phát triển trong tương lai bao gồm:

- Phát triển các thuật toán học máy mạnh mẽ hơn để phát hiện lừa đảo, bao gồm việc sử dụng các mô hình học sâu tiên tiến và các kỹ thuật học không giám sát.
- Sử dụng phân tích hành vi nâng cao để xác định các mẫu hoạt động đáng ngờ, giúp cải thiện độ chính xác và hiệu quả của hệ thống.
- Hợp tác giữa các nhà cung cấp dịch vụ, các cơ quan chính phủ và các nhà nghiên cứu để chia sẻ thông tin và phát triển các biện pháp bảo mật hiệu quả hơn, tạo ra một môi trường an toàn hơn cho người dùng.

6.3. Đề xuất các phương pháp nâng cao độ chính xác của mô hình

Để nâng cao độ chính xác của mô hình, tôi đề xuất nghiên cứu và áp dụng các kỹ thuật học máy tiên tiến, bao gồm:

- Sử dụng các mô hình học sâu như Transformer và các biến thể của nó để cải thiện khả năng hiểu ngữ cảnh và ngữ nghĩa của tin nhắn.
- Áp dụng các kỹ thuật tăng cường dữ liệu để tạo ra thêm các mẫu huấn luyện, giúp mô hình học tốt hơn từ dữ liệu đa dạng.

6.4. Ứng dụng cho các nền tảng khác

Ngoài việc phát hiện lừa đảo trong tin nhắn SMS và URL, hệ thống có thể được mở rộng để ứng dụng cho các nền tảng khác như email, mạng xã hội và các ứng dụng nhắn tin khác. Điều này sẽ giúp bảo vệ người dùng trên nhiều nền tảng khác nhau, đảm bảo an toàn và bảo mật thông tin cá nhân.

Việc mở rộng ứng dụng cho các nền tảng khác cũng đòi hỏi nghiên cứu và phát triển thêm các kỹ thuật phân tích và mô hình học máy phù hợp với từng nền tảng, đảm bảo hiệu suất và độ chính xác cao trong việc phát hiện lừa đảo.

TÀI LIỆU THAM KHẢO

[1] “Grumbletext.” [Online]. Available: https://www.grumbletext.co.uk/

[2] “National University of Singapore.” [Online]. Available: https://www.comp.nus.edu.sg/

[3] “A Corpus Linguistics Study of SMS Text Messaging.” [Online]. Available: https://etheses.bham.ac.uk/id/eprint/253/1/Tagg09PhD.pdf

[4] “SMS Spam Collection Dataset.” [Online]. Available: https://www.kaggle.com/datasets/uciml/sms-spam-collection-dataset

[5] “Phishing Website Detector.” [Online]. Available: https://www.kaggle.com/datasets/eswarchandt/phishing-website-detector