**TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT TP.HCM**



**KHOA ĐÀO TẠO CHẤT LƯỢNG CAO**

**🕯✡🕮🕮✡🕯**



**ĐỀ TÀI**

**PHÂN LOẠI THƯ RÁC ÁP DỤNG THUẬT TOÁN**

**NAÏVE BAYES**

**GVHD: TS. Vũ Quang Huy**

**SVTH: MSSV**

**Nguyễn Đặng Nhật Minh 15110081**

**Lê Thị Phương Nha 15110090**

**Tp. Hồ Chí Minh, tháng 11 năm 2018**

# **LỜI NÓI ĐẦU**

Ngày nay, Internet mở ra nhiều kênh liên lạc, nhiều dịch vụ mới cho người sử dụng, một trong những dịch vụ mà Internet mang lại là dịch vụ thư điện tử (Email), đó là phương tiện giao tiếp rất đơn giản, tiện lợi và hiệu quả đối với cộng đồng người sử dụng dịch vụ này. Chính vì những lợi ích do thư mang lại nên số lượng thư trao đổi trên Internet ngày càng tăng, và một số không nhỏ trong đó là thư rác (Spam). Trong những năm gần đây, spam hay các thư không mong muốn đã trở thành một vấn nạn và đe dọa khả năng giao tiếp của con người trên kênh liên lạc này, đó là một trong những thách thức lớn mà khách hàng và các nhà cung cấp dịch vụ phải đối phó. Spam đã trở thành một hình thức quảng cáo chuyên nghiệp, phát tán virus, ăn cắp thông tin với nhiều thủ đoạn và mánh khóe cực kỳ tinh vi. Người dùng sẽ phải mất khá nhiều thời gian để xóa những thư “không mời mà đến”, nếu vô ý còn có thể bị nhiễm virus và nặng nề hơn là mất thông tin như thẻ tín dụng, tài khoản ngân hàng qua các thư dạng phishing.... Theo báo cáo tình hình thư rác do Kaspersky Lab vừa công bố, tỷ lệ thư rác trong lưu lượng truy cập thư của quý 3/2014 tăng 1,7 % so với quý trước, đạt trung bình 66,9%. Ba nguồn phát tán thư rác hàng đầu gồm có Mỹ (14%) và Nga (6,1%) và Việt Nam đứng vị trí thứ 3 với 6%. Để ngăn chặn spam, nhiều tổ chức, cá nhân đã nghiên cứu và phát triển những kỹ thuật phân loại thư thành các nhóm; từ đó xác định, nhận biết giữa thư rác và thư có giá trị. Tuy nhiên, những người tạo nên thư rác luôn tìm mọi cách vượt qua các bộ phân loại này và phát tán chúng. Vì vậy, cần có một hệ thống phân loại đâu là spam mail và đâu là mail tốt. Xuất phát từ thực trạng đó, tôi chọn hướng nghiên cứu “Phân loại thư rác áp dụng thuật toán Naïve Bayes” với mục đích tìm hiểu, thử nghiệm một số phương pháp tiếp cận cho bài toán phân loại thư, từ đó ngăn chặn thư spam hiệu quả hơn.

**MỤC LỤC**

[**LỜI NÓI ĐẦU** 2](#_Toc531644730)

[**CHƯƠNG 1**](#_Toc531644731). [**GIỚI THIỆU CHUNG** 4](#_Toc531644732)

[**1.1. Lý do chọn đề tài** 4](#_Toc531644733)

[**1.2. Mục đích** 4](#_Toc531644734)

[**1.3. Ý nghĩa** 4](#_Toc531644735)

[**CHƯƠNG 2**](#_Toc531644736). [**TỔNG QUAN VỀ GIẢI THUẬT** 5](#_Toc531644737)

[**2.1. Tổng quan** 5](#_Toc531644738)

[**2.2. Định luật Bayes** 5](#_Toc531644739)

[**2.3. Ứng dụng** 10](#_Toc531644740)

[**CHƯƠNG 3**](#_Toc531644741). [**GIẢI QUYẾT BÀI TOÁN LỌC THƯ RÁC** 11](#_Toc531644742)

[**3.1. Đặt vấn đề** 11](#_Toc531644743)

[**3.2. Bài toán** 11](#_Toc531644744)

[**3.3. Dùng luật Bayes tính xác suất** 11](#_Toc531644745)

