ĐẠI HỌC ĐÀ NẴNG

TRƯỜNG ĐẠI HỌC BÁCH KHOA ĐÀ NẴNG

KHOA ĐIỆN TỬ VIỄN THÔNG

BÁO CÁO CUỐI KÌ

ỨNG DỤNG THUẬT TOÁN HỌC MÁY TRONG PHÂN LOẠI EMAIL SPAM VÀ TIN NHẮN LỪA ĐẢO

MÔN HỌC: CHUYÊN ĐỀ 2

Giảng viên: TS Nguyễn Văn Hiếu

Nhóm 16

Sinh viên: Trần Thanh Tín - 106210253

Nguyễn Văn Chí - 106210207

Lê Quang Hùng - 106210238

Nhóm: 21.44

Đà Nẵng, Tháng 12 năm 2025

MỤC LỤC

1. Phân công công việc

Tên	Nội dung	Tỷ lệ đóng góp
Nguyễn Văn Chí	- Viết phần mở đầu, đặt vấn đề, các kỹ thuật Phishing - Thiết kế mô hình hệ thống, sơ đồ hoạt động - Phân tích và xây dựng bộ đặc trưng dữ liệu (Bảng thuộc tính trang 14-15: *HTML_Content, NumURL, body_wDear...*). - Triển khai thuật toán: K-Nearest Neighbors (KNN).	33.3%
Lê Quang Hùng	- Viết phần Mô hình hóa toán học: Không gian dữ liệu, hàm mất mát, bài toán tối ưu. - Triển khai và viết lý thuyết thuật toán: Cây quyết định (Decision Tree) và Rừng ngẫu nhiên (Random Forest). - Phân tích độ phức tạp thuật toán (Time Complexity) cho nhóm LR, DT, RF.	33.3%
Trần Thanh Tín	- Triển khai và viết lý thuyết thuật toán: Naive Bayes và Hồi quy Logistic (Logistic Regression). - Phân tích độ phức tạp cho nhóm NB, KNN. - Phân tích kết quả : Tính toán các chỉ số (Precision, Recall, F1...), vẽ Ma trận nhầm lẫn, vẽ biểu đồ so sánh và viết kết luận - Tổng hợp báo cáo, định dạng văn bản.	33.3%

2. Nội dung dự án

2.1. Yêu cầu bài toán

2.1.1. Đặt vấn đề

Từ khi bùng nổ công nghệ, hầu hết các thiết bị thông minh đều được kết nối internet, thư điện tử (email) trở nên phổ biến, đóng vai trò quan trọng trong cuộc sống. Nếu như trước kia, thư từ được viết tay bằng giấy, mực thì nay đã được viết hầu hết bằng máy tính. Để chuyển thư từ người này cho người khác, trước kia sẽ cần một bên trung gian làm công việc này. Việc vận chuyển thư thủ công này tiêu tốn nhiều nhân lực và thời gian gây tốn chi phí, khả năng thư bị thất lạc cũng rất cao.

Thư điện tử ra đời đã làm cho việc trao đổi thư từ trở nên dễ dàng và nhanh chóng, không tốn chi phí vận chuyển. Thư điện tử rất hữu ích nhưng cũng có rất nhiều rủi ro. Trong quá trình trao đổi thư, những kẻ tấn công có thể sử dụng các công cụ, kỹ thuật để chặn bắt, sửa chữa và làm lộ thông tin ra bên ngoài. Việc này đòi hỏi chúng ta phải sử dụng các phương pháp bảo mật an toàn khi gửi thư như việc tạo chữ ký số, mã hóa thông tin trước khi gửi, ... Ngoài ra, kẻ tấn công còn có thể sử dụng những hình thức tinh vi hơn để chiếm đoạt thông tin người dùng. Đó là hình thức lừa đảo qua email hay còn gọi là phishing email. Kẻ tấn công hoàn toàn có thể mạo danh một cá nhân, tổ chức hay doanh nghiệp uy tín nhằm tạo những email mạo danh, lừa đảo để lấy cắp thông tin của nạn nhân.

Hiện nay, thời đại số hóa, nhiều công ty, doanh nghiệp, tổ chức đều tiến hành làm việc online, việc trao đổi email công việc ngày càng nhiều. Đây cũng là cơ hội cho những kẻ tấn công khai thác thông tin với mục đích chuộc lợi. Thậm chí, kẻ tấn công còn mạo danh các tổ chức từ thiện, gửi các email giả mạo để quyên góp từ thiện, đánh vào lòng tin của con người để chiếm đoạt tài sản bất hợp pháp, ...

Với hình thức lừa đảo tinh vi như vậy, bài toán đặt ra là phải làm sao để có thể phát hiện, lọc và ngăn chặn được những email lừa đảo, lấy lại lòng tin của con người vào internet và công nghệ. Một phương pháp hiệu quả hiện đã và đang được sử dụng là mô hình Machine Learning.

2.1.2. Phishing email

1. Khái niệm

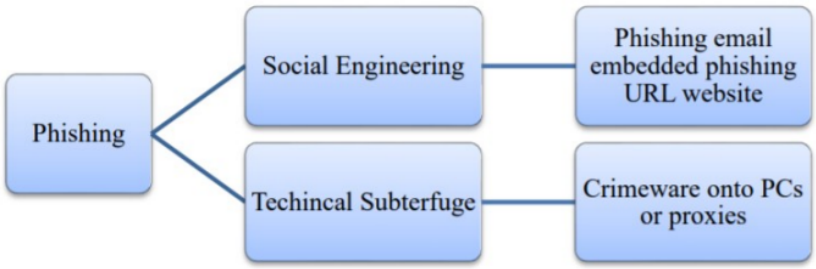
Phishing email là một hình thức lừa đảo trực tuyến, trong đó bọn tội phạm mạng mạo danh các tổ chức, cá nhân hợp pháp qua email, tin nhắn văn bản, quảng cáo hoặc các phương tiện khác để lấy thông tin nhạy cảm từ phía người dùng.

Hầu hết các thư lừa đảo được gửi qua email thường nhắm mục tiêu đến một cá nhân hoặc một công ty cụ thể. Nội dung của tin nhắn lừa đảo thường rất khác nhau, tùy theo mục đích của kẻ tấn công. Nhìn chung, phishing email được thiết kế gần giống với giao diện ngân hàng, tổ chức. Người dùng dễ dàng bị "lừa" nếu không để ý và tin rằng đó là email thật, sau đó họ dễ dàng cung cấp những thông tin cá nhân quan trọng như: Mật khẩu đăng nhập hệ thống, mật khẩu giao dịch, thẻ tín dụng và các thông tin tuyệt mật khác. Kẻ tấn công có thể sử dụng thông tin lấy cắp được để trực tiếp lấy tiền từ nạn nhân.

Trong một email lừa đảo, nạn nhân thường được yêu cầu cung cấp thông tin về: Ngày/tháng/năm sinh Số an sinh xã hội Số điện thoại Thông tin địa chỉ nhà Thông tin chi tiết về tài khoản ngân hàng, thẻ tín dụng Thông tin mật khẩu (các thông tin cần thiết để chúng có thể thay đổi và đặt lại mật khẩu tài khoản của bạn), ...

2. Các kỹ thuật phishing email

Kẻ lừa đảo sử dụng hai kỹ thuật chính để đạt được mục tiêu đó là lừa đảo Social Engineering (kỹ thuật xã hội) và Technical Subterfuge (phần mềm độc hại).



