PHÂN LOẠI THƯ RÁC VỚI GIẢI THUẬT BOOSTING CÂY QUYẾT ĐỊNH NGẪU NHIÊN XIÊN PHÂN ĐƠN GIẢN

Huỳnh Phụng Toàn[1], Nguyễn Vũ Lâm[2], Nguyễn Minh Trung[1] và Đỗ Thanh Nghị[3]

ABSTRACT

Our investigation aims at classifying spam emails based on machine learning algorithms. The representation of the email that we use for classification is the bag-of-words model, which is constructed from the counting the word occurrence in a histogram like fashion. The pre-processing step brings out a dataset with a very large number of dimensions. Thus, we propose a new algorithm boosting of random oblique decision stumps that is usually suited for classifying very-high-dimensional datasets. The numerical test results on a real dataset collected from 1143 spam and 778 non-spam emails showed that our algorithm boosting of random oblique decision stumps outperforms support vector machine (SVM) and Naïve Bayes in terms of Accuracy, F1-Measure, Precision, TP Rate and TN Rate.

Keywords: Spam emails classification, boosting of random oblique decision stump, classification, data mining.

Title: Spam emails classification with boosting of random oblique decision stump

TÓM TẮT

Trong bài viết này chúng tôi đưa ra hướng tiếp cận học tự động để phát hiện thư rác với giải thuật Boosting cây quyết định ngẫu nhiên xiên phân đơn giản (Boosting of Random Oblique Decision Stump). Để thực hiện, đầu tiên phải tạo ra tập dữ liệu gồm một bộ sưu tập các thư rác và thư không phải là thư rác. Kế tiếp thực hiện tiền xử lý dữ liệu, bao gồm các bước phân tích từ vựng, chọn tập hợp từ hữu dụng để phân loại thư rác, xây dựng mô hình túi từ. Bước tiền xử lý sinh ra tập dữ liệu có số chiều rất lớn, chúng tôi đề nghị giải thuật mới có tên là Boosting cây quyết định ngẫu nhiên xiên phân đơn giản cho phép phân lớp hiệu quả tập dữ liệu này. Kết quả thực nghiệm trên tập dữ liệu thực thu thập từ 1143 thư rác và 778 thư không phải thư rác cho thấy giải thuật do chúng tôi đề nghị phân lớp chính xác hơn so với giải thuật SVM và Naïve Bayes qua các tiêu chí so sánh như Accuracy, F1-Measure, Precision, TP Rate và TN Rate.

Từ khóa: Phân loại thư rác, giải thuật học Boosting cây quyết định ngẫu nhiên xiên phân đơn giản, giải thuật phân lớp dữ liệu, khai mỏ dữ liệu.

1 GIỚI THIỆU

Trong những năm 1990, cuộc cách mạng kỹ thuật số cho phép số hóa thông tin dễ dàng và chi phí thấp, thêm vào đó là sự phát triển của công nghệ thông tin cả phần cứng lẫn phần mềm, công nghệ truyền thông, web, internet đã góp phần đưa máy tính vào các sinh hoạt thường nhật. Dịch vụ thư điện tử hiện trở thành phương tiện liên lạc được nhiều người sử dụng nhất nhờ vào sự tiện lợi như chi phí thấp, nhanh, hiệu quả. Tuy nhiên, nghiên cứu cũng cho thấy rằng, người dùng máy tính

[1] Bộ môn Tin Học Ứng Dụng, khoa Khoa Học Tự Nhiên, Trường Đại học Cần Thơ
[2] Trung tâm Tin Học-Công Nghệ Phần Mềm, Trường Cao Đẳng Cộng Đồng Kiên Giang
[3] Bộ môn Khoa Học Máy Tính, khoa CNTT&TT, Trường Đại học Cần Thơ

trên toàn cầu nhận nhiều thư điện tử không phục vụ cho mục đích thiết thực của họ, mà thường chỉ là những thư quảng cáo, thư phản động, chơi đánh bạc, thậm chí là những đoạn mã độc hại, đồi trụy khác, mà chúng ta gọi đó là thư rác (spam emails).

