PHÂN LỚP Ý KIẾN NGƯỜI HỌC DỰA TRÊN MÔ HÌNH CHỦ ĐỀ

1st Phạm Thị Kim Ngoan, 2st Trieu Hai Nguyen
Bộ Môn Kỹ Thuật Phần Mềm, Khoa CNTT,
Trường Đại Học Nha Trang
Số 02 Nguyễn Đình Chiểu - Nha Trang - Khánh Hòa
e-mails: ngoanptk@ntu.edu.vn, trieu.science@gmail.com

***Tóm tắt nội dung*—Đảm bảo chất lượng đào tạo đang nhận được nhiều sự quan tâm của các cơ sở đào tạo Đại học. Người học đóng vai trò quan trọng trong việc đảm bảo chất lượng đào tạo. Với mục tiêu hiểu được các phản hồi của người học về các hoạt động giảng dạy tại trường Đại học Nha Trang nhằm góp phần nâng cao chất lượng đào tạo của Nhà trường, chúng tôi đề xuất xử lý các ý kiến phản hồi của người học thông qua phân lớp và gán nhãn các ý kiến phản hồi dựa trên mô hình chủ đề (Topic Model). Việc phân lớp và gán nhãn chủ đề được thực hiện theo phương pháp Support Vector Machine (SVM) và Naive Bayes Classifier (NBC). Thực nghiệm cho kết quả khả quan trên tập dữ liệu ý kiến đánh giá của người học trường Đại học Nha Trang với phương pháp SVM và NBC tương ứng là 88.27% và 85.11%.**

***Index Terms*—Topic Model, Support vector machine (SVM), Naïve Bayesian Classification (NBC), Ý kiến người học, Phân loại văn bản.**

I. Giới thiệu

Ở các nước phát triển, lấy ý kiến phản hồi của người học đã có từ lâu và là một hoạt động phổ biến. Tại Đại học Harvard, việc thu thập phản hồi của sinh viên diễn ra thường xuyên vào đầu học kỳ, giữa kỳ và cuối học kỳ [6]. University of Malta thiết kế các mẫu đánh giá về bài học, chương trình học để thu nhận các ý kiến từ người học định kỳ cuối bài, cuối chương trình [9]. Các trường Đại học thông qua phản hồi từ người học nhằm thu nhận những thông tin về chất lượng giảng dạy và học tập tại Trường.

Ở Việt Nam, người học đóng vai trò quan trọng trong việc đảm bảo chất lượng đào tạo, vai trò này được xác định trong Luật Giáo dục Đại học số 08/2012/QH13 và trong Bộ tiêu chuẩn kiểm định chất lượng giáo dục trường đại học theo thông tư số 12/2017/TTBGDDT. Hầu hết các trường Đại học đều có các kênh để lấy ý kiến phản hồi từ người học về quá trình đào tạo, các hoạt động giảng dạy của giảng viên. Tuy nhiên, mỗi trường có cách lấy ý kiến và xử lý số liệu thu được khác nhau.

Với mục tiêu tăng cường tinh thần trách nhiệm của sinh viên với quyền lợi, nghĩa vụ học tập, rèn luyện của bản thân, duy trì và nâng cao chất lượng đào tạo, trong nhiều năm qua, công tác "Lấy ý kiến phản hồi từ người học về hoạt động giảng dạy của giảng viên" là nhiệm vụ thường xuyên tại cuối mỗi học kỳ tại Trường Đại học Nha Trang. Trong phiếu đánh giá của Trường, ngoài những tiêu chí định lượng còn có các câu hỏi mở. Thông qua câu hỏi mở, Trường đã nhận được rất nhiều ý kiến khác được người học phản hồi dưới dạng dữ liệu văn bản. Các ý kiến khác thường liên quan đến các đề xuất của người học để nâng cao chất lượng đào tạo của Nhà trường, có nhiều ý hay nhưng chưa được xử lý, vì việc xử lý thủ công gặp nhiều khó khăn và mất rất nhiều thời gian.

Trong báo cáo này, chúng tôi đề xuất xử lý các ý kiến khác nhận được qua các câu hỏi mở trong phiếu đánh giá của người học tại trường Đại học Nha Trang bằng phương pháp phân lớp dựa trên mô hình chủ đề. Các phần của báo cáo gồm: mô tả và tiền xử lý dữ liệu, các mô hình phân lớp, kết quả thử nghiệm trên tập dữ liệu ý kiến người học, và kết luận.