Hình 1: Các loại Email lừa đảo

Social Engineering:

- Hầu hết các kiểu lừa đảo sử dụng một hình thức đánh lừa kỹ thuật được thiết kế làm cho một liên kết được gửi đi trong email dường như thuộc về một tổ chức mà kẻ tấn công đang mạo danh. Đó là những liên kết gần giống với liên kết chính xác hoặc nó đã được biến đổi từ liên kết thật bằng cách sử dụng tên miền phụ.
- Ví dụ như URL sau: http://www.teckcombank.com/. Ở đây, URL được biến đổi giả mạo URL thật bằng cách sửa đổi 1 ký tự trong URL thật. Khi người dùng không nhìn kĩ sẽ nhấp vào URL này, URL sẽ đưa nạn nhân đến một trang web

giống hệt với trang web của ngân hàng techcombank để yêu cầu nhập tài khoản, mật khẩu giao dịch và kẻ tấn công dễ dàng lấy được thông tin tài khoản của nạn nhân.

- Hầu hết các loại lừa đảo liên quan đến một số loại kỹ thuật xã hội, trong đó người dùng bị thao túng về mặt tâm lý để thực hiện một hành động như nhấp vào liên kết, mở tệp đính kèm hoặc tiết lộ thông tin bí mật. Ngoài việc mạo danh một cá nhân, tổ chức hay một doanh nghiệp đáng tin cậy, kẻ tấn công còn tạo ra nội dung cấp bách, khẩn cấp như tuyên bố tài khoản của nạn nhân sẽ bị đóng băng hoặc bị khóa nếu như không thực hiện yêu cầu theo email mà chúng gửi. Việc này thường xảy ra với nạn nhân sử dụng tài khoản ngân hàng hay tài khoản bảo hiểm.
- Một kỹ thuật tinh xảo khác mà kẻ tấn công có thể sử dụng thay vì việc mạo danh đó là sử dụng những bài viết giả mạo được thiết kế để kích động sự phẫn nộ từ nạn nhân, khiến nạn nhân không do dự mà nhấp vào liên kế chúng gửi kèm trong email. Liên kết này sẽ đưa nạn nhân đến một trang web chuyên nghiệp giống hệt với trang web của một tổ chức hợp pháp nào đó. Khi đó, kẻ tấn công sẽ sử dụng các kỹ thuật khác để cố gắng khai thác lỗ hổng nhằm thu thập thông tin nhạy cảm của nạn nhân.
- Hiện nay, thời đại chuyển đổi số, kẻ tấn công đã sử dụng kỹ thuật xã hội bằng cách lợi dụng lòng tin và tình thương của con người để gửi các email lừa đảo để gây quỹ, ủng hộ cho người gặp khó khăn để thu lợi về mình.

Technical Subterfuge:

- Kỹ thuật lừa đảo dựa trên phần mềm độc hại không trực tiếp lấy thông tin của khách hàng, thay vào đó, chúng sử dụng các loại mã độc hoặc phần mềm độc hại kết hợp với các kỹ thuật khác. Nếu người dùng nhấp vào liên kết hoặc các tệp đính kèm được gửi trong email thì thiết bị của nạn nhân sẽ bị nhiễm các phần mềm độc hại. Các mã độc này sẽ khai thác các lỗ hổng trên thiết bị của nạn nhân để thực hiện lấy cắp thông tin một cách bí mật.

3. Hậu quả

Lừa đảo là kỹ thuật được sử dụng để lấy cắp thông tin cá nhân cho các mục đích khác nhau, sử dụng các email giả mạo có vẻ đến từ các tổ chức, doanh nghiệp uy tín, hợp pháp. Điều này thường được thực hiện bằng cách gửi các email dường như đến từ nguồn đáng tin cậy để có quyền truy cập vào thông tin bí mật và riêng tư của nạn nhân.

Email lừa đảo được coi là phương thức tội phạm trực tuyến gia tăng nhanh nhất được sử dụng để đánh cắp dữ liệu tài chính cá nhân và đánh cắp danh tính khiến bản thân và các tổ chức của họ gặp rủi ro:

- Gây mất niềm tin của khách hàng vào các cá nhân, tổ chức

- Lộ thông tin, kế hoạch, dự án của công ty gây thiệt hại tài sản, thậm chí là phá sản
- Mất niềm tin của người dùng vào internet, tổ chức từ thiện

4. Các kỹ thuật phát hiện

Phương pháp truyền thống: gồm 2 loại chính, một là bảo vệ bằng xác thực, hai là bảo vệ mở mức mạng. Bảo vệ ở mức mạng gồm 2 loại bộ lọc, bộ lọc danh sách đen và bộ lọc danh sách trắng được đưa vào sử dụng nhằm ngăn chặn các địa chỉ IP và domain lừa đảo từ mạng. Ngoài ra còn có bộ lọc dựa trên quy tắc và bộ lọc so sánh đối mẫu.

- Bộ lọc danh sách đen - Blacklist Filter:Bộ lọc này giúp bảo vệ ở lớp mạng bằng cách phân loại các địa chỉ DNS, địa chỉ IP của người nhận hoặc địa chỉ người gửi, trích xuất chi tiết header của email rồi so sánh nó với danh sách có trước. Nếu dữ liệu có trong danh sách thì email bị từ chối, nếu không thì email được chấp nhận.
- Bộ lọc danh sách trắng - Whitelist Filter: Đây cũng là bộ lọc giúp bảo vệ ở mức mạng. Phương pháp này sử dụng kỹ thuật lọc so sánh dữ liệu trích xuất từ email với dữ liệu được xác định trước trong danh sách chứa địa chỉ IP và IP tĩnh hợp pháp. Chỉ những email đến từ các địa chỉ hợp lệ có trong danh sách mới được chấp nhận, còn lại sẽ bị từ chối.
- Pattern Matching Filter: Đây cũng là bộ lọc bảo vệ ở mức mạng. Bộ lọc này sử dụng dữ liệu đã được chỉ định sẵn bao gồm cả chuỗi, từ, văn bản, ký tự được sử dụng trong nội dung của email. Nó sử dụng để phân loại các email theo danh sách mẫu, nếu email nhận được có lượng lớn các từ bị cấm thì sẽ bị loại bỏ.
- Xác thực email - Email verification: Hệ thống xác thực email yêu cầu việc xác nhận giữa người gửi và người nhận email. Các email được gửi đến mà không có xác nhận giữa người gửi và người nhận sẽ không được chuyển vào hộp thư đến. Việc sàng lọc email này mang lại độ chính xác cao trong việc xác định thư rác nhưng sẽ mất rất nhiều thời gian do phải đợi phản hồi từ phía người nhận. Điểm yếu nửa của là người gửi và người nhận dễ bị mất mát thông tin.
- Bộ lọc mật khẩu - Password Filter: Bộ lọc là bộ lọc xác thực cấp người dùng. Bộ lọc này cho phép nhận tất cả các email. Nếu bộ lọc phát hiện mật khẩu sai hoặc không phát hiện được mật khẩu thì email sẽ bị từ chối.

Phương pháp tự động

- Bên cạnh các phương pháp phát hiện truyền thống dựa trên quy tắc và danh sách định sẵn, phương pháp tự động dựa trên học máy (Machine Learning) ngày càng được sử dụng rộng rãi trong việc phát hiện email spam và email/tin nhắn lừa đảo.