Trong những năm gần đây, các phương pháp chống thư rác đều theo hai nhóm tiếp cận chính (Goodman *et al*., 2007), (Guzella and Caminhas, 2009), (Sebastiani, 2002). Nhóm giải pháp đầu tiên dựa vào việc tìm kiếm chính xác các mẫu xác định danh sách chính xác các địa chỉ người gửi thư rác, các địa chỉ mạng, tìm kiếm chính xác từ khóa, để ngăn chặn thư rác. Nhóm giải pháp này thường rất nhanh nhưng rất dễ bị người phát tán thư rác qua mặt do chỉ cần thay đổi một ít thông tin là các chương trình không thể phát hiện được. Trong thực tế, các nghiên cứu chủ yếu tập trung vào máy học để phân loại thư rác dựa vào nội dung. Nghiên cứu của Drucker và các cộng sự của ông (Drucker *et al*., 1999) sử dụng mô hình túi từ và giải thuật máy học véctơ hỗ trợ (SVM (Vapnik, 1995)) để phân lớp hiệu quả thư rác. Một nghiên cứu khác của Sahami và các cộng sự (Sahami *et al*., 1998) đề xuất kết hợp mô hình túi từ và giải thuật học Bayes thơ ngây (Naïve Bayes (Good, 1965)) cho phân lớp thư rác. So với sử dụng SVM thì Bayes thơ ngây chạy nhanh hơn nên được dùng nhiều trong thực tế mặc dù kết quả thấp hơn. Trong bài viết này, chúng tôi đề xuất giải thuật phân lớp thư rác bằng phương pháp kết hợp mô hình túi từ và giải thuật mới do chúng tôi đề xuất tên là Boosting cây quyết định ngẫu nhiên xiên phân đơn giản (Boosting of Random Oblique Decision Stump - BRODS). BRODS là một giải thuật đơn giản, phân lớp hiệu quả trên tập dữ liệu có số chiều rất lớn và thời gian thực hiện tương đối nhanh. Kết quả thực nghiệm trên tập dữ liệu thư điện tử cho thấy rằng BRODS phân loại thư rác với độ chính xác cao hơn so với SVM (Vapnik, 1995) và Bayes thơ ngây (Good, 1965) trên các tiêu chí như Accuracy, F1-Measure, Precision, TP Rate và TN Rate.

Trong phần 2, chúng tôi sẽ trình bày tóm tắt các bước tiền xử lý dữ liệu dùng để phân loại thư rác dựa trên thư viện Bow (McCallum, 1998). Tiếp theo, chúng tôi sẽ mô tả giải thuật BRODS dùng cho việc phân loại thư rác trong phần 3. Kết quả thực nghiệm sẽ được trình bày trong phần 4 trước phần kết luận và hướng phát triển trong phần 5.

2 TIỀN XỬ LÝ DỮ LIỆU

Tiếp cận của phương pháp mô hình túi từ và máy học bắt đầu từ việc tạo ra tập dữ liệu để học mô hình phân lớp thư điện tử có phải là thư rác hay không. Để giải quyết vấn đề này, chúng tôi phải tạo ra tập dữ liệu học gồm các thư rác và không phải thư rác, phân tích từ vựng và tách các từ trong nội dung thư của tập dữ liệu này. Sau đó chọn tập hợp các từ mà có thể dùng để phân loại thư rác, biểu diễn dữ liệu thư điện tử về dạng bảng để từ đó các giải thuật máy học có thể học để phân loại thư.

