BIỂU DIỄN NGỮ CẢNH TRONG KHAI TRIỂN CHỮ VIẾT TẮT DÙNG TIẾP CẬN HỌC MÁY

REPRESENTING CONTEXT IN ABBREVIATION EXPANSION USING MACHINE LEARNING APPROACH

Ninh Khánh Duy, Nguyễn Văn Quý

Trường Đại học Bách khoa, Đại học Đà Nẵng; nkduy@dut.udn.vn, quynguyen3490@gmail.com

Tóm tắt - Chuẩn hóa văn bản là bài toán rất cần thiết trong các ứng dụng liên quan đến xử lý ngôn ngữ tự nhiên vì văn bản đầu vào thường chứa nhiều từ không chuẩn như chữ viết tắt, chữ số, và từ ngữ nước ngoài. Bài báo này giải quyết vấn đề chuẩn hóa chữ viết tắt trong văn bản tiếng Việt khi có nhiều lựa chọn để khai triển. Để khử nhập nhằng trong khai triển chữ viết tắt, tiếp cận học máy được sử dụng, trong đó thông tin ngữ cảnh của chữ viết tắt được biểu diễn bởi một trong hai mô hình: Bag-of-words hoặc Doc2vec. Các thử nghiệm với bộ phân lớp Naïve Bayes trên một bộ dữ liệu chữ viết tắt do chúng tôi xây dựng cho thấy tỉ lệ khai triển đúng trung bình của hai mô hình Bag-of-words và Doc2vec lần lượt là 86,0% và 79,7%. Kết quả thực nghiệm cũng cho thấy thông tin ngữ cảnh đóng vai trò quan trọng trong việc lựa chọn khai triển đúng cho một chữ viết tắt.

Từ khóa - chuẩn hóa văn bản; khai triển chữ viết tắt; biểu diễn ngữ cảnh; mô hình Bag-of-words; mô hình Doc2vec; học máy

Abstract - Text normalization is an essential problem in applications involving natural language processing since the input text often contains non-standard words such as abbreviations, numbers, and foreign words. This paper deals with the problem of normalizing abbreviations in Vietnamese text when there are several possible expansions of an abbreviation. To disambiguate the abbreviation expansions, the machine learning approach is used, in which contextual information of abbreviations is represented by either of the two models: Bag-of-words or Doc2vec. Experiments with Naïve Bayes classifier on a dataset of abbreviations collected by us show that the average ratios of expanding correctly for Bag-of-words and Doc2vec are 86.0% and 79.7%, respectively. Experimental results also show that contextual information plays an important role in the correct expansion of an abbreviation.

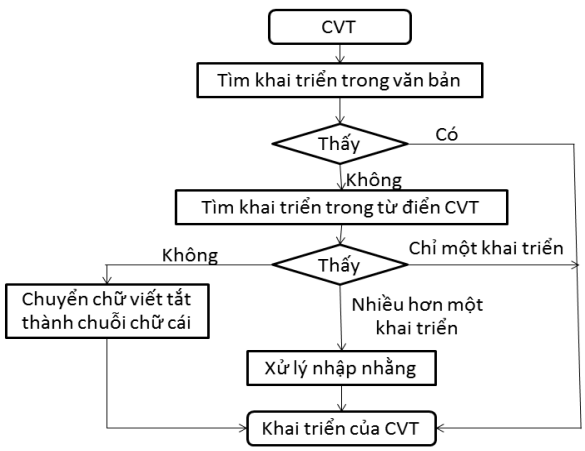
Key words - text normalization; abbreviation expansion; context representation; Bag-of-words model; Doc2vec model; machine learning

1. Đặt vấn đề

Chuẩn hóa văn bản là một vấn đề cần thiết trong các ứng dụng liên quan đến xử lý ngôn ngữ tự nhiên vì văn bản cần xử lý thường chứa những từ không chuẩn (non-standard words) như chữ số, ngày tháng, chữ viết tắt, đơn vị tiền tệ, và từ ngữ nước ngoài [1]. Trong nhiều ứng dụng, chúng ta cần phải chuẩn hóa những từ không chuẩn này bằng cách thay thế chúng bằng những từ phù hợp với ngữ cảnh. Tuy nhiên, việc này không dễ dàng do các từ không chuẩn thường có xu hướng nhập nhằng về ngữ nghĩa hoặc cách phát âm cao hơn so với các từ thông thường. Do đó, cần phát triển các thuật toán thông minh để giải quyết bài toán chuẩn hóa văn bản.