- Khác với các phương pháp truyền thống, phương pháp học máy không phụ thuộc hoàn toàn vào các quy tắc cố định hay danh sách địa chỉ được xác định trước. Thay vào đó, hệ thống tự động học các đặc trưng và mẫu hành vi từ dữ liệu email đã được gán nhãn, từ đó xây dựng mô hình có khả năng phân biệt giữa email hợp lệ và email lừa đảo.
- Phương pháp học máy tiếp cận bài toán phát hiện phishing email như một bài toán phân loại, trong đó mỗi email hoặc tin nhắn được biểu diễn dưới dạng các đặc trưng và được gán vào một trong các lớp tương ứng như email hợp lệ, email spam hoặc email/tin nhắn lừa đảo.
- Ưu điểm nổi bật của phương pháp này là khả năng thích nghi với các hình thức lừa đảo mới, khi nội dung email liên tục thay đổi để tránh bị phát hiện bởi các bộ lọc dựa trên quy tắc. Ngoài ra, phương pháp học máy có thể xử lý khối lượng lớn email một cách tự động, giảm sự phụ thuộc vào can thiệp thủ công của con người.
- Tuy nhiên, phương pháp học máy cũng tồn tại một số hạn chế như phụ thuộc vào chất lượng dữ liệu huấn luyện, yêu cầu tài nguyên tính toán và có thể gặp khó khăn khi giải thích kết quả phân loại.

Trong phạm vi báo cáo này, phương pháp tự động dựa trên học máy được xem là hướng tiếp cận chính để giải quyết bài toán phân loại email spam và email/tin nhắn lừa đảo. Các mô hình, thuật toán cụ thể, cũng như quá trình huấn luyện và đánh giá sẽ được trình bày chi tiết ở các phần tiếp theo.

2.2. Mô hình hệ thống

2.2.1. Tổng quát hệ thống

Bài toán phân biệt và phân loại email spam, email/tin nhắn lừa đảo được mô hình hóa như một bài toán phân loại trong xử lý dữ liệu văn bản.

Mô hình tổng quát của bài toán gồm các thành phần chính:

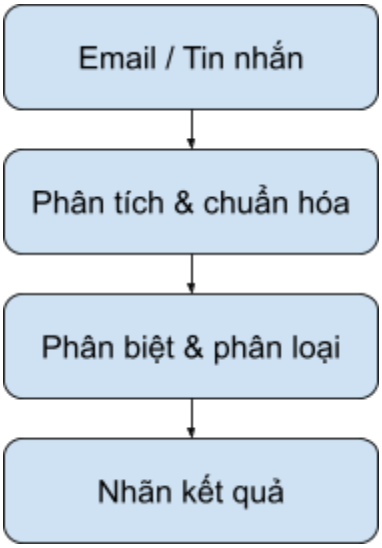
1. Dữ liệu đầu vào
- Email hoặc tin nhắn văn bản cần phân loại
- Nội dung có thể bao gồm văn bản, liên kết hoặc các thành phần phụ trợ khác

2. Khối xử lý và phân tích
- Chuẩn hóa và phân tích nội dung đầu vào
- Trích xuất các thông tin cần thiết phục vụ cho việc phân loại

3. Khối phân loại
- Thực hiện phân biệt email/tin nhắn dựa trên các đặc điểm đã phân tích
- Gán nhãn tương ứng cho mỗi email/tin nhắn

4. Kết quả đầu ra
- Nhãn phân loại thể hiện mức độ an toàn của email/tin nhắn

2.2.2. Biểu diễn mô hình bài toán



Hình 2: Luồng xử lý của bài toán

2.2.3. Nhiệm vụ của bài toán

Bài toán có các nhiệm vụ chính sau:

- Tiếp nhận nội dung email hoặc tin nhắn cần kiểm tra
- Phân tích nội dung và các dấu hiệu bất thường trong email/tin nhắn
- Phân biệt email hợp lệ và email không mong muốn
- Phân loại email/tin nhắn thành các nhóm: Email hợp lệ, Email spam/lừa đảo.
- Hỗ trợ người dùng nhận diện sớm các email/tin nhắn có nguy cơ gây hại

2.2.4. Cách thức hoạt động

Bước 1: Tiếp nhận dữ liệu

Nội dung email hoặc tin nhắn văn bản được đưa vào bài toán dưới dạng dữ liệu đầu vào. Dữ liệu này có thể bao gồm văn bản thuần, liên kết URL hoặc các thành phần phụ trợ khác.

Bước 2: Chuẩn hóa và phân tích nội dung

Nội dung đầu vào được chuẩn hóa nhằm đảm bảo tính thống nhất. Sau đó tiến hành phân tích các đặc điểm cơ bản của email/tin nhắn như:

- Nội dung văn bản
- Sự xuất hiện của liên kết
- Các dấu hiệu bất thường trong cách diễn đạt

Bước 3: Phân biệt và phân loại

Dựa trên kết quả phân tích, bài toán thực hiện phân biệt email/tin nhắn theo mức độ an toàn và gán nhãn phân loại tương ứng.

Bước 4: Trả kết quả phân loại

Kết quả cuối cùng của bài toán là nhãn phân loại của email/tin nhắn, phản ánh mức độ an toàn và nguy cơ lừa đảo.

2.2.5. Phạm vi và giới hạn

- Bài toán tập trung vào phân tích nội dung email/tin nhắn
- Không xét đến hành vi người dùng sau khi nhận email
- Không phân tích sâu mã độc trong tệp đính kèm

3. Mô tả và giải quyết bài toán

3.1. Mô hình hóa bài toán

3.1.1. Không gian dữ liệu và tập mẫu

Gọi $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^{N}$ là tập dữ liệu gồm N mẫu, trong đó:

- x_i là biểu diễn của email hoặc tin nhắn thứ i

- y_i là nhãn tương ứng của mẫu x_i

Mỗi email hoặc tin nhắn x_i được biểu diễn trong không gian đặc trưng:

$x_i \in \mathbb{R}^d$

với d là số chiều của vector đặc trưng trích xuất từ nội dung email hoặc tin nhắn.

3.1.2 Không gian nhãn

Bài toán phân loại nhị phân:
$y_i \in \{0, 1\}$

trong đó:

0: Email hợp lệ

1: Email spam hoặc email/tin nhắn lừa đảo

3.1.3. Hàm phân loại

Bài toán đặt ra là tìm một hàm phân loại:

$f : \mathbb{R}^d \to \mathcal{Y}$
sao cho nhãn dự đoán của email hoặc tin nhắn x_i được xác định bởi:
$\hat{y}_i = f(x_i)$
trong đó:

- $\mathcal{Y}$ là tập nhãn hợp lệ
- $\hat{y}_i$ là nhãn dự đoán

3.1.4. Hàm mất mát

Hàm mất mát giữa nhãn dự đoán và nhãn thực tế:
$\mathcal{L}(y_i, \hat{y}_i)$
Hàm mất mát tổng thể trên tập dữ liệu:

$$\mathcal{L}_{total} = \frac{1}{N} \sum_{i=1}^{N} \mathcal{L}(y_i, f(x_i))$$

Mục tiêu của bài toán là tối thiểu hóa giá trị hàm mất mát này.

3.1.5. Bài toán tối ưu

Bài toán phân loại được mô hình hóa dưới dạng bài toán tối ưu:

$$\min_{f} \frac{1}{N} \sum_{i=1}^{N} \mathcal{L}(y_i, f(x_i))$$

Nếu hàm phân loại phụ thuộc vào tham số θ:

$$f(x) = f(x; \theta)$$

thì bài toán trở thành:

$$\min_{\theta} \frac{1}{N} \sum_{i=1}^{N} \mathcal{L}(y_i, f(x_i; \theta))$$

3.1.6. Quy tắc quyết định

Nhãn dự đoán được xác định theo quy tắc:

$$\hat{y} = \arg\max_{c \in \mathcal{Y}} P(y = c \mid x)$$

trong đó $P(y = c \mid x)$ là xác suất email hoặc tin nhắn x thuộc lớp c.