Tập dữ liệu do chúng tôi sưu tầm gồm có 1921 tập tin sử dụng ngôn ngữ tiếng Anh và tiếng Việt (loại bỏ dấu) được phân vào hai thư mục tương ứng với hai lớp thư rác hay không phải thư rác. Tập dữ liệu có 1143 thư rác và 778 thư không phải thư rác. Chúng tôi chỉ sử dụng tiêu đề và nội dung của thư để xử lý. Tiếp theo, mỗi thư

được biểu diễn bằng một véctơ tần số của các từ trong lá thư đó. Véctơ này được xem như một phần tử trong tập dữ liệu, để làm được điều này, chúng tôi sử dụng thư viện Bow (McCallum, 1998) để tách từ và chuyển dữ liệu về với dạng bảng, gồm hai bước sau:

Bước 1: Xây dựng mô hình tách từ của các thư điện tử trong hai thư mục thư rác và không phải thư rác. Ở bước này chúng ta thu được mô hình gồm có 28719 từ đã bỏ qua các từ có ít ý nghĩa trong các thư, chẳng hạn như mạo từ, giới từ. Mô hình tách từ của Bow vừa xây dựng có dạng bảng với các cặp chỉ mục và tần số xuất hiện từ (như Hình 1).

- Dòng : 1921 dòng tương ứng với 1921 thư trong tập dữ liệu.

- Cột : 57440 cột, ý nghĩa của từng cột như sau

  + Cột 1, 2 tương ứng là : tên tập tin và tên lớp (tên thư mục)

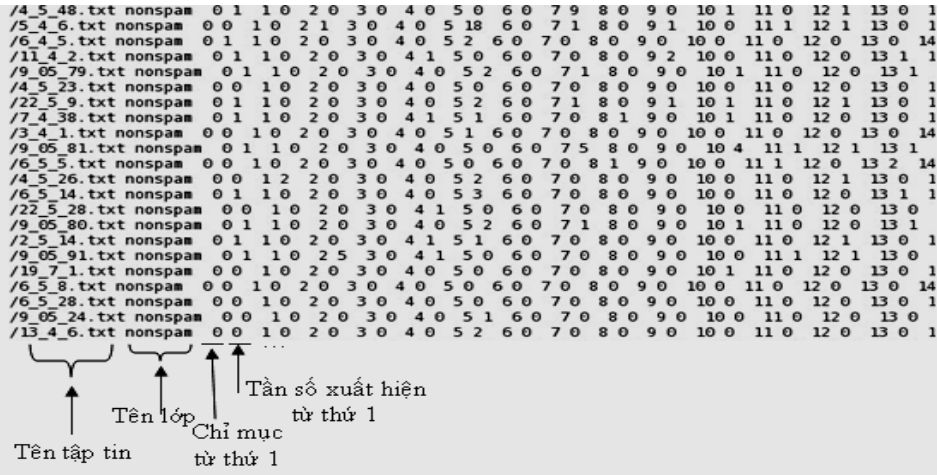
  + Cột 3 là chỉ mục từ thứ nhất (bắt đầu là số 0)

  + Cột 4 là tần số xuất hiện của chỉ mục từ thứ nhất

  …

  + Cột 57439 là chỉ mục từ thứ 28718

  + Cột 57440 là tần số xuất hiện của chỉ mục từ thứ 28718



Hình 1: Mô hình tách từ của thư điện tử thu được từ thư viện Bow

Bước 2: Dựa trên mô hình tách từ của Bow vừa xây dựng, chúng tôi biểu diễn thư điện tử về mô hình túi từ bằng cách trích ra cột thứ 2 làm lớp dữ liệu và các cột tần số xuất hiện của các từ đưa về một bảng dữ liệu. Với mô hình túi từ, chúng tôi thu được bảng dữ liệu có 1921 dòng (mỗi dòng tương ứng với một thư) và 28719 thuộc tính (mỗi thuộc tính tương ứng với một từ, giá trị mỗi thuộc tính là tần số xuất hiện của từ trong thư) và thuộc tính cuối cùng là lớp dữ liệu (thư rác hay không phải thư rác).

Qua bước tiền xử lý dữ liệu, chúng ta có thể thấy rằng tập dữ liệu thư có số lượng chiều (thuộc tính) rất lớn. Ngoại trừ giải thuật SVM (Vapnik, 1995) và rừng ngẫu nhiên xiên phân RF-ODT (Do *et al*., 2009), hầu hết các phương pháp học tự động hiện nay gặp nhiều khó khăn trong việc xử lý tập dữ liệu có số chiều lớn. Để có thể phân lớp hiệu quả tập dữ liệu thư này, chúng tôi đã đề xuất giải thuật Boosting cây quyết định ngẫu nhiên xiên phân đơn giản (BRODS) được kết hợp từ cây ngẫu nhiên xiên phân (Do *et al.*, 2009) với kỹ thuật Boosting (Freund and Schapire, 1995).