[**3.4. Huấn luyện cho bộ lọc Bayes** 13](#_Toc531644746)

[**KẾT LUẬN** 17](#_Toc531644747)

# **CHƯƠNG 1**

# **GIỚI THIỆU CHUNG**

## **1.1. Lý do chọn đề tài**

Sự phát triển của khoa học công nghệ, đặc biệt là mạng truyền thông, đã và đang ngày càng phát triển, không chỉ đáp ứng nhu cầu thông tin hàng ngày, mà còn tạo ra những dịch vụ trợ giúp cho mọi người. Việc người dùng trên mạng truyền thông trong đổi thông tin qua những tin nhắn, email, diễn đàn (forum), … ngày càng trở nên phổ biến. Cùng với sự phát triển của dịch vụ gửi các thông tin qua các dịch vụ, ta cũng thấy được sự xuất hiện của những tin nhắn, thư rác với vô vàn loại nội dung khác nhau nhưng đa phần đều gây phiền nhiễu cho người dùng khi phải nhận được những thông tin này. Để hỗ trợ người sử dụng tự động phân loại các tin nhắn, email như vậy, hiện có rất nhiều phần mềm cũng như dịch vụ hỗ trợ, các chương trình hiện tại thường sử dụng những từ khoá, blacklist để phân loại, những cách làm này có hiệu quả nhưng không triệt để, vì vậy cần đưa ra giải pháp phân loại các tin nhắn, email như vậy dựa trên nội dung văn bản. Chính lý do trên em đã tập trung tìm hiểu về bài toán phân loại văn bản Text Categorization.

## **1.2. Mục đích**

Nội dung đề tài này định hướng việc phân loại thông tin của một tập lớn dữ liệu dựa trên phương pháp dự đoán hay gợi ý sử dụng thuật toán **Naïve Bayes**. Cụ thể, đề tài sẽ thực hiện phân loại bình luận tốt hay chưa tốt.

## **1.3. Ý nghĩa**

Thông qua việc phân tích các thông tin thể hiện trong câu bình luận nhằm hỗ trợ cơ chế trong việc xây dựng hệ thống đánh giá, phân loại thông tin nhằm hạn chế bớt sự phức tạp khi xử lý ngôn ngữ tự nhiên.

# **CHƯƠNG 2**

# **TỔNG QUAN VỀ GIẢI THUẬT NAÏVE BAYES CLASSIFIER**

## **2.1. Tổng quan**

Naive Bayes Classification (NBC) là một thuật toán dựa trên định lý Bayes về lý thuyết xác suất  để đưa ra các phán đoán cũng như phân loại dữ liệu dựa trên các dữ liệu được quan sát và thống kê. Naive Bayes Classification là một trong những thuật toán được ứng dụng rất nhiều trong các lĩnh vực Machine learning dùng để đưa các dự đoán chính xác nhất dự trên một tập dữ liệu đã được thu thập, vì nó khá dễ hiểu và độ chính xác cao. Nó thuộc vào nhóm Supervised Machine Learning Algorithms (thuật toán học có hướng dẫn), tức là máy học từ các ví dụ từ các mẫu dữ liệu đã có. Naive Bayes Classification (NBC) là một thuật toán dựa trên định lý Bayes về lý thuyết xác suất  để đưa ra các phán đoán cũng như phân loại dữ liệu dựa trên các dữ liệu được quan sát và thống kê. Naive Bayes Classification là một trong những thuật toán được ứng dụng rất nhiều trong các lĩnh vực Machine learning dùng để đưa các dự đoán chính xác nhất dự trên một tập dữ liệu đã được thu thập, vì nó khá dễ hiểu và độ chính xác cao. Nó thuộc vào nhóm Supervised Machine Learning Algorithms (thuật toán học có hướng dẫn), tức là máy học từ các ví dụ từ các mẫu dữ liệu đã có.

Ví dụ như ta có thể ứng dụng vào việc thiết kế một ứng dụng nghe nhạc có thể phán đoán được sở thích của nghe nhạc của người dùng dựa trên các hành vi như nhấn nút “thích” bài hát, “nghe đi nghe” lại nhiều lần các bài hát,  “bỏ qua” các bài hát không thích…. Dựa trên tập dữ liệu đó ta có thể áp dụng NBC để tính toán ra các phong cách nhạc mà người dùng thích nhất, từ đó chúng ta có thể đưa ra các “gợi ý” nghe nhạc gần đúng nhất cho người dùng từ việc học hỏi từ những thói quen đó.