3.2. Phương pháp giải quyết bài toán

Bộ dữ liệu được sử dụng cho bài toán phân loại nhị phân này bao gồm 9804 email, được phân loại thành hai lớp: email hợp lệ (legit) và email lừa đảo (phishing/spam).

Nguồn gốc dữ liệu:

- Email lừa đảo (Phishing/Spam): Thu thập từ website monkey.org.
- Email hợp lệ (Legit): Bộ dữ liệu Enron lấy từ classes.cs.uchicago.edu.

Phân bổ dữ liệu tổng thể:

- Email hợp lệ: 7200 mẫu.
- Email Phishing/Spam: 2604 mẫu.

- Tổng cộng: 9804 mẫu.

Cấu trúc chia tập Huấn luyện (Training) và Kiểm tra (Testing):

Dataset	**Email Phishing**	**Email Hợp lệ**	**Tổng**
Training	2104	5739	7843
Testing	500	1461	1961

Bảng 1: Tổng quan Dataset

Mỗi email hoặc tin nhắn được biểu diễn dưới dạng vector đặc trưng x_i, bao gồm các đặc trưng về cấu trúc, nội dung và các dấu hiệu lừa đảo tiềm năng. Các đặc trưng chính được sử dụng bao gồm:

Tên Thuộc tính	**Mô tả**
Cấu trúc & Định dạng	
HTML_Content	Trả về 1 nếu email có nội dung HTML, ngược lại là 0.
LenSubject	Trả về số lượng từ có trong Subject của email.
bodyImg	Trả về 1 nếu tìm thấy thẻ <img> trong body của email, ngược lại là 0.
Từ khóa Nội dung Lừa đảo	
sub_wBank	Trả về 1 nếu tìm thấy từ “bank” trong Subject, ngược lại là 0.
sub_wDebit	Trả về 1 nếu tìm thấy từ “debit” trong Subject, ngược lại là 0.
sub_wVerify	Trả về 1 nếu tìm thấy từ “verify” trong Subject, ngược lại là 0.
body_wDear	Trả về 1 nếu tìm thấy từ “dear” trong body của email, ngược lại là 0.
body_wSuspension	Trả về 1 nếu tìm thấy từ “suspension” trong body của email,

	ngược lại là 0.
BlackList	Trả về số lượng các từ nhạy cảm (như "account", "password", "bank") được tìm thấy trong email.
Đặc trưng URL & Liên kết	
NumURL	Trả về số lượng URL trong email.
NumIP	Trả về số lượng địa chỉ IP trong URL (thay vì tên miền).
HaveAt	Trả về 1 nếu tìm thấy kí tự "@" trong URL (che giấu tên miền thật), ngược lại là 0.
Redirection	Trả về 1 nếu tìm thấy ký tự "//" bổ sung trong URL (chuyển hướng), ngược lại là 0.
httpDomain	Trả về 1 nếu tìm thấy https trong tên miền (chỉ báo bảo mật), ngược lại là 0.
tinyURL	Trả về 1 nếu tìm thấy link rút gọn trong email, ngược lại là 0.
Prefix-Suffix	Trả về 1 nếu tìm thấy kí tự "-" trong URL (giả mạo tên miền), ngược lại là 0.
url_wClick, url_wHere, url_wLogin, url_wUpdate	Trả về 1 nếu tìm thấy các từ khóa hành động trong URL, ngược lại là 0.
Kỹ thuật Che giấu & Mã độc	
NumScript	Trả về số lượng các khối <script> (thực thi mã) trong email.
iFrame	Trả về 1 nếu iframe rỗng hoặc không tìm thấy phản hồi (che giấu nội dung), ngược lại là 0.
OnMouseOver	Trả về 1 nếu tìm thấy onmouseover hoặc không tìm thấy phản hồi (che giấu liên kết thực), ngược lại là 0.
RightClick	Trả về 1 nếu không tìm thấy sự kiện right click (chặn kiểm tra nguồn), ngược lại là 0.
Web-traffic	Trả về 1 nếu xếp hạng web Alexa `< 100000 (chỉ báo tin cậy), ngược lại là 0.

Bảng 2: Các đặc trưng

Chi tiết các thuật toán học máy được sử dụng để giải quyết bài toán phân loại

email/tin nhắn (spam/legit). Các phương pháp bao gồm: Cây quyết định (Decision Tree), K-NEAREST NEIGHBORS (KNN), Naive Bayes (NB), Random Forest - đại diện cho nhóm Ensemble và Hồi quy Logistic (Logistic Regression).

3.2.1. Decision Tree - DT

Cây quyết định là một mô hình phân loại hoạt động dựa trên việc chia nhỏ không gian dữ liệu thành các vùng nhỏ hơn thông qua các quy tắc quyết định (decision rules) dưới dạng cấu trúc cây.

1. Nguyên lý:

Mô hình học cách phân chia dữ liệu dựa trên giá trị của các đặc trưng $x^{(j)}$ (với $j = 1...d$) để tối đa hóa độ tinh khiết (purity) của các nút con.

2. Tiêu chuẩn phân chia:

Để chọn đặc trưng phân chia tốt nhất tại mỗi nút, ta sử dụng các chỉ số đo lường độ vẩn đục (impurity) như Entropy hoặc Gini Index.

- Entropy tại một nút *S*:
  $H(S) = -\sum_{c \in C} p_c \log_2(p_c)$
  Trong đó:
  - $C = \{0, 1\}$ là tập nhãn lớp.
  - p_c là xác suất (tần suất) của lớp c trong nút S.
- Information Gain (IG) (Lượng thông tin thu được) khi chia nút S theo đặc trưng A:
  $$IG(S, A) = H(S) - \sum_{v \in Values(A)} \frac{|S_v|}{|S|} H(S_v)$$
  Trong đó S_v là tập con của S mà tại đó đặc trưng A có giá trị v.

3. Hàm quyết định:

Duyệt từ nút gốc đến nút lá dựa trên giá trị đặc trưng của mẫu x. Nhãn dự đoán $\hat{y}$ là nhãn chiếm đa số tại nút lá đó.

3.2.2. K-Nearest Neighbors - KNN

KNN là thuật toán học dựa trên trường hợp (instance-based learning), không cần quá

trình huấn luyện mô hình (lazy learning). Việc phân loại dựa trên khoảng cách giữa mẫu cần dự đoán và các mẫu trong tập huấn luyện.

1. Khoảng cách:

Khoảng cách giữa hai mẫu x_i và x_j trong không gian d chiều thường được tính bằng khoảng cách Euclid:

$$d(x_i, x_j) = \sqrt{\sum_{k=1}^{d}(x_{i,k} - x_{j,k})^2}$$

2. Quy tắc quyết định:

Với một mẫu mới x, thuật toán tìm tập hợp $N_K(x)$ gồm K mẫu gần nhất trong tập huấn luyện $\mathcal{D}$. Nhãn dự đoán được xác định bởi quy tắc bỏ phiếu đa số:

$$\hat{y} = \underset{c \in \{0,1\}}{\operatorname{argmax}} \sum_{x_i \in N_K(x)} \mathbb{I}(y_i = c)$$

Trong đó $\mathbb{I}(\cdot)$ là hàm chỉ thị (trả về 1 nếu điều kiện đúng, ngược lại là 0).