II. Mô tả và tiền xử lý dữ liệu

Phân loại văn bản (text) là một bài toán thuộc lĩnh vực học máy (Machine Learning). Do đó, để thực hiện phân lớp phải trải qua hai bước đó là *học* (Learning) và *dự đoán* (Prediction). Để thực hiện bước đầu tiên, chúng ta cần xây dựng một bộ dữ liệu mẫu (training data) và áp dụng các thuật toán learning trên bộ dữ liệu mẫu đó để thu được mô hình (model). Cuối cùng, model này sẽ được dùng để dự đoán trong thực tế.

A. Mô tả dữ liệu

Định kỳ vào các tuần cuối mỗi học kỳ, người học tại trường Đại học Nha Trang vào hệ thống quản lý đào tạo để thực hiện đánh giá hoạt động giảng dạy của Giảng viên. Qua phiếu đánh giá trực tuyến, Trường muốn thu nhận ý kiến phản hồi của người học về chất lượng đào tạo và các hoạt động liên quan đến giảng dạy các học phần, qua đó giúp Nhà Trường, các đơn vị chức năng, Giảng viên nắm được thực trạng đáp ứng nhu cầu của người học, trên cơ sở đó đưa ra các đề xuất phù hợp để nâng cao chất lượng đào tạo của Nhà trường.

Trong phiếu đánh giá, ngoài những tiêu chí định lượng được đánh giá theo thang đo 5 mức độ, còn có các câu hỏi mở để người học có thể góp ý cho Nhà trường và Giảng viên nhằm nâng cao hơn nữa chất lượng giảng dạy. Các ý kiến này được cung cấp dưới dạng dữ liệu văn bản. Sau đợt đánh giá, dữ liệu được xuất ra tập tin bảng tính excel để gửi cho các bên liên quan.

Tập dữ liệu chúng tôi sử dụng trong báo cáo này được lấy ngẫu nhiên một phần từ tập tin excel ý kiến người học tại trường Đại học Nha Trang trong học kỳ 2 năm học 2018–2019. Tập dữ liệu này mô tả các ý kiến người học đánh giá cho các hoạt động

giảng dạy các học phần khác nhau của các Giảng viên thuộc nhiều Khoa, Viện. Sau khi loại bỏ các ý kiến không có nghĩa như: "không, không có, không có ý kiến, em nghĩ vậy là ổn rồi, ...", chúng tôi thu bộ dữ liệu thử nghiệm gồm *1300* ý kiến cho Giảng viên và *600* ý kiến cho Nhà trường. Cuối cùng, dữ liệu được gán làm hai nhãn: *Giảng Viên* và *Nhà Trường*.

B. Tiền xử lý dữ liệu

Đối với bài toán phân lớp ý kiến của người học, chúng tôi áp dụng thuật toán phổ biến hỗ trợ xử lý ngôn ngữ tự nhiên là *Bag-of-words (BoW)*. BoW có nhiệm vụ phân tích và phân nhóm dựa theo "Bag of Words"(corpus) tạo ra bộ *từ điển*. Dựa vào số lần từng từ xuất hiện trong "bag", chúng tôi thu được các vector đặc trưng của văn bản. Đầu vào của Bag-of-words là đoạn văn bản đã được tách từ (Words segmentation). Tách từ là một bước tiền xử lý ngôn ngữ rất quan trọng, đặc biệt là tách từ trong tiếng Việt. Trong tiếng Việt, dấu cách chỉ dùng để phân cách các âm, không được sử dụng để phân tách từ. Ví dụ như khi tách "giảng viên" thành "giảng" và "viên" sẽ gây ra sự vô nghĩa của từ. Trong bài viết này, để thực hiện tách từ chúng tôi sử dụng công cụ **ViTokenizer** của thư viện **pyvi** có sẵn trên *Python* do tác giả "Viet-Trung Tran" xây dựng [16]. Kết quả tách từ thu được độ chính xác từ 96%–98% (xem ví dụ bảng I).

