TRƯỜNG ĐẠI HOC BÁCH KHOA HÀ NỘI
VIỆN CNTT&TT

BÁO CÁO BÀI TẬP LỚN

Đề tài : Phân loại thư rác sử dụng Naïve Bayes

GVHD : TS Nguyễn Nhật Quang

SVTH : Đoàn Tiến Huy Hoàng 20173130
Đoàn Văn Lợi 20173241
Phạm Thế Tài 20173351
Nguyễn Văn Toàn 20173405

Hà Nội, Tháng 5 năm 2020

Lời mở 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, nhóm em chọn hướng nghiên cứu "Phân loại thư rác bằng phương pháp phân loại Naïve Bayes".

Chương I: Tổng quan về thư rác

1 Khái quát về thư rác

Thư rác hay còn gọi là spam mail là các thư điện tử vô bổ thường chứa các loại quảng cáo được gửi một cách vô tội vạ và chất lượng của loại thư này thường thấp. Đôi khi, nó dẫn dụ người nhẹ dạ, tìm cách đọc số thẻ tín dụng và các tin tức cá nhân của họ.

Có thể nói thư rác là một hình thức "tra tấn người dùng thư điện tử" bằng quảng cáo. Các thư rác có thể vô hại nhưng mỗi ngày một người có thể vì các thư rác này mà bị đầy cả hộp .Có thể chúng ta sẽ thắc mắc tại sao người ta lại lặp đi lặp lại một cái thư quảng cáo cả chục lần cho một người, cũng đơn giản là vì họ muốn dùng hiệu ứng tâm lý… Khi hình ảnh sản phẩm nào đó cứ đập vào mắt mình mãi thì đến lúc cần mua một thứ có chức năng tương tự (hay cùng loại) thì chính hình ảnh thương hiệu của cái thư rác sẽ hiện lên trong óc chúng ta trước tiên.

Như vậy, theo định nghĩa thì các thư rác có thể có hại cho máy tính (hiểu theo nghĩa vật chất), đôi khi còn làm chúng ta bực mình khó chịu hoặc làm cho các thư từ khác (nhất là các thư gửi có nghĩa quan trọng) bị lẫn lộn trong một đống thư mà chủ yếu là các thư rác. Khiến cho việc tìm kiếm cũng mất thời gian và cũng có thể khi xóa thư rác lại xóa nhầm thư quan trọng.

2 Các đặc trưng của thư rác

Các loại thư rác hiện nay có một số đặc điểm sau :

- Thư rác được gửi đi một cách tự động
- Thư rác được gửi đến những địa chỉ ngẫu nhiên trên một diện rộng.
- Nội dung thư rác thường là những nội dung bất hợp pháp, gây phiền hà

cho người dùng.

- Địa chỉ của những người gửi thư rác thường là những địa chỉ trá hình.

3 Tác hại của thư rác

Theo thống kê thư rác hiện chiếm hơn một nửa số email truyền trên internet và chính thư rác là nguồn lây lan virus nhanh nhất. Thiệt hại do chúng gây ra rất lớn đối với sự phát triển internet nói chung và người sử dụng thư điện tử nói riêng.

Theo thống kê toàn cầu của hãng nghiên cứu Ferris Research ở San Francisco, thư rác gây thiệt hại 50 tỷ USD trong năm 2005. Chỉ tính riêng ở Mỹ, thiệt hại do thư rác gây ra đối với các doanh nghiệp ước tính khoảng 17 tỷ USD/năm. Thư rác chiếm khoảng 80% lưu lượng thư điện tử thế giới trong quý 1/2006, đó là kết luận của nhóm hợp tác chống thư rác gồm các công ty AOL, Bell Canada, Cigular Wireless, EarthLink, France Telecom, Microsoft, Verizon, và Yahoo. Microsoft và AOL cho biết hai hãng này trung bình mỗi ngày chặn gần 5 tỷ thư rác. Ước tính, cứ 9 trong 10 email sử dụng dịch vụ MSN Hotmail của Microsoft là thư rác.

Tại Việt Nam, tình hình thư rác cũng đang rất phức tạp. Công ty Điện toán và Truyền số liệu (VDC) - ISP lớn nhất Việt Nam - cho biết, thư rác hiện nay chiếm phần lớn lưu lượng email qua hệ thống máy chủ thư của ISP này.

Không chỉ gây thiệt hại về tiền bạc, thư rác còn làm giảm hiệu quả làm việc, gây stress, tiêu tốn thời gian của nhân viên... Những điều này cũng đồng nghĩa với việc, năng suất lao động giảm, ảnh hưởng tới tình hình kinh doanh và doanh thu của công ty.

Số lượng email spam vẫn luôn luôn tăng và ngày càng tinh vi hơn, người ta nhận định rằng việc chống email spam sẽ luôn luôn phải thực hiện, tùy vào ý thức của cư dân internet và sức mạnh của công nghệ mà việc email spam chỉ được hạn chế phần nào.