3.2.3. Naive Bayes (NB)

Naive Bayes là thuật toán phân loại dựa trên xác suất, áp dụng định lý Bayes với giả định "ngây thơ" rằng các đặc trưng độc lập với nhau khi biết trước nhãn lớp.

1. Định lý Bayes:

Xác suất hậu nghiệm (posterior) của nhãn y khi biết đặc trưng x:

$$P(y|x) = \frac{P(x|y)P(y)}{P(x)}$$

2. Giả định độc lập có điều kiện:

Giả sử các đặc trưng $x_1, x_2, ..., x_d$ độc lập với nhau khi biết y, ta có:

$$P(x|y) \approx \prod_{j=1}^{d} P(x_j|y)$$

3. Hàm phân loại (MAP - Maximum A Posteriori):

Vì $P(x)$ là hằng số đối với mọi lớp, ta chỉ cần tối đa hóa tử số:

$$\hat{y} = \underset{y \in \{0,1\}}{\operatorname{argmax}} \left(P(y) \prod_{j=1}^{d} P(x_j|y) \right)$$

Trong bài toán phân loại văn bản/email, $P(x_j|y)$ thường được ước lượng bằng phân phối Multinomial hoặc Bernoulli (tần suất từ xuất hiện trong lớp spam/non-spam).

3.2.4. Random Forest - RF

Random Forest là một phương pháp học tổ hợp (Ensemble Learning) sử dụng kỹ thuật Bagging (Bootstrap Aggregating) kết hợp với việc chọn ngẫu nhiên các đặc trưng.

1. Nguyên lý:

Xây dựng M cây quyết định $\{T_1, T_2, ..., T_M\}$ độc lập.

- Mỗi cây được huấn luyện trên một tập dữ liệu con được lấy mẫu lặp lại (bootstrap) từ $\mathcal{D}$.
- Tại mỗi nút phân chia, chỉ một tập con ngẫu nhiên các đặc trưng được xem xét.

2. Hàm quyết định:

Kết quả dự đoán cuối cùng được tổng hợp từ kết quả của M cây (thông qua cơ chế bỏ phiếu số đông):

$$\hat{y} = \text{mode}\{T_1(x), T_2(x), ..., T_M(x)\}$$

Phương pháp này giúp giảm hiện tượng quá khớp (overfitting) thường gặp ở cây quyết định đơn lẻ.

3.2.5. Logistic Regression - LR

Mặc dù có tên là "hồi quy", đây là thuật toán phân loại tuyến tính mạnh mẽ cho bài toán nhị phân.

1. Hàm giả thuyết:

Mô hình sử dụng hàm Sigmoid để ánh xạ giá trị đầu ra tuyến tính vào khoảng $(0, 1)$, biểu thị xác suất mẫu thuộc lớp 1 (Spam):

$$h_\theta(x) = \sigma(w^T x + b) = \frac{1}{1 + e^{-(w^T x + b)}}$$

Trong đó:

- $w \in \mathbb{R}^d$ là vector trọng số.
- $b \in \mathbb{R}$ là hệ số chệch (bias).

$\sigma(z) = \frac{1}{1 + e^{-z}}$ là hàm Sigmoid.

2. Hàm mất mát (Log-Loss / Cross-Entropy):

Để tìm tham số w, b tối ưu, ta cực tiểu hóa hàm mất mát:

$$J(w, b) = -\frac{1}{N} \sum_{i=1}^{N} [y_i \log(h_\theta(x_i)) + (1 - y_i) \log(1 - h_\theta(x_i))]$$

3. Quy tắc quyết định:

Ngưỡng phân loại thường được chọn là 0.5:

$$\hat{y} = \begin{cases} 1 & \text{if } h_\theta(x) \geq 0.5 \\ 0 & \text{if } h_\theta(x) < 0.5 \end{cases}$$

3.3. Đánh giá phương pháp

Gọi N là số lượng mẫu trong tập dữ liệu, d là số chiều của vector đặc trưng.

Phương pháp	Độ phức tạp khi Huấn luyện (Training)	Độ phức tạp khi Dự đoán (Testing/Inference)	Tốc độ Hội tụ (Convergence)
Hồi quy Logistic (LR)	O(k · N · d)	O(d)	Hội tụ tuyến tính (Linear), phụ thuộc vào k (số lần lặp của Gradient Descent)

Naive Bayes	O(N · d)	O(d)	Không có khái niệm hội tụ (Là phương pháp tối ưu giải tích, không lặp)
Decision Tree	O(N · d · logN)	O(d)	Không có khái niệm hội tụ (Là phương pháp phân chia đệ quy)
Random forest	O(M · N · d · logN)	O(M · d)	Không áp dụng (Là tổ hợp của nhiều DT)
KNN	O(1) (Không có giai đoạn huấn luyện)	O(N · d)	Không có khái niệm hội tụ (Dựa trên khoảng cách)

Bảng 3: Đánh giá các phương pháp

3.3.1. Phân tích chi tiết về Độ phức tạp

A. Hồi quy Logistic (LR)

- Độ phức tạp Huấn luyện: $O(k \cdot N \cdot d)$.
  + Thuật toán tối ưu (thường là Gradient Descent) lặp k lần.
  + Trong mỗi lần lặp, ta cần tính đạo hàm (gradient) của hàm mất mát trên toàn bộ N mẫu, mỗi mẫu có d chiều.
  + LR được coi là độ phức tạp huấn luyện cao so với NB, nhưng hiệu quả cao cho dữ liệu thưa thớt (sparse data).
- Độ phức tạp Dự đoán: $O(d)$.
  + Chỉ cần tính tích vô hướng $w^T x + b$ và áp dụng hàm Sigmoid, rất nhanh.
- Tốc độ Hội tụ: Hàm mất mát Cross-Entropy là hàm lồi (convex), đảm bảo thuật toán tối ưu (Gradient Descent) sẽ hội tụ về nghiệm tối ưu toàn cục (Global Minimum). Tốc độ hội tụ thường là tuyến tính (linear) hoặc siêu tuyến tính (super-linear) nếu dùng các biến thể tối ưu hơn.

B. Naive Bayes (NB)

- Độ phức tạp Huấn luyện: $O(N \cdot d)$.
  + Giai đoạn huấn luyện chỉ là việc đếm và thống kê tần suất xuất hiện của từ/đặc trưng trong mỗi lớp (spam/non-spam). Đây là một trong những thuật toán nhanh nhất để huấn luyện.
- Độ phức tạp Dự đoán: $O(d)$.

+ Chỉ cần tính tích của d xác suất có điều kiện và so sánh. Rất nhanh.

C. Decision Tree (DT) và Random forest (RF)

Độ phức tạp Huấn luyện (DT): $O(N \cdot d \cdot \log N)$.

- Việc tìm ngưỡng phân chia tối ưu trên N mẫu đòi hỏi phải sắp xếp (sorting) hoặc quét qua các giá trị. Độ phức tạp này tương đối cao nhưng vẫn chấp nhận được.

Độ phức tạp Huấn luyện (RF): $O(M \cdot N \cdot d \cdot \log N)$.

- Do RF xây dựng M cây quyết định gần như độc lập, độ phức tạp nhân lên M lần. RF có độ phức tạp huấn luyện cao nhất trong nhóm này.

Độ phức tạp Dự đoán (DT/RF): $O(d)$ hoặc $O(M \cdot d)$.

- Việc đi từ gốc đến lá trong cây rất nhanh chóng.

D. K-Nearest Neighbors (KNN)

Độ phức tạp Huấn luyện: $O(1)$.

- KNN chỉ lưu trữ toàn bộ tập dữ liệu, không có tính toán trong giai đoạn này.