## **2.2. Định luật Bayes**

Định luật Bayes được phát biểu như sau:

Định lý Bayes cho phép tính xác suất xảy ra của một sự kiện ngẫu nhiên A khi biết sự kiện liên quan B đã xảy ra. Xác suất này được ký hiệu là **P(A|B)**, và đọc là “xác suất của A nếu có B”. Đại lượng này được gọi xác suất có điều kiện hay xác suất hậu nghiệm vì nó được rút ra từ giá trị được cho của B hoặc phụ thuộc vào giá trị đó.

Theo định lí Bayes, xác suất xảy ra A khi biết B sẽ phụ thuộc vào 3 yếu tố:

\* Xác suất xảy ra A của riêng nó, không quan tâm đến B. Kí hiệu là **P(A)**và đọc là xác suất của A. Đây được gọi là xác suất biên duyên hay xác suất tiên nghiệm, nó là “tiên nghiệm” theo nghĩa rằng nó không quan tâm đến bất kỳ thông tin nào về B.

\* Xác suất xảy ra B của riêng nó, không quan tâm đến A. Kí hiệu là **P(B)**và đọc là “xác suất của B”. Đại lượng này còn gọi là hằng số chuẩn hóa (normalising constant), vì nó luôn giống nhau, không phụ thuộc vào sự kiện A đang muốn biết.

\* Xác suất xảy ra B khi biết A xảy ra. Kí hiệu là **P(B|A)** và đọc là “xác suất của B nếu có A”. Đại lượng này gọi là khả năng (likelihood) xảy ra B khi biết A đã xảy ra. Chú ý không nhầm lẫn giữa khả năng xảy ra B khi biết A và xác suất xảy ra A khi biết B.

Tóm lại định lý Bayes sẽ giúp ta tính ra xác suất xảy ra của một giả thuyết bằng cách thu thập các bằng chứng nhất quán hoặc không nhất quán với một giả thuyết nào đó. Khi các bằng chứng tích lũy, mức độ tin tưởng vào một giả thuyết thay đổi. Khi có đủ bằng chứng, mức độ tin tưởng này thường trở nên rất cao hoặc rất thấp, tức là xác xuất sảy ra giả thuyết sẽ thay đổi thì các bằng chứng liên quan đến nó thay đổi.

Công thức của định luật Bayes được phát biểu như sau:

Trong đó

– P(A|B) là  xác suất xảy ra của một sự kiện ngẫu nhiên A khi biết sự kiện liên quan B đã xảy ra.

– P(B|A) là xác suất xảy ra B khi biết A xảy ra

– P(A) là xác suất sảy ra của riêng A mà không quan tâm đến B.

– P(B) là xác suất xảy ra của riêng B mà không quan tâm đến A.

Ở trên ta có thể thấy xác suất sảy ra của giả thuyết A phụ thuộc và xác suất của giả thuyết B, nhưng trong thực tế xác suất A có thể phụ thuộc vào xác suất của nhiều các giác thuyết khác có thể là B1, B2, B3 … Bn. Vậy định luật Bayes có thể được mở rộng bằng công thức sau:

**Ví dụ**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Chills** | **Runny nose** | **Headache** | **Fever** | **Flu?** |
| Y | N | Mild | Y | N |
| Y | Y | No | N | Y |
| Y | N | Strong | Y | Y |
| N | Y | Mild | Y | Y |
| N | N | No | N | N |
| N | Y | Strong | Y | Y |
| N | Y | Strong | N | N |
| Y | Y | Mild | Y | Y |

Lớp: C1: Flu = “Y” (số người bị cúm)

C2: Flu = “N” (số người không bị cúm)

Chọn mẫu thử X= (Chills = “Y”, Runny nose = “N”, Headache = “Mild”, Fever = “N”)

**Tính P():**

P(C1) = = 0.625 (có 5 người bị cúm trong tổng 8 người khám)

P(C2) = = 0.375 (có 3 người không bị cúm trong tổng 8 người khám)

**Tính P(X|) cho mỗi lớp:**

P(Chills = “Y” | Flu = “Y”) = = 0.6 (có 3 người có triệu chứng ớn lạnh trong 5 người bị cúm)