4 Các phương pháp lọc thư rác

4.1 Naïve Bayes

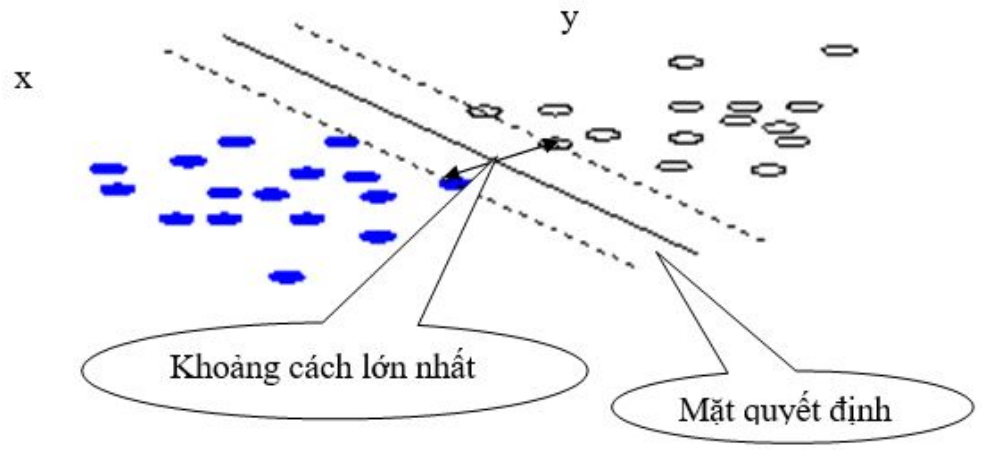
Kỹ thuật phân hoạch của Naive Bayes dựa trên cơ sở định lý Bayes và đặc biệt phù hợp cho các trường hợp phân loại có kích thước đầu vào là lớn. Mặc dù Naive Bayes khá đơn giản nhưng nó có khả năng phân loại tốt hơn rất nhiều phương pháp phân hoạch phức tạp khác. Với mỗi loại văn bản, thuật toán Naive Bayes tính cho mỗi lớp văn bản một xác suất mà tài liệu cần phân hoạch có thể thuộc loại đó. Tài liệu đó sẽ được gán cho lớp văn bản nào có xác suất cao nhất.

Tại sao lại chọn Phân loại Naive Bayes :

Naïve Bayes có khả năng học tập và dự đoán hiệu quả cao, nó thường được sử dụng để so sánh với các phương pháp phức tạp hơn vì nó nhanh và có khả năng mở rộng cao .

4.2 SVM

SVM là một phương pháp tiếp cận gần đúng thường áp dụng để phân loại hai lớp văn bản. Phương pháp này được xác định dựa trên một không gian vector mà trong không gian này vấn đề phải tìm ra một "mặt quyết định" giữa hai lớp sao cho khoảng cách giữa các điểm dữ liệu giữa hai lớp này là lớn nhất.



Hình 1: Ví dụ minh họa bằng phương pháp SVM

Nếu văn bản cần phân loại nằm về phía nào của mặt quyết định thì nó được phân vào loại văn bản đó. Phương pháp SVM thường áp dụng để phân loại hai lớp văn bản nhưng vẫn có thể áp dụng để phân loại nhiều lớp văn bản.

4.3 Cây quyết định

Cây quyết định là một trong những kĩ thuật học máy được nhiều người biết đến. Chúng được áp dụng rộng rãi và đa dạng của các lĩnh vực đặc biệt là các ứng dụng trong trí tuệ nhân tạo. Thành công của phương pháp này được chứng minh bằng khả năng xử lý các vấn đề phức tạp theo cách trình bày một khả năng có thể chấp nhận được dễ dàng cho việc diễn dịch và thậm chí còn có khả năng đưa ra các kết luận từ các luật logic. Một số phương pháp đã được đề xuất để xây dựng các cây quyết định. Những thuật toán này có đầu vào là một tập các tài liệu mẫu được đưa vào làm ví dụ. Ở đó mỗi tài liệu được mô tả bởi việc thiết lập các giá trị thuộc tính và được gán cho các lớp. Đầu ra là một cây quyết định bảo đảm sự phân hoạch cho các ví dụ đưa vào. Vấn đề chính được nói rõ trong các thuật toán chuẩn của cây quyết định là kết quả có

thể thay đổi bất chợt tùy thuộc vào dữ liệu. Sự không chắc chắn đó có thể xuất hiện trong các cấu tử hoặc có thể xuất hiện trong các giai đoạn phân hoạch.

Cây quyết định được sử dụng để phân hoạch các đối tượng mới. Thuật giải trong cây quyết định được bắt đầu từ gốc của cây quyết định. Chúng ta sẽ đánh giá thử những thuộc tính có liên quan và chọn một nhánh tương ứng với sự lựa chọn của đó. Quy trình này sẽ được lặp đi lặp lại đến khi gặp phải một lá. Như vậy đối tượng mà ta đang xét sẽ thuộc vào loại của lá mà ta vừa gặp phải. Điều đó cũng có nghĩa là thuật toán cây quyết định kết thúc khi mà quá trình phân tích gặp được một nút lá.