Độ phức tạp Dự đoán: $O(N \cdot d)$.

- Đây là nhược điểm lớn của KNN. Để phân loại một mẫu, ta phải tính khoảng cách đến toàn bộ N mẫu huấn luyện.
- Khi N lớn, KNN trở nên rất chậm trong giai đoạn dự đoán.

3.3.2. Kết luận về Tốc độ và Khả năng mở rộng

Tốc độ Huấn luyện: NB (nhanh nhất) $\gg$ LR $\gg$ DT $\gg$ RF (chậm nhất).

Tốc độ Dự đoán: NB/LR/DT (nhanh nhất) $\gg$ RF $\gg$ KNN (chậm nhất khi N lớn).

Khả năng mở rộng (Scalability): LR và NB dễ dàng mở rộng cho dữ liệu lớn (Big Data) vì chúng hoạt động hiệu quả với các phương pháp tối ưu hóa phân tán hoặc là mô hình học theo lô (batch learning) đơn giản. KNN có khả năng mở rộng kém nhất do chi phí dự đoán cao.

4. Phân tích kết quả

4.1. Thông số đánh giá hệ thống

Từ các thuật toán, ta xác định được những số liệu để đánh giá hiệu suất thu được từ bộ dữ liệu đã được xử lí các đặc trưng

Độ chính xác: số liệu này được tính dựa trên tỷ lệ phần trăm của quyết định chính xác trong số tất cả các mẫu thử nghiệm

$$\mathrm{ACC} = \frac{\mathrm{TP} + \mathrm{TN}}{\mathrm{TP} + \mathrm{TN} + \mathrm{FP} + \mathrm{FN}}$$

trong đó

- True positive (TP): Số các ca dự đoán dương tính đúng hay dương tính thật.
- True negative (TN): Số các ca dự đoán âm tính đúng hay âm tính thật.
- False positive (FP): Số các ca dự đoán dương tính sai hay dương tính giả.
- False negative (FN): Số các ca dự đoán âm tính sai hay âm tính giả.

Ma trận nhầm lẫn: là một trong những kỹ thuật đo lường hiệu suất phổ biến nhất và được sử dụng rộng rãi cho các mô hình phân loại. Nó cho phép trực quan hóa hiệu suất của một thuật toán.

Thực tế\Dự tính	Dương tính	Âm tính
Dương tính	TP	FP
Âm tính	FN	TN

Bảng 4: Ma trận nhầm lẫn

Precision: trả lời cho câu hỏi trong các trường hợp được dự báo là positive thì có bao nhiêu trường hợp là đúng

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

Recall: đo lường tỷ lệ dự báo chính xác các trường hợp positive trên toàn bộ các mẫu thuộc nhóm positive

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

F1 Score: là trung bình điều hòa giữa precision và recall. Do đó nó đại diện hơn trong

việc đánh giá độ chính xác trên đồng thời precision và recall.

$$\text{F1-Score} = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}} = \frac{2 \cdot \text{TP}}{2 \cdot \text{TP} + \text{FP} + \text{FN}}$$

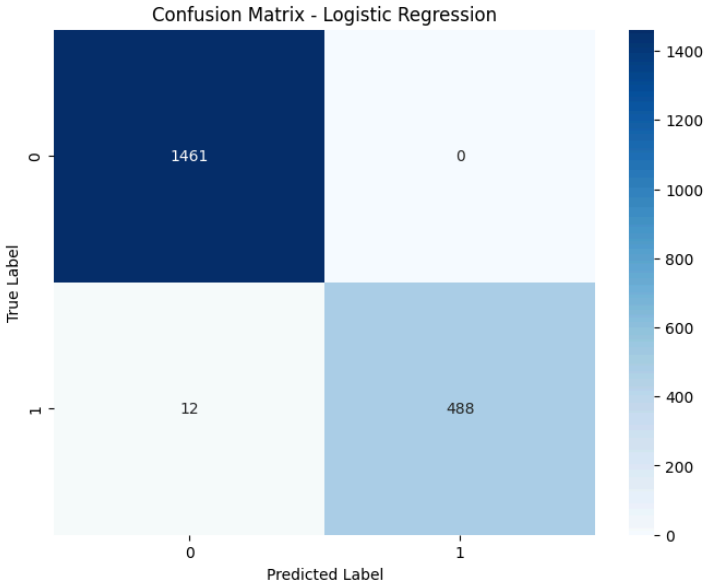
FPR: là tỷ lệ dự đoán sai

$$\text{FPR} = \frac{\text{FP}}{\text{FP} + \text{TN}}$$

4.2. Mô tả kết quả

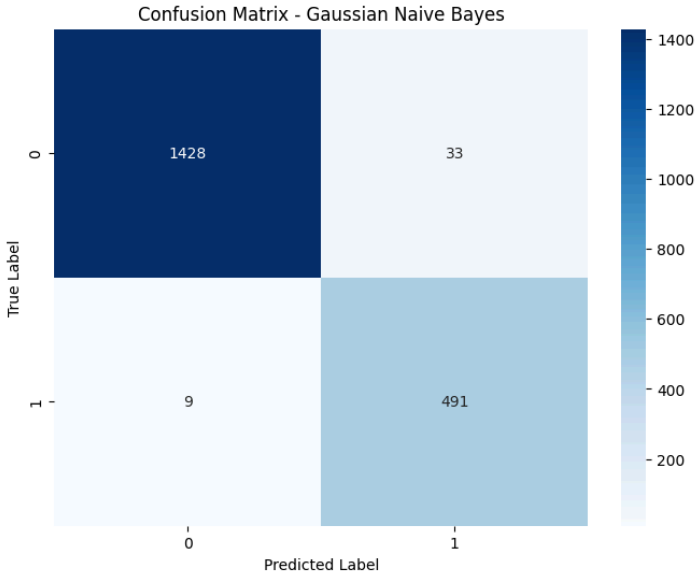
Regression Logistic

Accuracy: 0.9939
Precision: 1.0000
Recall: 0.9760
F1-Score: 0.9879
False Positive Rate (FPR): 0.0000



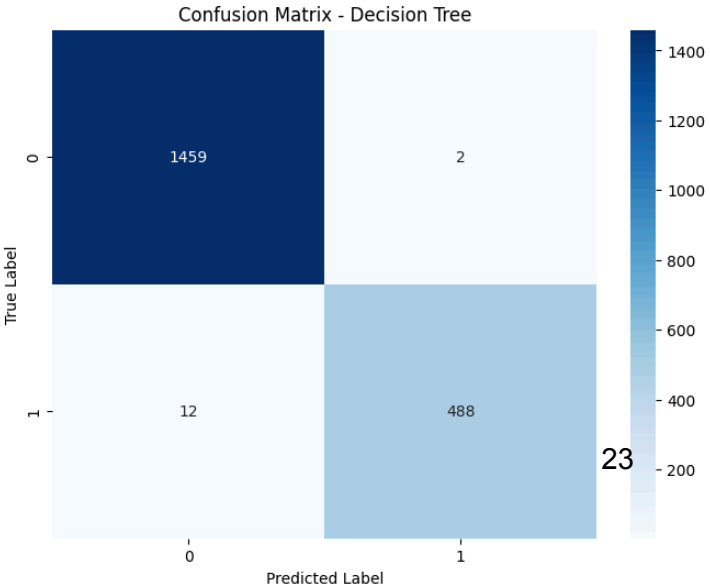
Naive Bayes

Accuracy: 0.9786
Precision: 0.9370
Recall: 0.9820
F1-Score: 0.9590
False Positive Rate (FPR): 0.0226