Câu gốc	Áp dụng ViTokenizer
Đầu tư thêm trang thiết bị giảng dạy	Đầu_tư thêm trang_thiết_bị giảng_dạy
Cần phải đi vào chuyên sâu vấn đề giảng dạy hơn nữa	Cần phải đi vào chuyên_sâu vấn_đề giảng_dạy hơn_nữa
Giảng dạy tận tâm	Giảng_dạy tận_tâm
Gv nên chú trọng vào lý thuyết trong bài hơn	Gv nên chú_trọng vào lý_thuyết trong bài hơn

Bảng I
VÍ DỤ TÁCH TỪ TIẾNG VIỆT BẰNG CÔNG CỤ VITOKENIZER CỦA THƯ VIỆN PYVI

Tuy nhiên BoW có một số nhược điểm [13], [15]: *từ điển* chứa rất lớn số lượng từ (từ điển của tập dữ liệu "Ý kiến người học trường Đại Học Nha Trang" của chúng tôi sử dụng trong bài viết này có kích thước là *1555*), dẫn đến vector đặc trưng thu được sẽ có kích thước rất lớn, có rất nhiều từ trong từ điển không xuất hiện trong văn bản dẫn đến trường hợp *vector thưa (sparse vector)*, tức là vector đặc trưng chứa nhiều phần tử 0 và những từ hiếm thì lại mang thông tin quan trọng, có giá trị phân loại cao hơn một số từ thông dụng xuất hiện. Để khắc phục nhược điểm này, chúng tôi áp dụng phương pháp *Term Frequency-Inverse Document Frequency (TF–IDF)* [2], [10], [13] để đánh giá độ quan trọng của một từ dựa vào trọng số của từ đó trên toàn bộ văn bản. Trên thực tế, hầu hết các ngôn ngữ đều có các từ thường xuyên xuất hiện như "the", "a", "is", ... trong tiếng anh hoặc "là", "thì", "của" ... trong tiếng Việt. Các từ này ít có giá trị phân loại văn bản, nếu chỉ dựa trên tần suất xuất hiện thì sẽ làm ảnh hưởng đến những từ ít xuất hiện hơn nhưng có giá trị phân loại cao. Để đánh lại trọng số của từ, đầu tiên cần tính tần số xuất hiện *tf* của một từ trong trong một văn bản dự trên toàn bộ văn bản trong tập training.

$$tf(t,d) = \frac{f_{t,d}}{\sum_{t' \in d} f_{t',d}} \quad (1)$$

trong đó, $f_{t,d}$ là số lần từ t xuất hiện trong văn bản d trên toàn bộ tổng số từ trong văn bản d. Bảng II thể hiện các từ có tần suất xuất hiện nhiều nhất bộ training của chúng tôi. Rõ ràng các từ đó không có giá trị đặc trưng khi phân loại.

Từ	cần	cho	nhiều	sinh_viên	hơn
Tần suất	308	319	337	372	631

Bảng II
BẢNG TẦN SUẤT XUẤT HIỆN CỦA CÁC TỪ TRONG TOÀN BỘ VĂN BẢN

Để giảm giá trị đặc trưng của các từ thường xuyên ở bảng II, chúng tôi sẽ tính idf như sau

$$idf(t,D) = log\left(1 + \frac{|D|}{1 + |d \in D : t \in d|}\right) + 1 \quad (2)$$

trong đó, $|D|$– tổng số văn bản trong tập training. Mẫu số là số văn bản trong tập traning có chứa từ t. Trong công thức (2) được cộng thêm *1* vì nếu một từ không xuất hiện ở bất cứ văn bản nào trong tập training thì mẫu số sẽ bằng *0*. Bảng III cho thấy rằng các từ thường xuất hiện ở bảng II đã được đánh lại trọng số quan trọng trong toàn văn bản. Các từ có trọng số càng cao thì càng có giá trị trong phân loại text.

Từ	idf values
hơn	1.875770
nhiều	2.469349
cần	2.502575
mã_học_phần	7.499787
mô_hình	7.499787
ứng_xử	7.499787