P(Chills = “Y” | Flu = “N”) = = 0.33 (có 1 người có triệu chứng ớn lạnh trong 3 người không bị cúm)

P(Runny nose = “N” | Flu = “Y”) = = 0.2 (có 1 người không có triệu chứng sổ mũi trong 5 người bị cúm)

P(Runny nose = “N” | Flu = “N”) = = 0.66 (có 2 người không có triệu chứng sổ mũi trong 3 người không bị cúm)

P(Headache = “Mild” | Flu = “Y”) = = 0.4 (có 2 người có triệu chứng đau đầu nhẹ trong 5 người bị cúm)

P(Headache = “Mild” | Flu = “N”) = = 0.33 (có 1 người có triệu chứng đau đầu nhẹ trong 3 người không bị cúm)

P(Fever = “N” | Flu = “Y”) = = 0.2 (có 1 người không có triệu chứng sốt trong 5 người bị cúm)

P(Fever = “N” | Flu = “N”) = = 0.66 (có 2 người không có triệu chứng sốt trong 3 người không bị cúm)

**Tính P(X):**

P(Chills = “Y”) = = 0.5 (có 4 người bị ớn lạnh trên 8 người khám triệu chứng ớn lạnh)

P(Runny nose = “N”) = = 0.375 (có 3 người không bị sổ mũi trên 8 người khám triệu chứng sổ mũi)

P(Headache = “Mild”) = = 0.375 (có 3 người bị đau đầu nhẹ trên 8 người khám triệu chứng đau đầu nhẹ)

P(Fever = “N”) = = 0.375 (có 3 người không bị sốt trên 8 người khám triệu chứng sốt)

P(X) = (Xác suất của mẫu X)

= = 0.026

Xác suất của mẫu thử X có bị cúm:

P(X | Flu = “Y”) =

P(Headache = “Mild” | Flu = “Y”)

P(Fever = “N” | Flu = “Y”)

P(Flu = “Y”)

= 0.6 0.2 0.4 0.2 0.625 = 0.006

Xác suất của mẫu thử X không bị cúm:

P(X | Flu = “N”) =

P(Headache = “Mild” | Flu = “N”)

P(Fever = “N” | Flu = “N”)

P(Flu = “N”)

= 0.33 0.66 0.33 0.66 0.375 = 0.018

Vì P(X | Flu = “Y”) < P(X | Flu = “N”) nên mẫu X là không bị cúm.

Chúng ta có thể ứng dụng Naive Bayes Classification để tính tỷ lệ xác suất với rất nhiều các dạng bài toán khác nhau, với dữ liệu càng nhiều thì độ chính xác của thuật toán sẽ càng cao, và khi dữ liệu thay đổi thì kết quả cũng thay đổi theo.

**2.3. Ứng dụng**

Thuật toán Naive Bayes Classification được áp dụng vào các loại ứng dụng sau:

**Real time Prediction:** NBC chạy khá nhanh nên nó thích hợp áp dụng ứng dụng nhiều vào các  ứng dụng chạy thời gian thực, như hệ thống cảnh báo, các hệ thống trading …

**Multi class Prediction:**Nhờ vào định lý Bayes mở rộng ta có thể ứng dụng vào các loại ứng dụng đa dự đoán, tức là ứng dụng có thể dự đoán nhiều giả thuyết mục tiêu.

**Text classification/ Spam Filtering/ Sentiment Analysis:**NBC cũng rất thích hợp cho các hệ thống phân loại văn bản hay ngôn ngữ tự nhiên vì tính chính xác của nó lớn hơn các thuật toán khác. Ngoài ra các hệ thống chống thư rác cũng rất ưu chuộng thuật toán này. Và các hệ thống phân tích tâm lý thị trường cũng áp dụng NBC để tiến hành phân tích tâm lý người dùng ưu chuộng hay không ưu chuộng các loại sản phẩm nào từ việc phân tích các thói quen và hành động của khách hàng.