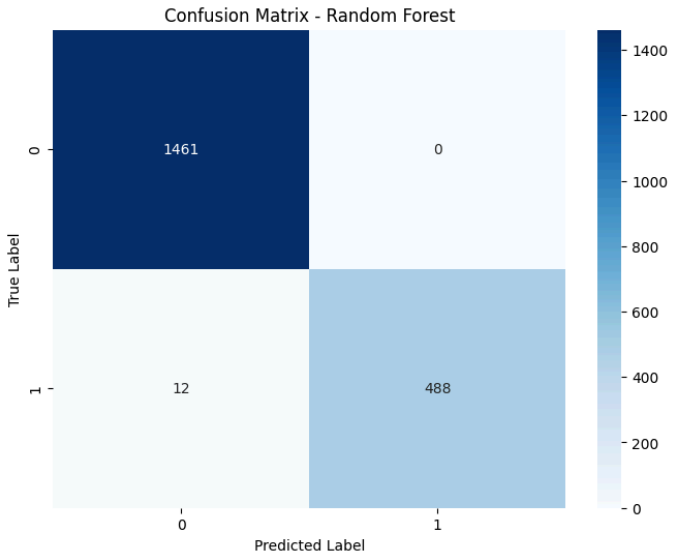


Decision Tree

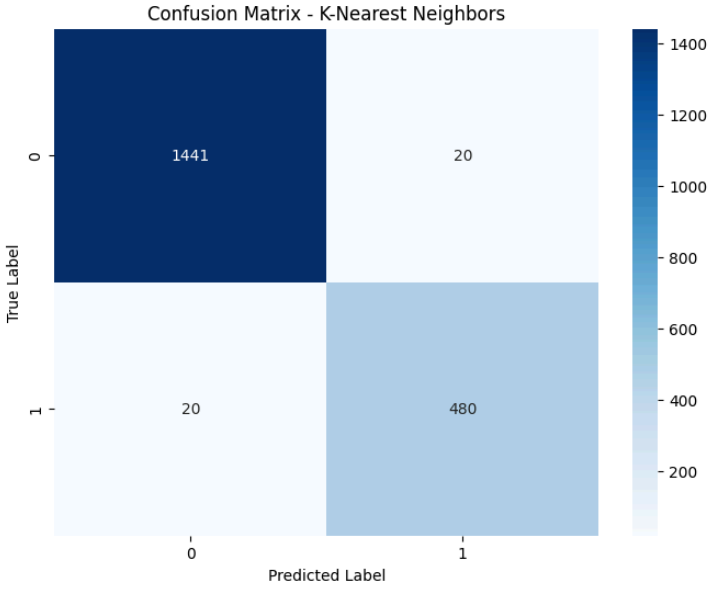
Accuracy: 0.9929
Precision: 0.9959
Recall: 0.9760
F1-Score: 0.9859
False Positive Rate (FPR): 0.0014



Random Forest
Accuracy: 0.9939
Precision: 1.0000
Recall: 0.9760
F1-Score: 0.9879
False Positive Rate (FPR): 0.0000



K-Nearest Neighbors
Accuracy: 0.9796
Precision: 0.9600
Recall: 0.9600
F1-Score: 0.9600
False Positive Rate (FPR): 0.0137



Kết quả thực nghiệm cho thấy các mô hình học máy có khả năng phân loại email/tin nhắn lừa đảo (phishing/spam) với độ chính xác rất cao.

- Hiệu năng vượt trội của mô hình tuyến tính và Ensemble: Hồi quy Logistic (LR) và Rừng Ngẫu nhiên (RF) dẫn đầu về hiệu năng tổng thể, đạt Accuracy và F1-Score cao nhất (0.9939 và 0.9879 tương ứng).
- Độ an toàn gần như tuyệt đối (Zero False Positive): LR và RF đạt Precision = 1.0000 và FPR = 0.0000. Điều này có nghĩa là, trên tập kiểm tra, không có bất kỳ

email hợp lệ nào bị đánh dấu nhầm là email spam. Đây là một thành tích rất quan trọng trong các hệ thống lọc spam.

- Khả năng phát hiện (Recall): Naive Bayes (NB) là mô hình có Recall cao nhất (0.9820), nghĩa là nó phát hiện được nhiều email spam nhất, chỉ bỏ sót khoảng 1.8% email spam thực tế (FN thấp nhất).
- Mô hình kém hiệu quả nhất: KNN có F1-Score thấp nhất (0.9600) và FPR cao thứ hai (0.0137), cho thấy nó không hiệu quả bằng các phương pháp khác trên bộ dữ liệu đặc trưng này.

4.3. Phân tích thông số

4.3.1. Tính hợp lý và Tiêu chuẩn đáp ứng

Trong hệ thống lọc spam, hai rủi ro cần được cân nhắc là:

- Đánh dấu nhầm email hợp lệ thành Spam (False Positive - FP): Gây thiệt hại lớn cho người dùng, có thể dẫn đến mất dữ liệu hoặc thông tin quan trọng.
- Bỏ sót email Spam (False Negative - FN): Gây phiền toái và rủi ro bảo mật cho người dùng.

Chỉ số	Yêu cầu/Tiêu chuẩn	Phân tích tính hợp lý
Precision	Cần đạt giá trị tối đa (gần 1.000)	LR và RF (1.0000) đáp ứng tiêu chuẩn này một cách hoàn hảo, cho thấy độ tin cậy tuyệt đối khi mô hình đưa ra cảnh báo spam.
FPR	Cần đạt giá trị tối thiểu (gần 0.000)	LR và RF (0.0000) đạt yêu cầu lý tưởng. Ngược lại, NB (0.0226) có FPR cao nhất, cho thấy NB có thể không phù hợp cho môi trường yêu cầu độ an toàn cao.
Recall	Cần đạt giá trị cao	NB (0.9820) dẫn đầu, nhưng sự khác biệt so với LR/RF (0.9760) là không đáng kể.

F1-Score	Cân bằng giữa Precision và Recall	LR và RF (0.9879) là các mô hình cân bằng và tối ưu nhất, cho thấy khả năng phân loại chính xác tổng thể.