Bảng III
KẾT QUẢ TÍNH GIÁ TRỊ TRỌNG SỐ CỦA idf

Sau khi tìm được tf, idf, công thức $tf - idf$ được tính như sau:

$$tfidf(t,d,D) = tf(t,d) \times idf(t,D) \quad (3)$$

III. CÁC MODEL CLASSIFICATION

Sau khi đã tiền xử lý bộ dữ liệu thô "ý kiến người học ở Đại Học Nha Trang" ở trên, chúng tôi sẽ áp dụng các thuật toán Machine Learning trên bộ data vừa thu được. Hiện nay có rất nhiều thuật toán phân loại văn bản như Naive Bayes Classifier, Decision Tree (Random Forest), Vector Support Machine (SVM), Boosting and Bagging algorithms, Convolution Neural Network (CNN), Recurrent Neural Network (RNN), Long Short-Term Memory (LSTM, Bi-LSTM), SLDA [13], [15]. Việc lựa chọn mô hình nào tốt sẽ phụ thuộc vào bộ dữ liệu văn bản đầu vào. Ví dụ như phương pháp học dựa trên xác suất NBC có hiệu quả cao và đơn giản cài đặt trong phân loại văn bản. CNN cho độ chính xác hầu như là cao nhất trong bài toán Sentiment Classification. SVM có hiệu quả đối với bộ data lớn, có nhiễu, dữ liệu phân tách từ phi tuyến. Trong khuôn khổ bài viết này, chúng tôi sẽ sử dụng phương pháp NBC và SVM vào việc phân loại ý kiến người học của trường Đại Học Nha Trang cũng như đánh giá độ hiệu quả của từng phương pháp.

A. Naive Bayes Classifier (NBC)

Naive Bayes Classification (NBC) là một thuật toán phân loại thuộc nhóm Supervised Learning (học có giám sát) dựa trên tính toán xác suất áp dụng Định lý Bayes (Bayes' Theorem) [2]. Kỹ thuật Naïve Bayesian ban đầu dựa trên dựa trên định nghĩa về xác suất có điều kiện (conditional probability) và "Maximum likelihood" [2], [5], [7], [8]. Định lý Bayes dùng để tính xác suất ngẫu nhiên của sự kiện y khi biết các "*feature vector*" $\mathbf{x} = x_1, \ldots, x_n$ như sau:

$$P(y \mid x_1, \ldots, x_n) = \frac{P(y)P(x_1, \ldots x_n \mid y)}{P(x_1, \ldots, x_n)}. \tag{4}$$

Giả sử rằng các thành phần của "*feature vector*" $\mathbf{x}$ là độc lập với nhau

$$P(\mathbf{x}|y) = P(x_1 \cap x_2 \cap \ldots \cap x_n|y) = \prod_{i=1}^{n} P(x_i \mid y). \tag{5}$$

Từ giả thiết của định lý Bayes ở (5), (4) được viết lại như sau:

$$P(y \mid x_1, \ldots, x_n) = \frac{P(y)\prod_{i=1}^{n} P(x_i \mid y)}{P(x_1, \ldots, x_n)}. \tag{6}$$

Ở các phương trình trên, ta có mẫu số $P(x_1, \ldots, x_n)$ là các hằng số đầu vào đã cho và không phụ thuộc vào $P(y \mid x_1, \ldots, x_n)$. Do đó, chúng ta có thể áp dụng quy tắc phân loại như sau

$$P(y \mid x_1, \ldots, x_n) \propto P(y) \prod_{i=1}^{n} P(x_i \mid y), \tag{7}$$

trong đó, $\propto$ là phép tỉ lệ thuận. (7) được viết lại như sau

$$\hat{y} = \arg\max_{y} P(y) \prod_{i=1}^{n} P(x_i \mid y) \tag{8}$$

Chúng ta có thể sử dụng ước lượng Maximum A Posteriori (MAP) hoặc Maximum Likelihood để tính các phân phối $P(y)$ và $P(x_i \mid y)$ dựa trên tần số tương đối của lớp y trong training data. Ước lượng Maximum Likelihood đưa ra giả sử rằng *feature vector* $\mathbf{x}$ tuân theo một phân phối bất kì và được mô tả bằng tham số θ. Ý tưởng chính của Maximum Likelihood [2], [15] là việc đi tìm bộ tham số θ để xác suất

$$\theta = \max_{\theta} P(x_1, x_2, \ldots, x_n|\theta)$$

đạt giá trị lớn nhất. Trong đó, $P(x_1, |\theta)$ là một xác suất có điều kiện và $P(x_1, \ldots, x_n|\theta)$ là xác suất để toàn bộ các sự kiện $x_1, \ldots, x_n$ xảy ra đồng thời (likelihood). Với giả thiết từ định lý Bayes rằng các thành phần của *feature vector* $\mathbf{x}$ là độc lập, ta có thể quy về bài toán tối ưu

$$\theta = \max_{\theta} \prod_{i=1}^{n} P(x_i|\theta) \tag{9}$$

Bài toán tối ưu (9) được viết lại dưới dạng tương đương bằng cách lấy *log* của vế phải

$$\theta = \max_{\theta} \sum_{i=1}^{n} \log\left(P(x_i|\theta)\right) \tag{10}$$