**Recommendation System:**Naive Bayes Classifier và Collaborative Filtering được sử dụng rất nhiều để xây dựng cả hệ thống gợi ý, ví dụ như xuất hiện các quảng cáo mà người dùng đang quan tâm nhiều nhất từ việc học hỏi thói quen sử dụng internet của người dùng, hoặc như ví dụ đầu bài viết đưa ra gợi ý các bài hát tiếp theo mà có vẻ người dùng sẽ thích trong một ứng dụng nghe nhạc …

**CHƯƠNG 3**

**GIẢI QUYẾT BÀI TOÁN LỌC THƯ RÁC**

**3.1. Đặt vấn đề**

Thư rác hay còn gọi là Spam mail là những thư điện tử không cần thiết đối với người sử dụng cụ thể nào đó, những thư này thường được gửi tới nhiều người và thường mang nội dung quảng cáo, chất lượng thấp, đôi khi còn mang những thông tin lừa đảo. Các thư rác có thể vô hại đối với người sử dụng nhưng sẽ gây cảm giác khó chịu khi những thư này lẫn với những thư cần thiết.

Ngày nay có rất nhiều công nghệ lọc thư rác. Nó dựa trên các đặc trưng cơ bản của thư điện tử như: tiêu đề thư, địa chỉ người gửi, hay các cụm từ thường xuất hiện trong các thư rác. Hoặc cũng có thể so sánh các thư điện tử mẫu với các thư điện tử nhận được sau đó tìm ra những thư rác.

Như vậy, giải pháp ngăn chặn thư rác nào là hiệu quả và dùng được lâu dài. Một phương pháp tốt nhất đó là để chính người dùng thư điện tử ngăn chặn thư rác, bới họ hiểu vấn đề một cách tường minh nhất. Chúng ta sẽ dùng cảm nhận về thư rác của mỗi người để huấn luyện cho các bộ lọc thư rác của chính họ. Mỗi bộ lọc sẽ xử lý thư rác tùy theo phong cách của từng người dùng thư điện tử. Và mô hình thống kê Bayes được áp dụng để thực hiện ý tưởng này.

Từ những đặc điểm trên, ta thấy rằng việc xây dựng được một bộ lọc thư rác thông minh có thể loại bỏ một cách chính xác hiện nay là một nhiệm vụ còn nhiều thách thức.

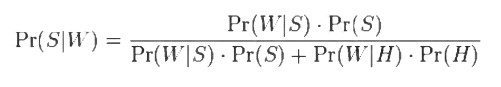
**3.2. Bài toán**

Thư điện tử là một trong những phương tiện để giao tiếp đáng tin cậy và hầu như không tốn kém chi phí sử dụng. Phạm vi của nó sử dụng rộng khắp trên toàn thế giới và có thể dễ dàng truy cập bằng hầu hết các phương tiện truyền thông đã biến nó thành nạn nhân của những kẻ spam. Hậu quả đơn giản nhất là làm tốn băng thông mạng và nghiêm trọng hơn là làm mất thời gian của người dùng thư điện tử, làm lan truyền vi rút máy tính. Có thời điểm người ta thống kê được rằng có đến 60% thưu điện tử là thư rác và mỗi ngày một người dùng thư điện tử phải nhận ít nhất là 6 cú spam.

Chúng ta không thể đổi địa chỉ hòm thư mỗi lần bị spam bởi điều này không những không hạn chế được thư rác mà có khi còn làm cho nó gia tăng. Vậy cần phải tìm ra một giải pháp chống thư rác sử dụng bộ lọc được gắn thuật toán phân loại với tính năng hiệu quả và kĩ thuật đơn giản dễ cài đặt. Và một yêu cầu không thể thiếu là có làm sao với thuật toán đó nhwungx kẻ spam hiểu rằng việc chúng cố tình spam là vô dụng.

**3.3. Dùng luật Bayes tính xác suất**

Tính xác suất cho mỗi thẻ ta dùng luật Bayes để tính. Giả sử ta cần tính xác suất cho thẻ chứa tò *‘promotion* ’.Từ này chúng ta thường xuyên gặp trong thư điện tử mời chào dịch vụ maketing. Công thức tính theo luật Bayes:



Trong đó:

* *Pr(S\ W)* là xác suất mà thư mà chứa từ *‘promotion ’*là thư rác
* *Pr(S)* là xác suất mà thư bất kì là thư rác
* *P(W \S)* là xác suất mà từ "*promotion*" xuất hiện trong thư rác
* *Pr(H)* là xác suất mà một bản tin bất ki không là thư rác
* *P(W \H)* là xác suất mà từ "*promotion*" xuất hiện trong thư rác