3 BOOSTING CÂY NGẪU NHIÊN XIÊN PHÂN ĐƠN GIẢN (BRODS)

Giải thuật boosting cây quyết định ngẫu nhiên xiên phân đơn giản (BRODS) là xây dựng tập các bộ phân lớp yếu là cây ngẫu nhiên xiên phân đơn giản sao cho mỗi bước xây dựng cây tập trung vào khắc phục lỗi từ các mô hình xây dựng từ các bước lặp trước đó.

3.1 Cây quyết định ngẫu nhiên xiên phân đơn giản

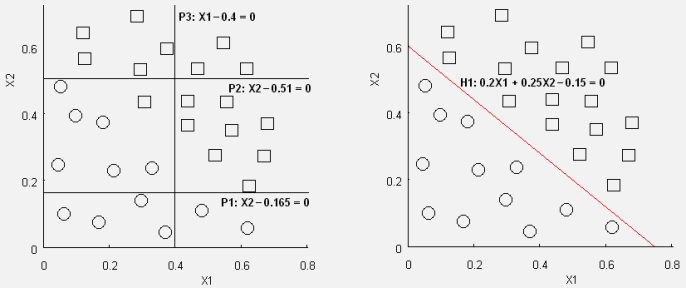
Quá trình xây dựng cây quyết định của các giải thuật học tự động CART (Breiman *et al.,* 1984) hay C4.5 (Quinlan, 1993) được làm như sau:

- Bắt đầu nút gốc, tất cả các dữ liệu học ở nút gốc,

- Nếu dữ liệu tại 1 nút có cùng lớp thì nút được cho là nút lá, nhãn của nút lá là nhãn của các phần tử trong nút lá (hay luật bình chọn số đông nếu nút lá có chứa các phần tử có lớp khác nhau),

- Nếu dữ liệu ở nút không thuần nhất (có sự lẫn lộn các phần tử của nhiều lớp khác nhau) thì nút được cho là nút trong, tiến hành phân hoạch dữ liệu một cách đệ quy bằng việc chọn 1 thuộc tính để thực hiện phân hoạch tốt nhất có thể.

Chúng ta có thể thấy rằng quá trình xây dựng cây của các giải thuật học chỉ chọn một thuộc tính cho việc phân hoạch dữ liệu tại các nút. Hơn nữa, khi Freund và các cộng sự (Freund and Schapire, 1995) đề xuất boosting trên cây quyết định đơn giản (decision stump chỉ gồm 1 nút gốc và 2 nút lá) có thể thấy rằng cây chỉ sử dụng duy nhất 1 thuộc tính để tạo bộ phân lớp yếu. Do đó, tính mạnh mẽ của cây bị giảm khi làm việc với các tập dữ liệu có số chiều lớn và phụ thuộc lẫn nhau, chẳng hạn như dữ liệu thư điện tử mà chúng tôi xử lý ở đây. Một ví dụ trong hình 2, bất kỳ việc phân hoạch đơn thuộc tính nào (song song với trục tọa độ) đều không thể tách dữ liệu một lần duy nhất thành hai lớp một cách hoàn toàn mà phải thực hiện nhiều lần phân hoạch, nhưng việc phân hoạch đa chiều (xiên phân, kết hợp 2 thuộc tính) có thể thực hiện một cách hoàn hảo với duy nhất một lần. Tức là, cây quyết định đơn giản (decision stump) không hiệu quả bằng cây quyết định xiên phân đơn giản (decision oblique stump).