Phương trình trên ta có thể áp dụng *log* vào vế phải vì *log* là một hàm đồng biến trên tập các số dương và một biểu thức sẽ là lớn nhất nếu *log* của nó là lớn nhất. Do đó, bài toán Maximum Likelihood được đưa về bài toán Maximum Log–likelihood [15]. Áp dụng quy tắc ở (10) vào (8), ta thu được

$$\hat{y} = \arg\max_{y} = \log(P(y)) + \sum_{i=1}^{n} \log(P(x_i|y)) \tag{11}$$

Trên thực tế, giả thiết Naive Bayes Classifiers đưa ra hầu như không thể xảy ra. Nhưng điều này lại giúp bài toán trở nên đơn giản, hoạt động hiệu quả và cực kì nhanh chóng trong nhiều trường hợp thực tế như bài toán phân loại văn bản, lọc tin nhắn rác hay lọc email spam. Việc tính toán phân phối $P(x_i|y)$ phụ thuộc vào loại dữ liệu. Trong trường hợp này là bài toán phân loại văn bản, chúng tôi sẽ sử dụng phân phối *"Multinomial Naive Bayes"*. Trong mô hình phân phối này, giá trị của thành phần x_i trong mỗi feature vector chính là số lần từ thứ i xuất hiện trong văn bản đó. Phân phối Multinomial Naive Bayes được tham số hóa bởi vector $\theta_y = (\theta_{y1}, \ldots, \theta_{yn})$ cho mỗi class y, trong đó n là số lượng các đặc trưng hay nói cách khác, n là kích thước của từ điển trong Bag-of-words ($n = 1555$ trên bộ dữ liệu training của chúng tôi). θ_{yi} là xác suất $P(x_i|y)$ của đặc trưng thứ i rơi vào các mẫu thuộc class y.

Như đã đề cập ở trên, θ_y được ước lượng bằng cách sử dụng *smoothed version of maximum likelihood* [2] (tương ứng với việc đếm tần suất xuất hiện của từ thứ i trong văn bản) như sau

$$\hat{\theta}_{yi} = \frac{N_{yi} + \alpha}{N_y + \alpha n}. \tag{12}$$

Trong đó

$N_{yi} = \sum_{x \in T} x_i$ là tổng số lần xuất hiện của đặc trưng thứ i rơi vào các văn bản của class y trong tập traning T.

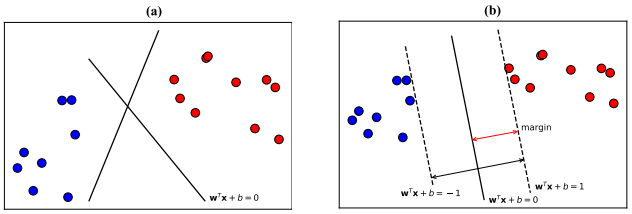
$N_y = \sum_{i=1}^{n} N_{yi}$ là tổng số lần của tất cả các đặc trưng $x_1, \ldots, x_n$ rơi vào class y.

Công thức (12) có thể tránh được hạn chế khi một đặc trưng mới thứ i không xuất hiện lần nào trong class y của tập traning T với mọi $\alpha > 0$. Thông thường, khi chọn $\alpha = 1$ thì được gọi là *Laplace smoothing*, $\alpha < 1$ là *Lidstone smoothing*.

B. Support Vector Machine(SVM)

Bên cạnh việc sử dụng phương pháp phân loại văn bản đơn giản như NBC, trong bài viết này chúng tôi cũng sử dụng phương pháp Support Vector Machine để phân loại ý kiến người học ở trường Đại Học Nha Trang. Các nghiên cứu [4], [12], [14] dựa trên phương pháp SVM cho bài toán phân loại text đều có kết quả rất tốt. SVM cũng là một phương pháp học có giám sát (supervised learning) trong các mô hình nhận dạng mẫu dựa trên việc cực đại hóa dải biên phân lớp (max margin classification) và lựa chọn các kernel phù hợp (xem hình 1). Phương pháp này có thể hoạt động với các dữ liệu được phân tách tuyến tính và phi tuyến.