Gần đây đã có một vài nghiên cứu về chuẩn hóa văn bản tiếng Việt, chủ yếu phục vụ cho các hệ thống chuyển văn bản thành tiếng nói [2] [3]. Các nghiên cứu này đã đưa ra các giải pháp chuẩn hóa cho tất cả các lớp từ không chuẩn của tiếng Việt. Tuy nhiên, việc phải xử lý nhiều lớp từ không chuẩn trong phạm vi một nghiên cứu làm cho phương pháp và kết quả chuẩn hóa cho một lớp từ cụ thể không được trình bày rõ ràng và thuyết phục. Điều này đặc biệt đúng với chữ viết tắt (CVT), một lớp từ không chuẩn được dùng khá phổ biến trong các văn bản tiếng Việt. Trong các nghiên cứu [2] [3], các tác giả chỉ trình bày phương pháp khai triển CVT mà không nêu rõ độ chính xác và ưu-nhược điểm của thuật toán khai triển. Thêm vào đó, vấn đề khử nhập nhằng trong khai triển CVT chưa được quan tâm đúng mức.

Từ những vấn đề trên, chúng tôi thấy cần có một nghiên cứu chuyên sâu về chuẩn hóa CVT trong văn bản. Dựa trên thuật toán khai triển CVT được mô tả trong [2], chúng tôi đề xuất thuật toán khai triển CVT như trong Hình 1. Ý tưởng của thuật toán này là ưu tiên tìm kiếm khai triển trong lân cận của CVT trong văn bản, nếu không tìm thấy thì sẽ tìm kiếm trong từ điển CVT. Nếu có nhiều hơn một khai triển trong từ điển thì xử lý nhập nhằng để tìm ra được khai triển tối ưu. Do bài toán tìm kiếm đã được khảo sát nhiều trong các nghiên cứu trước, chúng tôi chỉ tập trung giải quyết vấn đề khử nhập nhằng khi có nhiều khai triển cho một CVT trong bài báo này. Một ví dụ điển hình là chọn lựa một trong hai khai triển, "bài hát yêu thích" hay "bảo hiểm y tế", để chuẩn hóa cho CVT "BHYT".



Hình 1. *Sơ đồ khối thuật toán khai triển CVT*

Cách tiếp cận điển hình đối với bài toán khử nhập nhằng trong khai triển một CVT là sử dụng các quy tắc được thiết kế dựa trên kinh nghiệm rút ra từ một tập dữ liệu thu thập được của CVT đó. Phương pháp này có ưu điểm là đơn giản,

nhưng các quy tắc rút ra từ một tập dữ liệu này khó có khả năng tổng quát hóa cao đối với một tập dữ liệu khác [1]. Do đó, chúng tôi chọn cách tiếp cận dựa trên học máy để giải quyết bài toán gán một CVT vào khai triển đúng của nó. Đây là một dạng của bài toán phân lớp. Bằng việc áp dụng kỹ thuật học máy, mô hình phân lớp được ước lượng dựa trên một tập dữ liệu huấn luyện đủ lớn sẽ có khả năng tổng quát hóa cao đối với tập dữ liệu kiểm chứng bất kỳ.

Để khử nhập nhằng trong khai triển một CVT trong văn bản, thông tin về ngữ cảnh của CVT được sử dụng để ra quyết định phân lớp. Trong nghiên cứu này, chúng tôi chọn ngữ cảnh là toàn bộ câu văn chứa CVT cần khai triển. Vì ngữ cảnh của CVT là thông tin đầu vào của bộ phân lớp, việc biểu diễn ngữ cảnh đóng vai trò quan trọng, ảnh hưởng trực tiếp đến độ chính xác của bộ phân lớp. Chúng tôi đã thử nghiệm hai mô hình biểu diễn ngữ cảnh được sử dụng phổ biến: Bag-of-words [4] và Doc2vec [5] [6], và đưa ra các đánh giá.