Như đã nói ở trên, những thống kê gần đây cho thấy 80% thư điện tử là thư rác nên ta sẽ có:



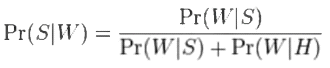
Tuy nhiên để cho đơn giản và đã qua thực tế nên người ta chọn các xác suất

trước là giống nhau và đều có giá trị bằng 0.5. Tức là:



Bộ lọc mà dùng giả thiết này được gọi là *"không đổi xứng*", có nghĩa rằng chúng không có sự đối xử phân biệt các thư đến. Giả thiết này cho phép rút gọn công

thức ở trên thành:

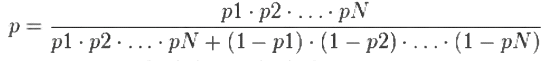


Bộ lọc thư rác Bayesspam vận dụng chính xác công thức trên để tính xác suất

cho mỗi từ đơn.

Sau khi đã tính được xác suất thư chứa từ đơn là thư rác ta cần kết hợp các xác suất đơn đó lại thành một xác suất cuối cùng. Xác suất này dùng để đánh giá thư mà chứa tất cả các từ đơn đó có xác suất là thư rác là bao nhiêu. Công thức tính xác suất

kết hợp là:



Trong đó:

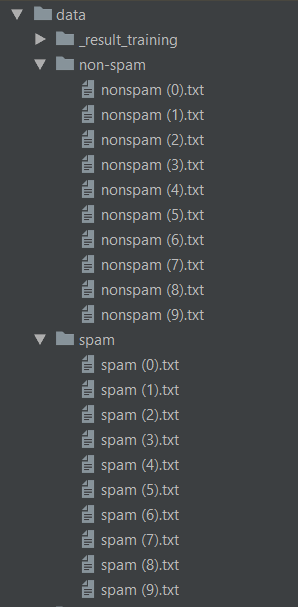
* *p* là xác suất thư đang xét là thư rác
* *pl* là xác suất*p(S\ W1),* ứng với từ đầu tiên (ví dụ từ "*promotion*")
* *p2* là xác suất*p(S\ W2)* , ứng với từ thứ hai (ví dụ tò "*offer*")
* *...*
* *pN* là xác suất p(S|W N), ứng với từ thứ N (ví dụ tò "*home*")

Kết quả *p* thường được dùng so sánh với một ngưỡng nào đó để quyết định thư đang xét có xác suất *p* đó có là thư rác hay không. Neu *p* lớn hơn giá trị ngưỡng, thư đó sẽ bị đánh dấu là thư rác, ngược lại sẽ không bị đánh dấu là thư rác.

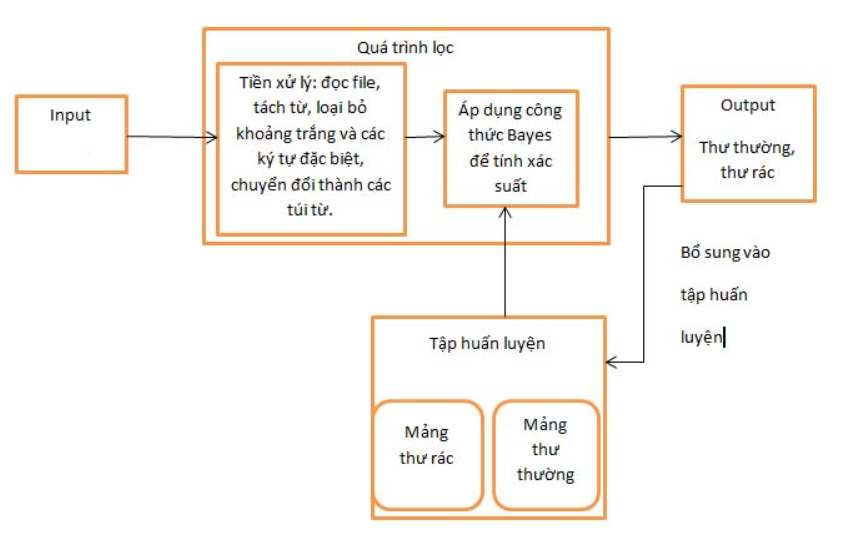
## **3.4. Huấn luyện cho bộ lọc Bayes**

**Dữ liệu ban đầu**

Ban đầu chuẩn bị 1 tập huấn luyện gồm 10 file text đánh dấu là spam và 10 file text đánh dấu là không spam (nếu bạn có càng nhiều file càng tốt). Tỉ lệ ở đây là 50:50 tức là nhận 1 email tới thì khả năng nó là spam là 50%.