Chương II: Thuật toán Phân loại Naïve Bayes

1 Cơ sở lý thuyết

1.1 Công thức xác suất có điều kiện

Xác suất điều kiện của biến cố A với điều kiện biến cố B đã xảy ra là một số không âm, kí hiệu là P(A|B) : Biểu thị khả năng xảy ra biến cố A trong tình huống biến cố B đã xảy ra

$$P(A|B) = \frac{P(AB)}{P(B)}$$

1.2 Công thức xác suất đầy đủ

Giả sử B_1 , B_2 , ..., B_n là nhóm đầy đủ các biến cố. Xét biến cố A sao cho A xảy ra chỉ khi một trong các biến cố B_1 , B_2 , …, B_n xảy ra. Khi đó

$$P(A) = \sum_{i=1}^{n} P(B_i) . P(A|B_i)$$

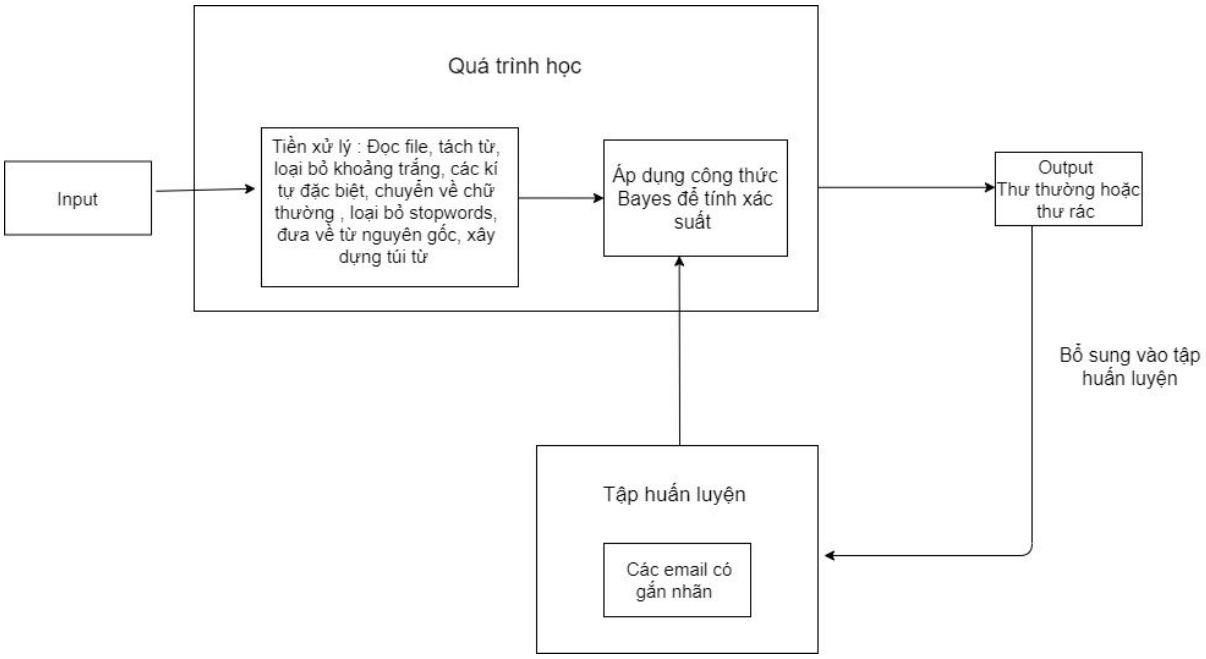
Cơ sở của phương pháp phân loại văn bản Naive Bayes là chủ yếu dựa trên các giả định của Bayes. Giả thuyết Bayes gán cho mỗi tài liệu văn bản cần phân loại một giá trị xác suất.

$$P(h|D) = \frac{P(D|h).P(h)}{P(D)}$$

- P(h) : Xác suất trước (tiên nghiệm) của giả thiết(Phân loại) h
- P(D) Xác suất trước (tiên nghiệm) của việc quan sát được dữ liệu D
- P(D|h) : Xác suất (có điều kiện) của việc quan sát được dữ liệu D , nếu giả thiết phân loại h là đúng
- P(h|D) : Xác suất (có điều kiện) của giả thiết (phân loại) h là đúng nếu quan sát được dữ liệu D

2 Chi tiết về phương pháp học máy

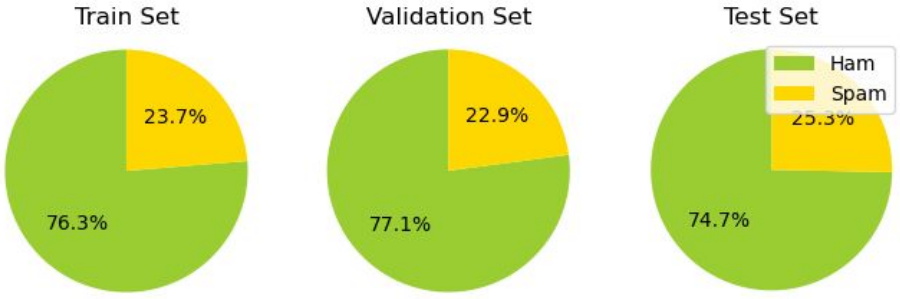
Sau khi tham khảo từ một số bài viết tương tự về chủ đề này nhóm em đã thống nhất và đưa ra mô hình chung cho bài toán như sau:



Hình 2 : Quy trình

2.1 Tiền xử lý dữ liệu, chọn các thuộc tính đặc trưng

Tập dữ liệu chúng em thu thập được trích từ nguồn kaggle .Áp dụng vào bài toán với phân bố lớp như hình vẽ dưới



Hình 3 : Phân bố lớp

2.1.1 Tiền xử lý dữ liệu

2.1.1.1 Xử lý Stopwords

Trong một văn bản những từ thực sự mang ý nghĩa tương đối ít trong khi đó những từ nối, tân ngữ hay mạo từ chỉ làm cho văn bản trôi chảy mà không có ý nghĩa đối với chủ đề của bài viết. Những từ stopword (từ dừng) lại có số lượng lớn hơn rất nhiều so với các từ chính mang ý nghĩa. Việc loại bỏ các từ này giúp cho bộ phân loại tốt hơn.