Bài báo có bố cục như sau: Phần 2 mô tả việc thu thập dữ liệu CVT; Phần 3 trình bày hai phương pháp biểu diễn ngữ cảnh của CVT; Kết quả thực nghiệm dùng tiếp cận học máy được báo cáo trong Phần 4; Phần 5 đưa ra những bàn luận; Kết luận được trình bày trong Phần 6.

2. Thu thập dữ liệu CVT

2.1. Định nghĩa CVT

Định nghĩa CVT khá không thống nhất, tùy thuộc từng tác giả và nghiên cứu [7]. Trong khuôn khổ một nghiên cứu lớn hơn về chuẩn hóa văn bản cho ứng dụng chuyển văn bản thành tiếng nói [8], bài báo này định nghĩa một từ trong văn bản là CVT nếu nó có độ dài từ hai ký tự trở lên và được cấu thành từ các thành phần sau:

- Ký tự chữ hoa từ "A" đến "Z","Đ", "Ư";
- Ký tự ký hiệu bao gồm: ".","&","-".

Các ví dụ CVT điển hình là: "GS.TS" (Giáo sư Tiến sỹ), "BCHTU" (Ban chấp hành Trung Ương).

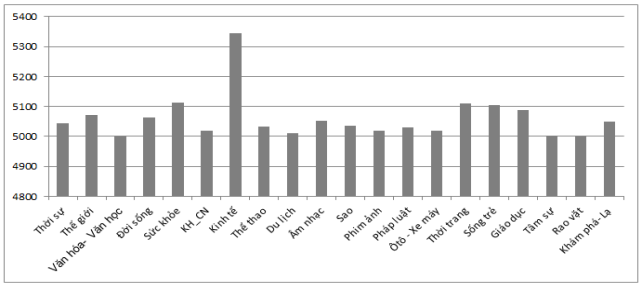
Bài báo này cũng định nghĩa hai trường hợp ngoại lệ sau không được xem là CVT do công cụ chuẩn hóa văn bản của chúng tôi đã phân các từ này vào lớp "Chữ số La Mã" hoặc lớp "Đơn vị tiền tệ" và có cách khai triển riêng:

- Chữ số La Mã (ví dụ: "IV", "XII").
- Đơn vị tiền tệ (ví dụ: "USD", "EUR").

2.2. Thống kê dữ liệu

Để đảm bảo tính đa dạng của nguồn dữ liệu, chúng tôi thu thập khoảng 100.000 bài báo từ 10 trang báo điện tử tiếng Việt phổ biến nhất dựa trên bảng xếp hạng của trang web alexa.com. Để đảm bảo sự đa dạng về nội dung, mỗi trang báo được chia thành 20 chủ đề lớn, và số lượng bài báo được thu thập cho mỗi chủ đề xấp xỉ bằng nhau. Hình 2 thống kê số lượng bài báo thu thập được theo các chủ đề.

Kết quả là chúng tôi đã thu thập được 1.011 CVT với 159.050 ngữ cảnh khác nhau từ dữ liệu các trang báo điện tử. Tuy nhiên, để phục vụ cho mục tiêu nghiên cứu của bài báo này, chúng tôi chỉ lọc ra được 5 CVT thỏa mãn các điều kiện huấn luyện và kiểm chứng mô hình phân lớp được nêu ở Phần 4.2.



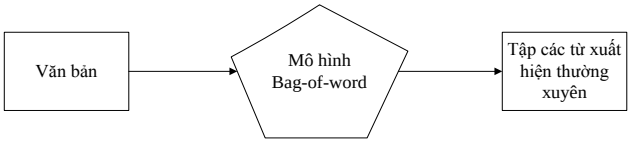
Hình 2. *Số lượng bài báo thu thập được theo các chủ đề*

3. Các phương pháp biểu diễn ngữ cảnh của CVT

Để xử lý sự nhập nhằng do một CVT có nhiều khai triển khác nhau, ngữ cảnh của CVT trong văn bản đóng vai trò quyết định trong việc lựa chọn khai triển đúng. Trong phần này, chúng tôi trình bày hai mô hình biểu diễn ngữ cảnh: Bag-of-words và Doc2vec.