Hình 2: Phân hoạch đơn thuộc tính (trái), phân hoạch đa thuộc tính (phải)

Để khắc phục nhược điểm trên, nhiều giải thuật xây dựng cây quyết định sử dụng phân hoạch đa thuộc tính (xiên phân) tại các nút được đề nghị. Vấn đề xây dựng cây quyết định xiên tối ưu đã được biết như là một vấn đề có độ phức tạp NP-hard. Nghiên cứu tiên phong của Murthy và các cộng sự trong (Murthy *et al.*, 1993) đã đưa ra giải thuật OC1, một hệ thống dùng để xây dựng các cây quyết định xiên trong đó dùng leo đồi để tìm một phân hoạch xiên tốt dưới dạng một siêu phẳng. Rừng ngẫu nhiên xiên phân RF-ODT của chúng tôi trong (Do, *et al.*, 2009) xây dựng các cây xiên phân ngẫu nhiên dựa trên siêu phẳng tối ưu (phân hoạch hiệu quả cao, khả năng chịu đựng nhiễu tốt) thu được từ huấn luyện SVM (Vapnik, 1995).

Để giải quyết 2 vấn đề chính là độ phức tạp và hiệu quả của bộ phân lớp yếu của kỹ thuật boosting, chúng tôi đề xuất chỉ xây dựng cây ngẫu nhiên xiên phân đơn giản (Random Oblique Decision Stump, RODS). Giải thuật RODS xây dựng cây như mô tả trong hình 3. Cây xiên phân 3 nút, bắt đầu với toàn bộ dữ liệu nằm ở nút gốc, chọn ngẫu nhiên ***n'*** thuộc tính từ tập ***n*** thuộc tính ban đầu của dữ liệu là tìm ra siêu phẳng tối ưu ***n'*** chiều (SVM (Vapnik, 1995)) để phân hoạch dữ liệu.

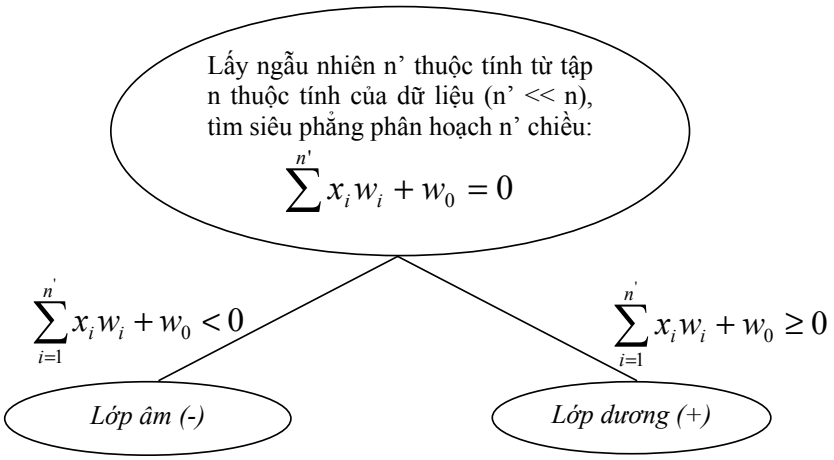
Siêu phẳng cần xác định có dạng:

$$\sum_{i=1}^{n} x_i w_i + w_0 = 0$$

Trong đó x_i là thuộc tính thứ i (chiều) của dữ liệu, w_i là trọng số véctơ pháp tuyến của siêu phẳng, w_0 là độ lệch của siêu phẳng.

Dựa vào dấu của biểu thức $\sum_{i=1}^{n} x_i w_i + w_0$ mà dữ liệu sẽ được phân hoạch qua trái hay qua phải để dự báo nhãn.

Cây xiên phân ngẫu nhiên đơn giản có thể làm việc hiệu quả trên tập dữ liệu có số chiều lớn do nó đảm bảo được 2 yếu tố cơ bản là thời gian xây dựng nhanh và hiệu quả phân lớp cao. Do đơn giản chỉ có 3 nút, việc xây dựng cây xiên phân ngẫu nhiên rất nhanh khi chỉ tìm một siêu phẳng tối ưu trong không gian ***n'*** chiều (***n'*** << ***n***). Việc kết hợp nhiều thuộc tính để tạo phân hoạch xiên phân giúp phân lớp hiệu quả dữ liệu có số chiều lớn.