Với mô hình bài toán phân loại thư chúng em sử dụng thư viện NLTK(Natural language toolkit) để loại bỏ các từ này. Ngoài ra chúng em còn loại bỏ các ký tự đặc biệt, dấu câu và chuyển các từ in hoa về từ thường.

2.1.1.2 Xử lý đưa các từ về nguyên gốc

Mỗi một loại về ngôn ngữ thì có những đặc trưng rất riêng và những đặc điểm này có thể ảnh hưởng rất nhiều đến bộ phân loại. Cụ thể trong bài này với văn bản đầu vào là các email bằng tiếng Anh thì các từ có thể có rất nhiều dạng biểu diễn nhưng cùng một nghĩa. Ví dụ như cùng mang ý nghĩa là "good" nhưng có các dạng biểu diễn như: "well", "better", ...

Thư viện NLTK có cung cấp các hàm để xử lý các vấn đề này. Các từ sau khi được xử lý thì sẽ được đưa về dạng nguyên gốc.

2.1.2 Chọn các thuộc tính đặc trưng

Lựa chọn các đặc trưng văn bản là bước đầu tiên trong một bài toán phân loại văn bản nói chung và bài toán phân loại thư nói riêng. Đây là tiền đề quan trọng để có thể học được một mô bộ phân loại tốt. Có rất nhiều phương pháp để lấy ra các đặc trưng của văn bản nhưng một trong các phương pháp thường được sử dụng là TF-IDF.

2.1.2.1 TF-IDF

Trọng số của một từ trong tập các văn bản sẽ được đánh giá trên 2 phương diện sau:

TF: Term Frequency(Tần suất xuất hiện của từ) là số lần từ xuất hiện trong văn bản. Vì các văn bản có thể có độ dài ngắn khác nhau nên một số từ có thể xuất hiện nhiều lần trong một văn bản dài hơn là một văn bản ngắn. Như vậy, term frequency thường được chia cho độ dài văn bản(tổng số từ trong một văn bản).

$$\mathrm{tf}(t,d) = \frac{\mathrm{f}(t,d)}{\max\{\mathrm{f}(w,d) : w \in d\}}$$

Trong đó:

- tf(t, d): tần suất xuất hiện của từ t trong văn bản d
- f(t, d): Số lần xuất hiện của từ t trong văn bản d

- max({f(w, d) : w ∈ d}): Số lần xuất hiện của từ có số lần xuất hiện nhiều nhất trong văn bản d

IDF: Inverse Document Frequency(Nghịch đảo tần suất của văn bản), giúp đánh giá tầm quan trọng của một từ . Khi tính toán TF , tất cả các từ được coi như có độ quan trọng bằng nhau. Nhưng một số từ như "is", "of" và "that" thường xuất hiện rất nhiều lần nhưng độ quan trọng là không cao. Như thế chúng ta cần giảm độ quan trọng của những từ này xuống.

$$\mathrm{idf}(t, D) = \log \frac{|D|}{|\{d \in D : t \in d\}|}$$

Trong đó:

- idf(t, D): giá trị idf của từ t trong tập văn bản
- |D|: Tổng số văn bản trong tập D
- |{d ∈ D : t ∈ d}|: thể hiện số văn bản trong tập D có chứa từ t.

2.1.2.2 Tần số tài liệu

Tần suất tài liệu DF là là **số tài liệu có sự xuất hiện của một từ** (term). Người ta đã tính toán tần suất tài liệu cho một từ đơn trong tập văn bản mẫu. Cốt lõi của phương pháp này là phải tìm ra được một không gian các từ đặc trưng. Với không gian này thì các từ phổ biến (xuất hiện thường xuyên trong mọi văn bản) đã được loại bớt, và cả những từ chỉ xuất hiện một lần (từ loại hiếm) cũng không được tính vào không gian các từ đặc trưng đó. Cách xác định DF là kĩ thuật đơn giản nhất để làm giảm bớt vốn từ có trong văn bản. Mặc dù đối với các văn bản lớn phương pháp này chỉ đạt đến độ phức tạp tuyến tính (các giá trị DF tính được thường nhỏ hơn trong thực tế) nhưng nó vẫn được coi là phép tính gần đúng để cải tiến hiệu quả của thuật toán**.** Các bước trong phương pháp lựa chọn này bao gồm:

- Tính DF của các từ trong văn bản
- Sắp xếp theo chiều giảm dần các DF
- Loại bỏ từ phổ biến và từ hiếm
- Chọn các đặc trưng có DF lớn: muốn thực hiện công việc này người ta phải định ra một ngưỡng(coi là giới hạn để chọn từ)

Với phương pháp này từ loại nào xuất hiện trong càng nhiều văn bản càng có giá trị và càng có thể được chọn vào không gian đặc trưng của tài liệu đang xét đó

Tóm lại: để phân loại văn bản cần dự vào các đặc trưng xác định. Các đặc trưng đó rất khó bị thay đổi. Có rất nhiều kĩ thuật lựa chọn tuy nhiên phương pháp TF-IDF tương đối đơn giản và thu được kết quả tốt.