Sau mỗi lần kiểm tra được đó có phải là email spam không ta lại thêm nó vào tập huấn luyện và tỉ lệ sẽ khác dần đi. Ví dụ nhận được 10 email tiếp theo đều không phải là spam thì tỉ lệ spam/non-spam sẽ là khoảng 30:70 như thế 1 email mới sẽ có khả năng spam là 30%.



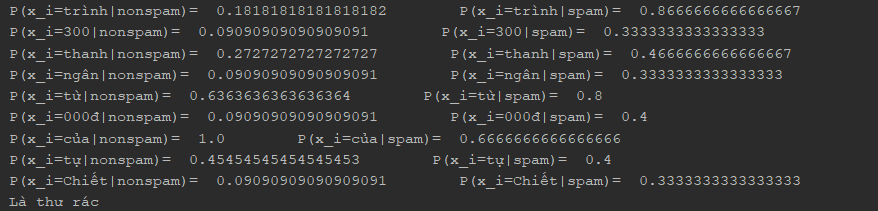
**Chạy tập huấn luyện**

Mình sẽ duyệt từng file spam và tách thành các túi từ (mỗi túi từ 1 Set các từ phân biệt nhau). Sau khi duyệt xong 10 file ta sẽ được 1 List gồm 10 túi từ spam. Tương tự duyệt và tác các file non-spam ta cũng được 1 list gồm 10 túi từ non-spam.

Sau khi tách được 2 List túi từ ta ghi nó vào file result\_training.dat để lúc thực hiện kiểm tra mail spam thì chỉ việc đọc từ file này ra chứ không cần chạy huấn luyên nữa.

package stackjava.com.filtermailspam;  
  
import java.io.File;  
import java.io.FileInputStream;  
import java.io.FileNotFoundException;  
import java.io.FileOutputStream;  
import java.io.IOException;  
import java.io.ObjectInputStream;  
import java.io.ObjectOutputStream;  
import java.util.ArrayList;  
import java.util.Set;  
  
import org.apache.commons.io.FileUtils;  
  
public class Demo {  
  
 // mảng chứa các túi từ của thư thường (non-spam)  
 static ArrayList<Set<String>> *listBagOfNonSpam* = new ArrayList<>();  
 // mảng chứa các túi từ của thư rác (spam)  
 static ArrayList<Set<String>> *listBagOfSpam* = new ArrayList<>();  
  
 // tinh xac xuat P(xi=x|nhan= nonspam)  
 public static double pNonSpam(String x) {   
 double k = 0;  
 for (int i = 0; i < *listBagOfNonSpam*.size(); i++) {  
 // moi lan x xuat hien trong 1 thu thuong thi k++  
 if (*listBagOfNonSpam*.get(i).contains(x))  
 k++;  
 }  
 return (k + 1) / (*listBagOfNonSpam*.size() + 1);  
 // P(xi|nhan= nonspam)= (k+1)/(sothuthuong+1);  
 // trong do: k la so cac mail nonspam xuat hien xi  
 // sothuthuong la so mail nonspam  
  
 }  
  
 // tinh xac xuat P(xi=x|nhan= spam)  
 public static double pSpam(String x) {  
 double k = 0;  
 for (int i = 0; i < *listBagOfSpam*.size(); i++) {  
 if (*listBagOfSpam*.get(i).contains(x))  
 // moi lan x xuat hien trong 1 thu rac thi k++  
 k++;  
 }  
 return (k + 1) / (*listBagOfSpam*.size() + 1);  
 // P(xi|nhan= spam)= (k+1)/(sothurac+1);  
 // trong do: k la so cac mail spam xuat hien xi  
 // sothurac la so mail spam  
 }  
  
 @SuppressWarnings("unchecked")  
 public static void main(String[] args) throws FileNotFoundException, IOException, ClassNotFoundException {  
  