Kỹ thuật của phương pháp SVM được mô tả tổng quát trong không gian d chiều như sau [3], [14], [15] cho trước $x_1, \ldots, x_N$ điểm và mỗi điểm thuộc vào một class bất kì, cần tìm một siêu phẳng (hyperplane) phân hoạch tối ưu sao cho dấu của hàm



Hình 1. *(a)*–Minh họa mặt phân cách giữa 2 class. *(b)*–Minh họa bài toán tối ưu SVM bằng cách tìm đường phân chia để thu được max margin [3].

ước lượng $H = x \mapsto sign(\mathbf{w}^T\mathbf{x} + b); \mathbf{w} \in R^d, b \in R$ sẽ thể hiện được điểm dữ liệu $x_i \in R^d$ nằm ở cụm dữ liệu nào. Để dễ dàng hiểu được ý tưởng của phương pháp SVM, chúng ta xem xét toán bài toán phân loại hai lớp trong không gian hai chiều như hình minh họa. Rõ ràng trong hình 1(a) chúng ta có thể tìm được nhiều đường phân tách, nhưng nếu chọn được một đường phân tách tối ưu như hình 1(b) thì kết quả sẽ tốt hơn. Nhiệm vụ của phương pháp SVM là đi tìm đường thẳng (siêu phẳng) như hình 1(b). Xem xét tập training có dữ liệu có thể tách rời tuyến tính $(x_1, y_1), (x_2, y_2), \ldots, (x_N, y_N)$. Với mỗi điểm x_i tương ứng với nhãn $y_i \in \pm 1$ (dấu về phía âm hoặc dương), ta thu được đường phân tách giữa 2 class là $H : \mathbf{w}^T\mathbf{x} + b = w_1x_1 + w_2x_2 + b = 0$ và 2 đường thẳng biên gốc H_1, H_{-1} song song với H và có cùng khoảng cách đến H. Với cặp dữ liệu $(\mathbf{x}_n, y_n)$ bất kỳ, khoảng cách từ điểm đó tới mặt phân chia là

$$\frac{y_n(\mathbf{w}^T\mathbf{x}_n + b)}{||\mathbf{w}||_2}$$

Trong hình 1(b), margin được tính là khoảng cách gần nhất từ một điểm bất kì trong class nào tới mặt phân cách

$$\text{margin} = \min_n \frac{y_n(\mathbf{w}^T\mathbf{x}_n + b)}{||\mathbf{w}||_2}$$

Bài toán tối ưu trong SVM trở thành bài toán xác định $\mathbf{w}$ và b sao cho "margin" đạt giá trị lớn nhất [15]

$$\begin{aligned}(\mathbf{w}, b) &= \arg\max_{\mathbf{w},b} \left\{ \min_n \frac{y_n(\mathbf{w}^T\mathbf{x}_n + b)}{||\mathbf{w}||_2} \right\} \\ &= \arg\max_{\mathbf{w},b} \left\{ \frac{1}{||\mathbf{w}||_2} \min_n y_n(\mathbf{w}^T\mathbf{x}_n + b) \right\} \end{aligned} \quad (13)$$

Giả sử rằng không có phần tử nào của tập mẫu nằm giữa H_1 và H_{-1}, tức là $\mathbf{w}.\mathbf{x} + b >= +1$ với $y = +1$ và $\mathbf{w}.\mathbf{x} + b >= -1$ với $y = -1$, ta thu được

$$y_n(\mathbf{w}^T\mathbf{x}_n + b) = 1, \forall n = 1, 2, \ldots, N \quad (14)$$

Bài toán tối ưu (13) đồng nghĩa với việc $||\mathbf{w}||$ đạt nhỏ nhất với ràng buộc ở (14)

$$\begin{aligned}(\mathbf{w}, b) &= \arg\max_{\mathbf{w},b} \frac{1}{||\mathbf{w}||_2} \quad (15) \\ \text{subject to:} \quad & y_n(\mathbf{w}^T\mathbf{x}_n + b) \geq 1, \forall n = 1, 2, \ldots, N\end{aligned}$$