Trong bài làm của mình của nhóm em đã sử dụng việc tạo túi từ từ việc loại bỏ các từ ít xuất hiện thông qua DF và lấy tập các từ thông qua trọng số TF-IDF. Việc này giúp loại bỏ các từ là tên riêng và các từ hiếm gặp.

2.2 Phương pháp thực hiện

Giả thiết mỗi một email được đại diện bởi một vector thuộc tính đặc trưng là x = (x_1 , x_2 , …, x_n) với x_1 , x_2 , …, x_n là giá trị các thuộc tính x_1 , x_2 , …, x_n tương ứng trong không gian vector đặc trưng x

Dựa vào công thức xác suất Bayes và công thức xác suất đầy đủ ta có được xác suất 1 email với vector đặc trưng x thuộc về loại c là :

$$P(C=c|X=x) = \frac{P(C=c)*P(X=x|\ C=c)}{\sum P(C=k)*P(X=x|\ C=k)}$$

với C là email được xét , c ={spam, non-spam}

Xác suất P(C=c) được tính dễ dàng từ tập huấn luyện.Thực tế rất khó để tính được xác suất P(X=x | C=c). Giả thiết rằng tất cả các biến cố x1,x2,...xn là độc lập với nhau do đó chúng ta có thể tính được xác suất P(X=x| C=c) dựa theo công thức :

$$P(X=x|\ C=c) = \prod_{i=1}^{n} P(Xi = xi \mid C = c\ )$$

Như vậy công thức tính xác suất một email là spam sẽ được viết thành :

$$P(C=c|X=x) = \frac{P(C=c\ ).\ \prod_{i=1}^{n} P(Xi=xi \mid C=c)}{\sum P(C=k).\ \prod_{i=1}^{n} P(Xi=xi \mid C=k)}$$

Để đánh giá một email ta phải chuyển mỗi một email sang một vector x = (x_1 , x_2 , …, x_n) với x_1 , x_2 , …, x_n là giá trị các thuộc tính X_1 , X_2 , …, X_n trong không gian vector đặc trưng X. Mỗi thuộc tính được thể hiện bởi một từ. Theo phương pháp đơn giản nhất ta có thể lập ra một từ điển chứa các token. Sau đó với mỗi token trong email nếu nó xuất hiện trong từ điển thì giá trị thuộc tính sẽ là 1, ngược lại thì là 0. Tuy nhiên trên thực tế, tập huấn luyện của ta không thường là một bộ từ điển như vậy. Thay vào đó tập huấn luyện lúc này sẽ gồm có hai kho ngữ liệu. Kho ngữ liệu Spam sẽ chứa một list các email đã được xác định là spam trước đó, và tương tự với kho ngữ liệu Ham sẽ chứa các email hợp lệ.

Như vậy nếu ta vẫn để giá trị các thuộc tính là 0 hoặc 1 thì sẽ rất khó đánh giá được một email là spam hay không. Đặc biệt nếu email nhận được là dài, khi đó nếu ta vẫn sử dụng giá trị thuộc tính là 0 hoặc 1 thì sự xuất hiện của một token 100 lần cũng tương đương với việc xuất hiện chỉ 1 lần.

Để khắc phục vấn đề này giá trị thuộc tính bây giờ ta sẽ thay bằng xác suất spam của token đó. Xác suất này tương đương với xác suất spam của một email chỉ chứa token đó và là email spam. Việc tính xác suất này thì có nhiều phương pháp. Ta có thể tính dựa trên số lần xuất hiện của token này trong mỗi kho ngữ liệu học ban đầu. Ví dụ một token w có số lần xuất hiện trong kho ngữ liệu spam là s và non-spam là n, số email tổng cộng ở kho spam và non-spam tương ứng là Ns và Nn thì xác suất spam của token w này sẽ là :

$$\mathbf{P(X=w \mid C=spam)} = \frac{s/Ns}{s/Ns+n/Nn}$$

Tuy nhiên nhược điểm của phương pháp này khả năng spam của một token xuất hiện 100 lần ở 100 email khác nhau là bằng với khả năng spam của một token xuất hiện 100 lần chỉ ở trong một email.

Thay vào việc tính xác suất này dựa theo số lần xuất hiện của token trong từng kho ngữ liệu ta có thể dựa vào số email chứa token trong từng kho ngữ liệu. Ví dụ một

token w có số email chứa nó trong kho ngữ liệu spam và non-spam là ns và nn thì xác suất spam của token w này sẽ là :

$$\mathbf{P(X=w \mid C=spam)} = \frac{s/Ns}{ns/Ns+nn/Nn}$$

Nhược điểm của phương pháp này là khả năng spam của một token xuất hiện 1 lần trong một email là bằng với khả năng spam của một token xuất hiện 100 lần trong một email.