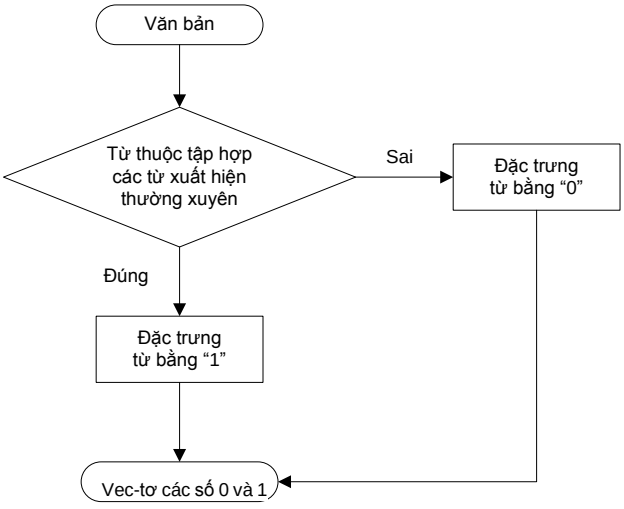
3.1. Mô hình Bag-of-words

Mô hình Bag-of-words (Hình 3) là một phương pháp biểu diễn văn bản đơn giản thường được sử dụng trong xử lý ngôn ngữ tự nhiên và tìm kiếm thông tin. Trong mô hình này, một văn bản được biểu diễn như một tập hợp (gọi là "túi") các từ xuất hiện trong văn bản, không quan tâm đến ngữ pháp và thứ tự xuất hiện của các từ mà chỉ lưu lại tần suất xuất hiện của mỗi từ trong văn bản. Mô hình Bag-of-words thường được sử dụng trong các phương pháp phân loại văn bản khi mà tần suất xuất hiện của từ được sử dụng như là một đặc trưng để huấn luyện một bộ phân lớp [4].



Hình 3. *Mô hình Bag-of-words*

Khi sử dụng mô hình này để biểu diễn văn bản, mỗi từ được biểu diễn bởi một số nhị phân tùy thuộc vào từ đó có thuộc tập hợp các từ xuất hiện thường xuyên hay không. Kết quả là văn bản đầu vào được biểu diễn bằng một vec-tơ gồm các số nhị phân 0 và 1 như thuật toán mô tả trong Hình 4.



Hình 4. *Thuật toán xác định đặc trưng nhị phân của văn bản*

Xét một ví dụ với hai câu sau: i) "*Liveshow tháng 1/2016 cũng đồng thời là liveshow cuối cùng của chương trình BHYT khép lại sau 4 năm kiên trì tạo dựng một thói quen thưởng thức âm nhạc cho công chúng.*", và ii) "*Mặt khác, sẽ có rất nhiều trường hợp phải đăng ký khai sinh, nhập hộ khẩu và đề nghị cấp thẻ BHYT diễn ra trong ngày trong khi cán bộ, công chức phải thực hiện nhiều nhiệm vụ khác nhau.*". Hai câu này là ngữ cảnh của CVT "BHYT" đối với hai khai triển lần lượt là "bài hát yêu thích" và "bảo hiểm y tế". Với giả định là các từ xuất hiện thường xuyên trong dữ liệu gồm {liveshow, thẻ, khai, sinh, bệnh, nhân, âm, nhạc, ca, khúc, hộ, khẩu} thì đặc trưng nhị phân của hai câu trên lần lượt là: i) [1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0], và ii) [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 1, 1, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]. Có thể thấy là mô hình Bag-of-words làm mất đi thông tin ngữ nghĩa có thể được suy diễn từ thứ tự xuất hiện của các từ trong câu, và vec-tơ đặc trưng biểu diễn câu thường rất thưa thớt (nghĩa là có nhiều thành phần bằng 0).