Trong đó, phương trình (15) đã chuyển sang dạng lấy bình phương và chia đôi để dễ dàng tính toán hơn và tối ưu lồi(cả hàm mục tiêu và hàm ràng buộc đều là lồi). Chúng ta có thể giải bài toán lồi này thông qua giải thông qua bài toán đối ngẫu của nó bằng cách cực tiểu hóa hàm Lagrange

$$\mathcal{L}(\mathbf{w}, b, \lambda) = \frac{1}{2}||\mathbf{w}||_2^2 + \sum_{n=1}^{N} \lambda_n(1 - y_n(\mathbf{w}^T\mathbf{x}_n + b)) \quad (16)$$

với $\lambda = [\lambda_1, \lambda_2, \ldots, \lambda_N]^T$ là các hệ số Lagrange, $\lambda_n \geq 0, \forall n$. Tiếp theo, bài toán được chuyển thành bài toán đối ngẫu bằng cách cực đại hóa hàm λ

$$\begin{aligned}\lambda &= \arg\max_{\lambda} \left[\min_{\mathbf{w},b} \mathcal{L}(\mathbf{w}, b, \lambda) \right] \quad (17) \\ \text{subject to:} \quad & \lambda \geq 0, \\ & \sum_{n=1}^{N} \lambda_n y_n = 0.\end{aligned}$$

Giải λ có thể được thực hiện bằng phương pháp quy hoạch động bậc 2 (Quadratic Programing). Từ đó ta có thể tìm được các tham số

$$\mathbf{w} = \sum_{i=1}^{N} \lambda_i y_i \mathbf{x}_i, \qquad b = y_i - \sum_{j=1}^{N} \lambda_j y_j \mathbf{x}_j^T \mathbf{x}_i$$

Trong đó, *Support Vector*: $(\mathbf{x}_i, y_i)$ là một tập điểm dữ liệu bất kì nào đó nằm trên đường biên gốc. Cuối cùng, khi phân loại một mẫu mới sẽ tiến hành kiểm tra hàm dấu

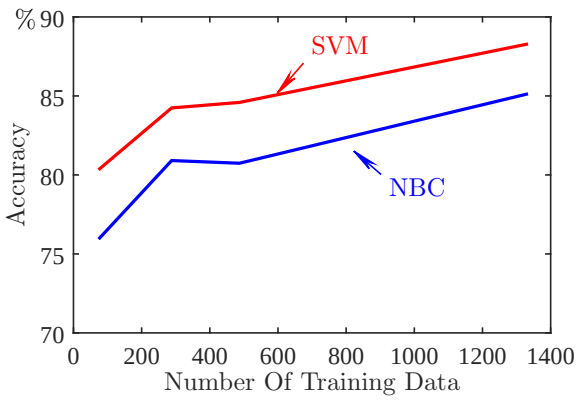
$$sign(\mathbf{w}\mathbf{x} + b).$$

Trong thực tế, dữ liệu được phân tách từ tập traning là phi tuyến, có sự chồng lấn nhau (nhiễu). Dẫn đến các siêu phẳng bây giờ có thể là một mặt cong để phù hợp phân tách dữ liệu. Siêu phẳng này có thể tìm thông qua ánh xạ dữ liệu vào một không gian có số chiều lớn hơn bằng cách sử dụng một hàm nhân K (kernel) thoả mãn điều kiện điều kiện Mercer. Một số kernel phổ biến [3] thường được sử dụng là

Hàm	Công thức
Polynomial Kernels	$K(x, y) = (x^T y + c)^d, c > 0, \forall x, y \in R^n$
Gaussian Kernels	$K(x, y) = exp\left(-\frac{\|\|x-y\|\|^2}{2\sigma^2}\right), \forall x, y \in R^n$
Sigmoid Kernels	$K(x, y) = tanh(ax^T y + b), a, b \geq 0,$ $\forall x, y \in R^n$

IV. Kết quả

Áp dụng các bước đã nêu ở trên cho bài toán phân loại trên tập training ý của kiến người học của trường Đại Học Nha Trang. Cụ thể với số lượng tập training có kích thước là *1329* ý kiến, và tập test là *571* ý kiến, chúng tôi thu được độ chính xác của mô hình NBC là **85.11%** và SVM là **88.27%**. Rõ ràng đối với tập dữ liệu có nhiễu như trong bộ data chúng tôi đã sử dụng thì phương pháp SVM cho kết quả dự đoán tốt hơn phương pháp NBC. Trong hình 2 biểu diễn kết quả độ chính xác của 2 mô hình trên tập training có kích thước lần lượt là *77, 288, 487* và *1329*



Hình 2. Kết quả dự đoán (tính bằng %) của mô hình NBC và SVM.