Hình 3: Cây ngẫu nhiên xiên phân đơn giản

3.2 Giải thuật Boosting cây ngẫu nhiên xiên phân đơn giản (BRODS)

Tuy nhiên, để cải thiện hơn hiệu quả phân lớp dữ liệu thư rác, chúng tôi tiếp tục áp dụng kỹ thuật boosting dựa trên bộ phân lớp yếu là cây ngẫu nhiên xiên phân đơn giản.

Boosting được Freund và các đồng nghiệp của ông phát triển trong thập niên 1990. Đây là một phương pháp xây dựng tập hợp các mô hình phân lớp yếu để nâng cao hiệu quả của các bộ phân lớp này.

Ý tưởng chính của giải thuật này là lặp lại quá trình học của một bộ phân lớp yếu nhiều lần. Sau mỗi bước lặp, bộ phân lớp yếu (chẳng hạn như cây ngẫu nhiên xiên phân đơn giản) sẽ tập trung học trên các phần tử bị phân lớp sai trong các lần lặp trước. Để làm được điều này, người ta gán cho mỗi phần tử một trọng số. Khởi tạo, trọng số của các phần tử bằng nhau. Sau mỗi bước học, các trọng số này sẽ được cập nhật lại bằng cách tăng trọng số cho các phần tử bị phân lớp sai và giảm cho các phần tử được phân lớp đúng. Kết thúc quá trình học thu được tập hợp các mô hình học dùng để phân lớp. Để phân lớp dữ liệu mới đến, người ta sử dụng luật bình chọn số đông từ kết quả phân lớp của từng mô hình phân lớp yếu. Độc giả quan tâm đến boosting có thể tham khảo chi tiết kỹ thuật ở tài liệu (Freund and Schapire, 1995).

Đầu vào:
- m phần tử dữ liệu : $\{(x_j, y_j)\}_{j=1,m}$ với $x_j \in R^n$ và $y_j \in \{1, -1\}$
- số bước lặp T

Huấn luyện:
► khởi động phân phối của m phần tử dữ liệu $Dist_1(j)$
cho j = 1 tới m thực hiện
$Dist_1(j) = 1/m$

► cho i = 1 tới T thực hiện (lặp T bước)
- lấy mẫu S_i phần tử dựa trên phân phối $Dist_i$
- học mô hình cây ngẫu nhiên xiên phân đơn giản h_i từ tập mẫu S_i
$h_i = RODS(S_i)$
- tính lại lỗi dự đoán của mô hình h_i
$\varepsilon_i = \sum_{j=1}^{m} {}_{h_i(x_j) \neq y_j} Dist_i(j)$
- tính trọng số cho mô hình h_i
$\alpha_i = (1/2)ln[(1-\varepsilon_i)/\varepsilon_i]$
- cập nhật lại phân phối của m phần tử dữ liệu
cho j = 1 tới m thực hiện
nếu $(y_j = h_i(x_j))$ thì $Dist_{i+1}(j) = Dist_i(j)\ exp(-\alpha_i) / fac_i$
ngược lại thì $Dist_{i+1}(j) = Dist_i(j)\ exp(\alpha_i) / fac_i$
với $fac_i = 2\sqrt{\varepsilon_i(1-\varepsilon_i)}$

► trả về T cây ngẫu nhiên xiên phân đơn giản và trọng số tương ứng $\{h_i, \alpha_i\}_{i=1,T}$

Phân lớp:
► phân lớp phần tử x: $H(x) = sign(\sum_{i=1}^{T} \alpha_i h_i(x))$

Giải thuật 1: Boosting của cây ngẫu nhiên xiên phân đơn giản

Giải thuật boosting cây ngẫu nhiên xiên phân đơn giản cải thiện được độ chính xác so với việc sử dụng mô hình đơn của một cây. Giải thuật đáp ứng được yêu cầu xử lý hiệu quả dữ liệu thư điện tử có số chiều lớn.