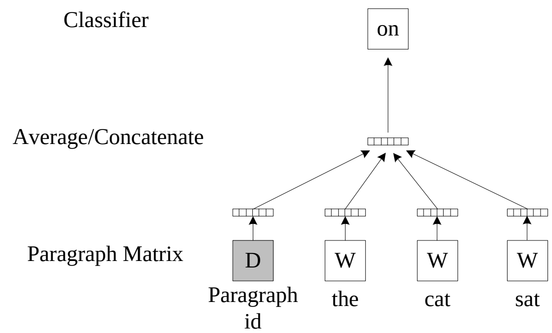
3.2. Mô hình Doc2vec

Vào năm 2013, các nhà nghiên cứu tại Google đã đề xuất mô hình Word2vec [5] học được cách biểu diễn phân tán của một từ trong không gian vec-tơ trong khi vẫn có thể giữ lại ngữ nghĩa của từ đó. Sau đó, mô hình Doc2vec [6] được mở rộng từ Word2vec để có thể tính toán biểu diễn phân tán cho câu, đoạn văn, hay cả văn bản. Doc2vec đã cho hiệu quả tốt hơn các phương pháp biểu diễn văn bản truyền thống trong một thử nghiệm phân loại văn bản và phân tích ngữ nghĩa [6]. Mô hình này đang thu hút sự chú ý của cộng đồng nghiên cứu về xử lý ngôn ngữ tự nhiên trong những năm gần đây.

Word2vec sử dụng một biểu diễn phân tán cho mỗi từ. Giả sử chúng ta dùng một vec-tơ từ với vài trăm chiều để biểu diễn. Mỗi từ sẽ được biểu diễn bởi một tập các trọng số tương ứng với các thành phần của vec-tơ. Như vậy, biểu diễn của một từ sẽ phân bố trên tất cả các thành phần của vec-tơ, và mỗi thành phần trong vec-tơ đều góp phần vào định nghĩa của nhiều từ. Hình 5 minh họa ý tưởng của Word2vec, trong đó các thành phần của vec-tơ từ giả thiết đã được gán nhãn để cho dễ hiểu (các chữ màu xanh da trời), mặc dù trong thuật toán gốc không có sự hiện diện của các nhãn này. Có thể thấy rằng mỗi một vec-tơ kết quả biểu diễn một từ (chữ màu xanh lá cây) theo một cách trừu tượng ý nghĩa của từ đó.

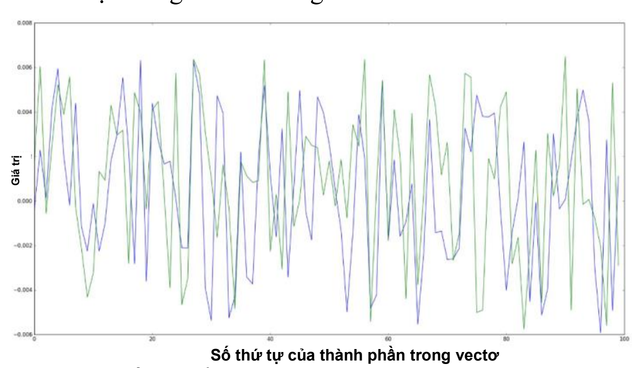
***Hình 5.** Biểu diễn từ bằng vec-tơ trong Word2vec [5]*

Doc2vec thừa kế ý tưởng của Word2vec và xây dựng thêm ma trận đoạn (paragraph matrix) (Hình 6). Các từ vẫn được ánh xạ thành các vec-tơ như trên. Thêm vào đó, mỗi đoạn (hay cả văn bản, nếu làm việc ở mức văn bản) cũng được ánh xạ thành một vec-tơ. Các vec-tơ từ là các cột của ma trận W, và các vec-tơ đoạn là các cột của ma trận D. Thay đổi duy nhất so với việc học vec-tơ từ là vec-tơ đoạn được ghép nối (hay lấy trung bình) với các vec-tơ từ, và các vec-tơ này được huấn luyện để tiên đoán được từ tiếp theo trong một ngữ cảnh (trong Hình 6 là ngữ cảnh gồm 3 từ "the", "cat", và "sat" dùng để tiên đoán từ thứ tư là "on"). Các ngữ cảnh có độ dài cố định và được lấy từ một cửa sổ trượt trên đoạn văn bản. Mỗi vec-tơ đoạn được dùng chung cho các cửa sổ sinh ra từ cùng một đoạn văn bản, nhưng không được sử dụng cho các đoạn khác. Ngược lại, các vec-tơ từ dùng chung cho tất cả các đoạn.