Vì vậy chúng ta sử dụng cách thứ ba là tổng hợp của hai cách trên :

$$\mathbf{P(X=w \mid C=spam)} = \frac{(s*ns)/Ns}{(ns*s)/Ns+(nn*n)/Nn)}$$

Còn đối với các token chỉ xuất hiện trong kho ngữ liệu này mà không xuất hiện trong kho ngữ liệu kia thì không thể kết luận một token chỉ xuất hiện ở kho ngữ liệu spam thì không bao giờ xuất hiện trong kho ngữ liệu non-spam và ngược lại.Trong bài này bọn em sử dụng smoothing.

Như vậy ta có công thức tính xác suất spam của token dựa trên số lần xuất hiện và số email chứa nó là :

$$\mathbf{P} = \frac{(s*ns + 1)/Ns}{(ns*s + 1)/Ns+(nn*n + 1)/Nn)}$$

ns : số email chứa token trong kho spam

nn : số email chứa token trong kho non-spam

s : số lần token xuất hiện trong kho spam

n : số lần token xuất hiện trong kho non-spam

Ns : tổng số email trong kho spam

Nn : tổng số email trong kho non-spam

Từ xác suất này ta so sánh với một giá trị ngưỡng t là ngưỡng để phân loại email là spam hay không, nếu xác suất này lớn hơn t ,ta cho email đó là spam , ngược lại email đó là non-spam.

Trong phân loại email có hai loại sai lầm, một là sai lầm nhận một email là spam thành non-spam và sai lầm thứ hai là nhận một email non-spam thành spam . Rõ ràng sai lầm thứ hai là nghiêm trọng hơn vì người dùng có thể chấp nhận một email spam vượt qua bộ lọc nhưng không thể chấp nhận một email hợp lệ quan trọng lại bị bộ lọc chặn lại .

Giả sử ta gọi S->N và N-> S tương ứng với hai loại lỗi ở trên. Để hạn chế loại lỗi thứ hai ta giả sử rằng lỗi N->S có chi phí gấp a lỗi S->N nghĩa là ta phân loại một email là spam dựa theo :

$$\frac{(P(C=spam\,|X=x)}{P(C=non-spam\,|\,X=x)} > \lambda$$

Mặt khác :

P(C=spam | X=x) =1-P(C=non-spam | X=x) và P(C=spam | X=x) > t

Như vậy ta có giá trị ngưỡng t phụ thuộc vào a :

$$t = \frac{\lambda}{\lambda+1}$$

3 Các chức năng chính của hệ thống

3.1 Hàm get_wordnet_pos

Hàm này sử dụng thư viện treebank của NLTK để kiểm tra và trả về từ loại của một từ.

3.2 Hàm preprocess

Đối với mỗi email đầu vào hàm này sẽ có chức năng tiền xử dữ liệu như tách từ, loại bỏ khoảng trắng, loại bỏ các kí tự đặc biệt, chuyển về chữ thường, loại bỏ stopwords, đưa về từ nguyên gốc.

3.3 Hàm draw

Hàm này có chức năng hiển thị phân bố lớp dữ liệu.

3.4 Hàm write_file

Hàm này sinh ra file train, val, test.

3.5 Hàm get_dict

Hàm này dùng TF-IDF để đánh trọng số từ.

3.6 Hàm set_dict

Hàm này có chức năng tạo từ điển.

3.7 Hàm BinomialClassifier

Hàm này có chức năng tính trước các xác suất.

3.8 Hàm performClassification

Hàm này tính xác suất tập test.

3.9 Hàm messageBox

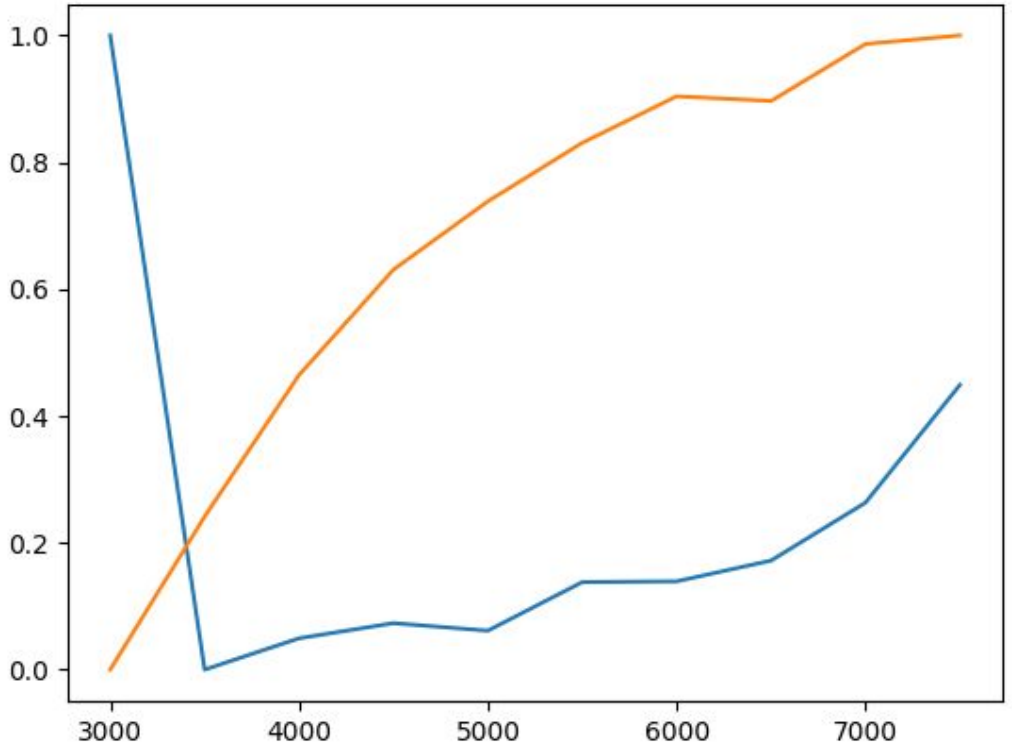
Hàm này có chức năng hiển thị kết quả.