 // đọc dữ liệu huấn luyện từ trước ở trong file result\_training.dat ra  
 System.*out*.println("Bắt đầu load dữ liệu huấn luyện");  
 ObjectInputStream inp = new ObjectInputStream(  
 new FileInputStream(new File("data/\_result\_training/result\_training.dat")));  
 *listBagOfSpam* = (ArrayList<Set<String>>) inp.readObject();  
 *listBagOfNonSpam* = (ArrayList<Set<String>>) inp.readObject();  
 inp.close();  
 System.*out*.println("Hoàn load dữ liệu huấn luyện");  
  
 // đọc dữ liệu từ mail cần kiểm tra  
 System.*out*.println("Đọc dữ liệu mail cần kiểm tra");  
 File mailTesting = new File("data/test/test (1).txt");  
 // Tiền xử lý mail cần kiểm tra  
  
 String mailData = FileUtils.*readFileToString*(mailTesting, "UTF-16");  
 Set<String> bagOfTest = RunTrainingData.*toBagOfWord*(mailData);  
  
 System.*out*.println("Bắt đầu kiểm tra:");  
  
 // xác xuất là thư thường. P(xi|non-spam)  
 double C\_NB1 = *listBagOfNonSpam*.size() / ((double) *listBagOfNonSpam*.size() + *listBagOfSpam*.size());  
 // xác xuất là thư rác. P(xi|spam)  
 double C\_NB2 = *listBagOfSpam*.size() / ((double) *listBagOfNonSpam*.size() + *listBagOfSpam*.size());  
  
 ArrayList<String> listStringTest = new ArrayList<>(bagOfTest);  
  
 for (String strTest : listStringTest) {  
 if (*pNonSpam*(strTest) != ((double) 1 / (*listBagOfNonSpam*.size() + 1))  
 || *pSpam*(strTest) != ((double) 1 / (*listBagOfSpam*.size() + 1))) {  
  
 System.*out*.println("P(x\_i=" + strTest + "|nonspam)= " + *pNonSpam*(strTest) + " " + "P(x\_i="  
 + strTest + "|spam)= " + *pSpam*(strTest));  
 C\_NB1 \*= *pNonSpam*(strTest);  
 C\_NB2 \*= *pSpam*(strTest);  
 }  
 }  
 if (C\_NB1 < C\_NB2) {  
 // Bổ sung thư vừa kiểm tra vào tập huấn luyện.  
 *listBagOfSpam*.add(bagOfTest);  
 System.*out*.println("Là thư rác");  
 } else {  
 *listBagOfNonSpam*.add(bagOfTest);  
 System.*out*.println("Là thư thường");  
 }  
  
 // Lưu lại tập huấn luyện mới.  
 ObjectOutputStream out = new ObjectOutputStream(  
 new FileOutputStream(new File("data/\_result\_training/result\_training.dat")));  
 out.writeObject(*listBagOfSpam*);  
 out.writeObject(*listBagOfNonSpam*);  
 out.close();  
 System.*out*.println("Kết thúc");  
 }  
  
}

**Kết quả:**



# **KẾT LUẬN**

Đồ án đã tập trung nghiên cứu về lý thuyết Bayes, từ bước cơ sở đó tìm hiểu tiếp về một ứng dụng của nó liên quan trực tiếp đến ngành công nghệ thông tin đó là ứng dụng lọc thư rác. Quá trình tìm hiểu về nguyên lý và cách thức hoạt động củabộ lọc đã rút ra được những kết luận về ưu nhược điểm của tiếp cận thống kê Bayes trong việc phân loại thư rác. Đối với vấn đề ứng dụng thực tế, khoá luận sử dụng plugin BayesSpam như một đối tượng chính để tìm hiểu và nghiên cứu. Đối với vấn đề áp dụng lý thuyết Bayes, khoá luận nghiên cứu xây dựng các công thức tính xác suất sao cho việc xử lý thông tin trở lên nhanh gọn và có độ chính xác cao.

Từ việc tìm hiểuứng dụng BayesSpam, khoá luận đã rút ra được một số nhận định về ưu điểm và nhược điểm của bộ lọc trong quá trình hoạt động. Kết quả phân loại thư rác nhìn chung là gần giống với các kết quả đánh giá thư bởi người dùng.

Tuy nhiên, do thời gian có hạn cũng như các kiến thức chuyên môn về hệ thống thư điện tử nên các kết luận rút ra được trong quá trình nghiên cứu còn nhiều hạn chế. Dưới đây là những ưu nhược điểm chính của bộ lọc thư rác Bayes.