4 KẾT QUẢ THỰC NGHIỆM

Để đánh giá hiệu quả của tiếp cận do chúng tôi đề xuất cho phân lớp thư rác, chúng tôi đã tiến hành cài đặt giải thuật boosting cây ngẫu nhiên xiên phân đơn giản (BRODS) bằng C/C++ trên hệ điều hành LINUX (Mandriva 2008) có sử dụng thư viện SGDSVM (Bottou and Bousquet, 2008).

Như đã trình bày ở phần 2, sau bước tiền xử lý chúng tôi thu được tập dữ liệu có 1921 phần tử (1143 thư rác và 778 thư không phải thư rác), 28719 thuộc tính, 2 lớp (thư rác hay không phải thư rác). Chúng tôi sử dụng giải thuật boosting cây ngẫu

nhiên xiên phân đơn giản, giải thuật máy học SVM chuẩn (LibSVM (Chang and Lin, 2001)) và Bayes thơ ngây trong thư viện máy học Weka (Witten and Frank, 2005) để phân lớp dữ liệu thư rác hay không. Nghi thức kiểm tra chéo (10-fold) được áp dụng để đánh giá hiệu quả của các giải thuật phân lớp. Cách làm như sau: tập dữ liệu chia thành 10 phần bằng nhau, ở lần thứ ***i*** lấy ra phần thứ ***i*** để làm tập kiểm tra và 9 phần còn lại dùng làm tập huấn luyện. Kết quả được tổng hợp từ 10 lần thực thi như vừa mô tả. Các tiêu chí đánh giá hiệu quả dựa vào các kết quả thu được từ phân lớp của các giải thuật.

tp: số thư rác được phân loại vào lớp thư rác,

fp: số thư không phải thư rác được phân loại vào lớp thư rác,

fn: số thư rác được phân loại vào lớp không phải thư rác,

tn: số thư không phải thư rác được phân vào lớp không phải thư rác.

Một số độ đo được sử dụng phổ biến hiện nay là:

TP Rate = Recall = tp/(tp+fn)

TN Rate = tn/(tn+fp)

Precision = tp/(tp+fp)

F1-Measure = (2*Precision*Recall)/(Precision + Recall)

Accuracy = (tp + tn)/(tp+fp+tn+fn)

Chúng tôi sử dụng cả 5 tiêu chí trên để so sánh hiệu quả của các giải thuật phân lớp thư rác. Bảng 1 trình bày kết quả thu được từ 3 giải thuật phân lớp boosting cây ngẫu nhiên xiên phân đơn giản, SVM và Bayes thơ ngây. Kết quả cao nhất trong bảng được in đậm, hạng nhì được in nghiêng. Nhìn vào bảng kết quả, có thể thấy được Bayes thơ ngây cho kết quả rất thấp khi phân lớp. Mặc dù Bayes thơ ngây được dùng phổ biến trong thực tế cho lọc thư rác nhưng với dữ liệu số chiều lớn thì mô hình này không còn hiệu quả nữa. Tiếp đến là giải thuật SVM, có lẽ do sử dụng mô hình đơn nên SVM cho kết quả thấp hơn giải thuật do chúng tôi đề xuất (boosting cây ngẫu nhiên xiên phân đơn giản BRODS). Kết quả này cho phép chúng tôi tin tưởng vào khả năng xử lý hiệu quả của giải thuật cho vấn đề phân lớp thư rác và dữ liệu có số chiều lớn.

Bảng 1: Kết quả phân lớp thư rác

	Accuracy	**F1-Measure**	**Precision**	**TN Rate**	**TP Rate**
BRODS	**97.44%**	**97.84%**	**97.43%**	**96.34%**	**98.25%**
SVM	*92.50%*	*93.70%*	*93.62%*	*90.61%*	93.78%
Naïve Bayes	56.27%	64.31%	48.03%	28.34%	*97.30%*

5 KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

Trong bài viết này, chúng tôi vừa trình bày một hướng tiếp cận trong việc phân loại thư rác. Chúng tôi đã sử dụng mô hình túi từ để biểu diễn các thư điện tử. Tập dữ liệu thư rác sau bước tiền xử lý cho ra tập dữ liệu có số chiều lớn đến 28719.