Chương III: Kết quả và đánh giá

1 Thí nghiệm xử lý giảm số chiều

Đặc trưng của các bài toán phân loại văn bản là các văn bản đầu vào là vectơ chứa nhiều giá trị thuộc tính. Việc giảm số chiều cho các vector này đảm bảo các yếu tố chi phí về thời gian, tài nguyên cũng như độ chính xác. Việc giảm số chiều cho vectơ giúp tăng tính tổng quát hóa cho mô hình học được.

Sau việc tham khảo rất nhiều bài báo viết về chủ đề này thì nhóm em thấy số chiều vector được đề nghị là 3000 từ. Xong khi chạy vào bài toán thì không thu được kết quả như mong muốn. Chúng em đã làm thí nghiệm và đưa ra được biểu đồ sau:

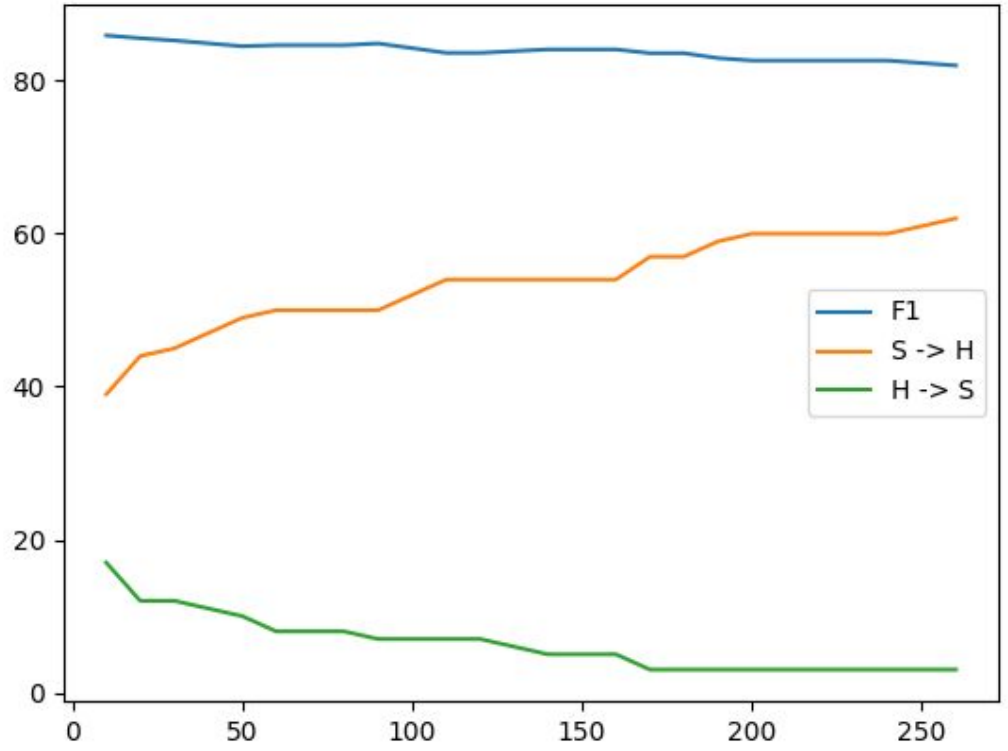


Hình 4 Kết quả thí nghiệm túi từ

Với ngưỡng chấp nhận được là 5% so với giá trị max và chi phí thời gian chạy thì nhóm chúng em chọn giá trị 6000 từ cho bài toán.

2 Thí nghiệm về ngưỡng phạt

Trong bài toán thực tế thì việc một email ham bị phân loại nhầm sang thư spam gây khó chịu lớn hơn cho người sử dụng hơn là phân loại nhầm từ thư spam sang ham. Để khắc phục vấn đề này nhóm em đã sử dụng giá trị ngưỡng T là tỷ lệ giữa xác suất email đó là thư spam so với xác suất thư đó là ham. Chúng em đã làm thí nghiệm vào có được biểu đồ sau:



Hình 5 : Kết quả thí nghiệm ngưỡng phạt

Với ngưỡng chấp nhận được 5% cho giá trị F1 và số email bị nhầm từ ham sang spam là ít nhất chúng em chọn giá trị cho ngưỡng T là 17.