Bảng 5: Tính hợp lý, tiêu chuẩn dựa theo các chỉ số

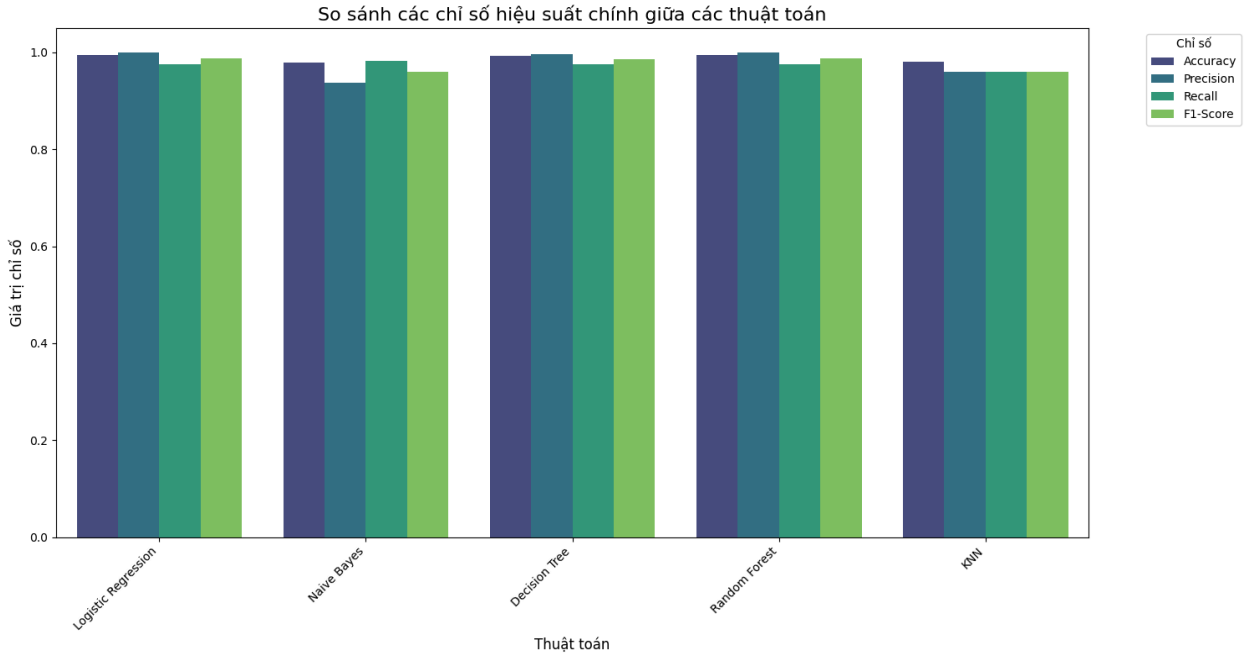
4.3.2. Tính ổn định (Stability)

- Rừng Ngẫu nhiên (RF): Là một mô hình rất ổn định vì nó được xây dựng trên nhiều cây quyết định độc lập (Ensemble). Việc đạt Precision = 1.0000 chứng tỏ RF đã giảm thiểu thành công sự dao động và quá khớp (overfitting) của các cây quyết định đơn lẻ (DT).
- Cây Quyết định (DT): Hiệu năng cao (F1=0.9859) nhưng có FPR nhỏ hơn 0, chứng tỏ DT đơn lẻ có nguy cơ quá khớp với tập huấn luyện và kém ổn định hơn so với RF.
- Naive Bayes (NB): Mặc dù đơn giản và nhanh, NB hoạt động ổn định nhưng bị hạn chế bởi giả định độc lập giữa các đặc trưng, dẫn đến Precision và FPR kém hơn các mô hình khác.

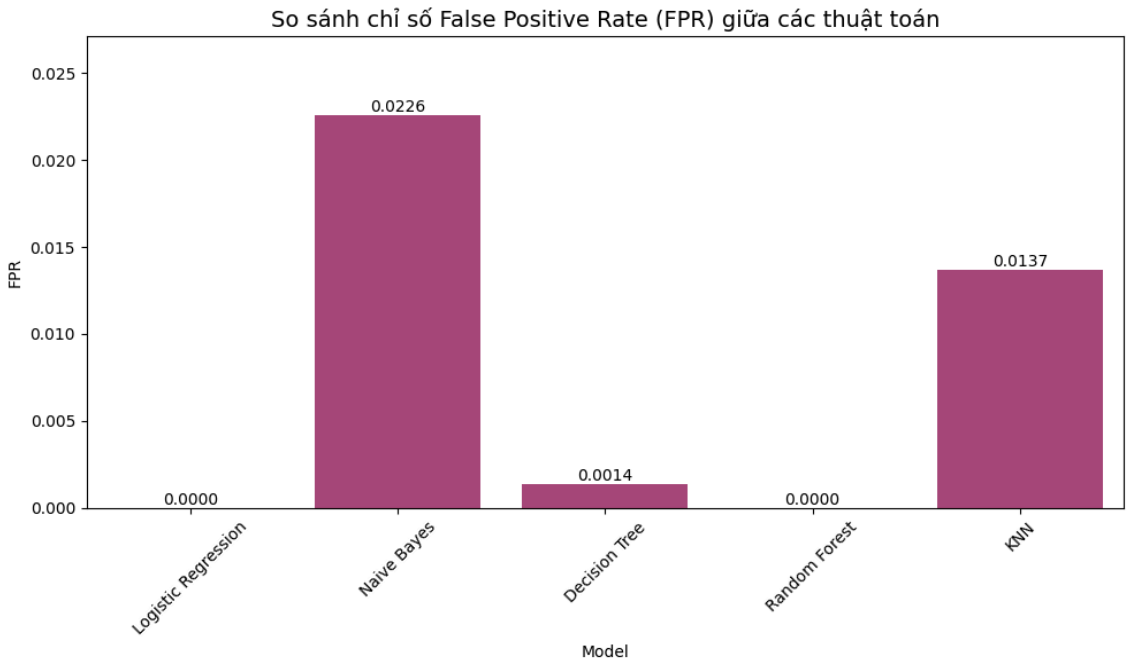
4.4. So sánh

Model	Accuracy	Precision	Recall	F1-Score	FPR
Logistic Regression	0.993881	1.000000	0.976	0.987854	0.000000
Naive Bayes	0.978582	0.937023	0.982	0.958984	0.022587
Decision Tree	0.992861	0.995918	0.976	0.985859	0.001369
Random Forest	0.993881	1.000000	0.976	0.987854	0.000000
KNN	0.979602	0.960000	0.960	0.960000	0.013689

Bảng 6: So sánh các thông số giữa các thuật toán



Hình 3: So sánh các chỉ số hiệu suất giữa các thuật toán



Hình 4: So sánh FPR giữa các thuật toán

Dựa trên sự kết hợp giữa hiệu năng và độ phức tạp, ta có thể so sánh chi tiết các thuật toán:

Thuật toán	Ưu điểm Chính	Nhược điểm Chính	Khuyến nghị cho Bài toán
LR	Precision = 1.0000, FPR=0.0000 (tuyệt đối an toàn), tốc độ dự đoán cực nhanh (O(d)).	Khó giải thích nếu không gian đặc trưng là phi tuyến tính cao.	Lựa chọn tối ưu nhất do kết hợp hiệu suất cao và chi phí vận hành thấp.
RF	Precision = 1.0000, FPR=0.0000, hiệu năng tổng thể cao nhất. Rất ổn định.	Thời gian huấn luyện lâu nhất (O(M . N . d . log N)) và độ phức tạp dự đoán cao hơn	Lựa chọn tốt nếu tính ổn định là ưu tiên hàng đầu, bất kể thời gian huấn luyện.

		LR (O(M . d)).	
DT	Dễ giải thích (Interpretability) và dự đoán nhanh.	Kém ổn định, dễ bị ảnh hưởng bởi nhiễu (FPR = 0.0014).	Chỉ nên dùng để tham khảo hoặc làm mô hình nền tảng cho RF.
NB	Recall cao nhất (TPR=0.9820) và tốc độ huấn luyện/dự đoán nhanh nhất (O(d)).	Precision và FPR thấp nhất, không đáp ứng yêu cầu an toàn cao.	Phù hợp cho giai đoạn lọc thô ban đầu hoặc khi tốc độ là yếu tố sống còn (latency-critical).
KNN	Không cần huấn luyện.	F1-Score thấp nhất và tốc độ dự đoán chậm nhất (O(N . d)).	Không phù hợp để triển khai trong môi trường sản phẩm do chi phí dự đoán cao.

Bảng 7: So sánh giữa các thuật toán

Kết luận So sánh:

Hồi quy Logistic (LR) là giải pháp được khuyến nghị nhất cho bài toán này. LR không chỉ đạt được hiệu năng tối ưu (ngang RF) mà còn mang lại ưu thế về tốc độ dự đoán, làm cho nó lý tưởng cho việc xử lý hàng triệu email trong thời gian thực.