Tiếp theo sau, chúng tôi đề nghị giải thuật mới Boosting cây quyết định ngẫu nhiên xiên phân đơn giản cho phép phân lớp hiệu quả trên dữ liệu thư điện tử có số chiều lớn. Kết quả thực nghiệm cho thấy giải thuật do chúng tôi đề xuất cho kết quả tốt hơn giải thuật SVM và Naïve Bayes qua các tiêu chí so sánh như Accuracy, F1-Measure, Precision, TP Rate và TN Rate.

Trong tương lai, chúng tôi sẽ phát triển để xây dựng ứng dụng tích hợp vào hệ thống của server cho phép lọc thư rác. Ngoài ra, cũng có thể mở rộng nghiên cứu sang nhận dạng tấn công qua mạng.

TÀI LIỆU THAM KHẢO

Bottou, L. and Bousquet, O. (2008). The Tradeoffs of Large Scale Learning. *Advances in Neural Information Processing Systems* Vol(20):161–168, Edited by J.C. Platt, D. Koller, Y. Singer and S. Roweis.

Breiman, L., Friedman, J., Olshen, R. and Stone, C. (1984). *Classification and Regression Trees*. Chapman & Hall, New York.

Chang, C. and Lin, C-J. (2001). LIBSVM: A library for support vector machines. Software available at http://www.csie.ntu.edu.tw/~cjlin/libsvm.

Cristianini, N., and Shawe-Taylor, J. (2000). *An Introduction to Support Vector Machinesand Other Kernel-based Learning Methods*. Cambridge University Press.

Do, T-N., Lenca, P., Lallich, S. and Pham, N-K. (2009). Classifying Very-high-dimensional Data with Random Oblique Decision Trees. in *Advances in Knowledge Discovery and Management*, H. Briand, F. Guillet, G. Ritschard, D. Zighed Eds, Springer-Verlag, pp. 39-55.

Drucker, H., Wu, D. and Vapnik, V. (1999). Support vector machines for spam categorization. *IEEE Transactions on Neural networks* 10(5):1048-1054.

Freund, Y., Schapire, R.E. (1995). A decision-theoretic generalization of on-line learning and an application to boosting. *Computational Learning Theory*, pp. 23–37.

Good, I. 1965. The Estimation of Probabilities: An Essay on Modern Bayesian Methods. *MIT Press*.

Goodman, J-G., Cormack, V. and Heckerman, D. (2007). Spam and the ongoing battle for the inbox. *Communications of the ACM* 50(2):25-33.

Guzella, T-S. and Caminhas, W-M. (2009). A review of machine learning approaches to spam filtering. *Expert Systems with Applications* 36:10206-10222.

McCallum, A. (1998). Bow: A Toolkit for Statistical Language Modeling, Text Retrieval, Classification and Clustering. http://www-2.cs.cmu.edu/~mccallum/bow.

Murthy, S., Kasif, S., Salzberg, S. and Beigel, R. (1993). Oc1: Randomized induction of oblique decision trees. Proc. of the 11th National Conference on AI, pp. 322–327.

Quinlan, J. (1993). C4.5: *Programs for Machine Learning*. Morgan Kaufmann Publishers.

Sahami, M., Dumais, S., Heckerman, D. and Horvitz, E. 1998. A bayesian approach to filtering junk e-mail. In Learning for Text Categorization Workshop. AAAI Technical Report, WS-98-05.

Sebastiani, F. (2002). Machine Learning in Automated Text Categorization. *ACM Computing Surveys* 34(1):1-47.

Vapnik, V. (1995). *The Nature of Statistical Learning Theory*. Springer-Verlag, New York.

Witten, I.H. and Frank, E. (2005). *Data Mining: Practical Machine Learning Tools and Techniques*. Morgan Kaufmann.