3 Kết quả phân loại trên tập test

Với các tham số lấy được thông qua thí nghiệm trên tập tối ưu chúng em chạy bộ phân loại trên tập test và thu được kết quả như sau:

```
Confusion Matrix:
 [[225  65]
 [ 17 839]]

Accuracy = 92.845, Precision = 92.975, Recall = 77.586, F1 = 84.586


***Repl Closed***
```

Hình 6: Kết quả chạy khi chưa thêm ngưỡng

Với độ chính xác 92.845% có thể nói là mô hình hoạt động rất tốt. Nhưng số lượng mail bị phân loại nhầm từ ham sang spam là tương đối lớn chiếm 1.5% số thư được phân loại.

```
Confusion Matrix:
 [[196  94]
 [  4 852]]

Accuracy = 91.449, Precision = 98.000, Recall = 67.586, F1 = 80.000


***Repl Closed***
```

Hình 7: Kết quả chạy khi thêm ngưỡng

Với việc thêm ngưỡng T cho bài toán chúng ta có thể thấy mặc dù độ chính xác của thuật toán giảm nhưng tương đối ít chỉ có 1.5% và độ chính xác này vẫn đủ để có thể nó hệ thống hoạt động tốt. Trong khi đó số lượng mail ham bị nhận nhầm sang mail spam giảm hơn 4 lần. Cùng với đó là bài toán mới làm trên môi trường nghiên cứu nên các yêu cầu của người dùng đều là giả định. Nhóm em hi vọng trong bài toán thực tế sẽ có các yêu cầu cụ thể hơn để có thể xây dựng bộ phân loại tốt hơn cho trải nghiệm người dùng, cũng như đảm bảo được độ chính xác để phân loại một cách hiệu quả.

Chương IV: Định hướng

Khó khăn chung trong các bài toán phân loại email là việc biểu diễn một email ra một vector. Việc định nghĩa biểu diễn các chiều như thế nào để đạt kết quả tốt nhất cho bài toán. Hướng giải quyết tạm thời của chúng em là loại bỏ các từ stopwords và đánh trọng số cho các từ theo TF-IDF. Ngoài ra còn có các từ sai chính tả của người dùng, vị trí của các từ hay ý nghĩa của các cụm từ như : "only 2$",...Cũng vẫn có thể đóng góp nhiều hơn cho bài toán. Trong bài toán cụ thể này chúng em xử lý các vấn đề trên bằng cách xóa và bỏ qua các từ, cụm từ như vậy. Trong tương lai chúng em hy vọng có thể tìm được các giải thuật tốt hơn trong xử lý ngôn ngữ tự nhiên để việc phân loại tốt hơn.

Khó khăn trong xử lý các từ cùng gốc, với thư viện nhóm em đăng sử dụng là lemmatization thì cho đảm bảo về thời gian chạy nhanh nhưng lại không thể xử lý khi có các từ mới không nằm trong thư viện. Một phương pháp xử lý khác chúng em có tham khảo thư viện spaCy có tốc độ xử lý chậm hơn xong kết cải thiện không đáng kể. Trong tương lai nhóm em có thể các phương pháp xử lý nâng cao hơn như CNN.

Trong quá trình làm việc nhóm em đã tranh luận rất nhiều về việc lựa chọn công thức để tính xác suất cho các từ trong túi từ. Qua nhiều buổi tranh luận mà không có kết quả nhóm em đã tìm hiểu được một công thức mà trong mô hình nhóm em sử dụng. Từ nhược điểm mà nhóm em thấy được trong hai công thức được trình bày trong phương pháp phần phương pháp thực hiện thì nhóm em quyết định sử dụng công thức ba với việc tận dụng cả hai tính chất của một từ là tần số từng từ và tần số tài liệu. Dù công thức chưa thật sự phù hợp. Trong tương lai chúng em hi vọng được công thức hiệu quả hơn và việc làm mịn cho công thức hiệu quả hơn mà không làm cho phân lớp của nó quả bị lệch.

Trong thực tế có nhiều phương diện cần tối ưu hơn là bài toán phân loại. Cụ thể là trải nghiệm của người dùng là một yếu tố cần xem xét. Phương án xử lý của nhóm em là xác định một ngưỡng T để đảm bảo giảm thiểu số thư hợp lệ bị phân nhầm thành thư spam. Điều này thì làm cho độ chính xác của thuật toán giảm đi. Trong tương lai nhóm em sẽ tìm hiểu thêm các phương pháp khác để phân loại tốt hơn.

Tài Liệu Tham Khảo

1. http://viet.jnlp.org/kien-thuc-co-ban-ve-xu-ly-ngon-ngu-tu-nhien/machine-learning-trong-nlp/phan-loai-van-ban-bang-dinh-ly-bayes
2. https://nguyenvanhieu.vn/tf-idf-la-gi/
3. https://www.kaggle.com/balakishan77/spam-or-ham-email-classification
4. https://www.it-swarm.dev/vi/python/tu-vung-wordnet-va-gan-pos-trong-python/1072796877/
5. https://machinelearningmastery.com/prepare-text-data-machine-learning-scikit-learn/
6. https://github.com/RobinManhas/Spam-filter