V. Kết luận

Đảm bảo chất lượng tại các cơ sở đào tạo Đại học là yếu tố then chốt trong sự phát triển hiện nay, trong đó người học đóng vai trò trung tâm, vì vậy các ý kiến phản hồi từ người học rất được các cơ sở đào tạo quan tâm. Tại trường Đại học Nha Trang, các dữ liệu này gần như chưa được xử lý. Với mong muốn phân tích để hiểu được các phản hồi của người học về các hoạt động giảng dạy của giảng viên để góp phần đảm bảo chất lượng đào tạo của Nhà trường. Trong bài báo này, chúng tôi đề xuất xử lý ý kiến người học thông qua mô hình chủ đề. Kết quả thử nghiệm khá khả quan trên tập dữ liệu ý kiến người học tại trường Đại học Nha Trang–một tập dữ liệu có nhiễu với phương pháp SVM là 88.27% và NBC là 85.11%. Bước tiếp theo, chúng tôi sẽ thực hiện phân nhiều lớp ứng với nhiều chủ đề cụ thể hơn để có thể hiểu rõ hơn các phản hồi của người học, từ đó hỗ trợ các đơn vị chức năng trong Trường đưa ra các đề xuất phù hợp để nâng cao chất lượng đào tạo của Nhà trường và cải thiện độ chính xác dự đoán của các mô hình trên.

Tài liệu

[1] Boser, B. E., I. Guyon, and V. Vapnik, A training algorithm for optimal margin classifiers . In Proceedings of the Fifth Annual Workshop on Computational Learning Theory (1992), p. 144–152.

[2] Documentation of scikit–learn, Feature extraction/ Naive Bayes/ Support Vector Machines, https://scikit-learn.org/stable/documentation.html

[3] Do Minh Hai , Support Vector Machine–SVM, https://dominhhai.github.io/vi/2018/03/ml-svm

[4] Durgesh K. Srivastava, Lekha Bhambhu, Data Classification Using Support Vector Machine, Journal of Theoretical and Applied Information Technology.

[5] Harry Zhang, The Optimality of Naive Bayes, Faculty of Computer Science University of New Brunswick.

[6] Harvard University, Ongoing feedback, The Derek Bok Center for Teaching and Learning, https://bokcenter.harvard.edu/ongoing-feedback

[7] Jiawei, Micheline Kamber, Jian Pei, Data Mining Concepts and Techniques, Third Edition.

[8] Karthika, K. and SairamA, N., Naïve Bayesian Classifer for Educational Qualifcation. Indian Journal of Science and Technology, Vol 8(16) (2015).

[9] L-Università ta' Malta (UM), Student Feedback, Academic Programmes Quality & Resources Unit (2019), https://www.um.edu.mt/apqru/studentfeedback

[10] Stephen Robertson, Understanding Inverse Document Frequency: On theoretical arguments for IDF, Journal of Documentation 60 no. 5, pp 503–520.

[11] Hồ Trung Thành, Đỗ Phúc, Mô hình tích hợp khám phá, phân lớp và gán nhãn chủ đề tiếp cận theo mô hình chủ đề.

[12] Thorsten Joachims, Text Categorization with Support Vector Machines: Learning with Many Relevant Features. In European Conference on Machine Learning (ECML), (1998).

[13] Thanh TT, Bag of Words (Bow) TF-IDF–Xử lý ngôn ngữ tự nhiên, https://codetudau.com

[14] Trần Cao Đệ, Phạm Nguyên Khang, Phân Loại Văn Bản Với Máy Học Vector Hỗ Trợ Và Cây Quyết Định, Tạp chí Khoa học 2012:21a 52-63, Trường Đại học Cần Thơ.

[15] Vũ Hữu Tiệp, Machine Learning cơ bản, https://machinelearningcoban.com

[16] Viet-Trung Tran, Python Vietnamese Toolkit, https://pypi.org/project/pyvi/