***Hình 6.** Mô hình học vec-tơ đoạn trong Doc2vec [6]*

Trong Doc2vec, mỗi đoạn văn bản được gắn một định danh (paragraph id) và được ánh xạ thành một vec-tơ đoạn thông qua ma trận D. Nếu áp dụng ở mức câu thì vec-tơ đoạn này có thể xem là vec-tơ đặc trưng của câu. Cũng với hai câu trong ví dụ ở Phần 3.1, chúng tôi tìm được hai vec-tơ đặc trưng tương ứng có các thành phần (tọa độ) biểu diễn dưới dạng đồ thị như trong Hình 7, trong đó câu i) là đường màu xanh da trời và câu ii) là đường màu xanh lá cây. Có thể nhận xét rằng, trái với Bag-of-words, vec-tơ đặc trưng biểu diễn câu dùng Doc2vec thường khá dày đặc (nghĩa là có nhiều thành phần khác 0). Tuy nhiên, việc dùng Doc2vec cũng làm cho số chiều của vec-tơ đặc trưng khá lớn so với Bag-of-words. Trong bài báo này, chúng tôi cố định số chiều của vec-tơ đặc trưng câu khi dùng Doc2vec là 100.



***Hình 7.** Đồ thị biểu diễn vec-tơ đặc trưng câu dùng Doc2vec*

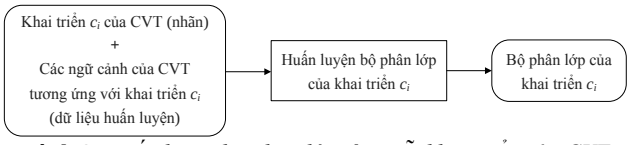
4. Khử nhập nhằng trong khai triển CVT dùng tiếp cận học máy

Để khử nhập nhằng khi khai triển một CVT, chúng tôi chọn tiếp cận học máy để đưa ra lựa chọn khai triển tối ưu trong tập hợp các khai triển có thể của CVT đó. Ở đây bài toán khử nhập nhằng có thể xem như bài toán phân lớp. Ưu điểm của việc sử dụng tiếp cận học máy là: một mô hình phân lớp nếu được huấn luyện trên một tập dữ liệu đủ lớn

sẽ có khả năng phân lớp chính xác đối với các dữ liệu mới (gọi là dữ liệu kiểm chứng) không nằm trong tập dữ liệu huấn luyện, hay còn gọi là có tính tổng quát hóa cao. Nhược điểm của nó là dữ liệu huấn luyện phải đủ lớn và có độ bao phủ tốt để có thể tạo nên bộ phân lớp đáng tin cậy. Mặc dù có nhiều mô hình phân lớp, chúng tôi chọn bộ phân lớp Naïve Bayes cho nghiên cứu này do tính phổ dụng và dễ cài đặt của nó. Các phần tiếp theo sẽ trình bày tiếp cận học máy với bộ phân lớp Naïve Bayes để khử nhập nhằng trong khai triển CVT và các kết quả thực nghiệm với hai phương pháp biểu diễn ngữ cảnh mô tả ở Phần 3.

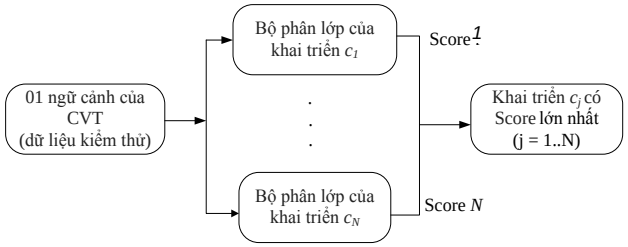
4.1. Tiếp cận học máy

Tiếp cận học máy (cụ thể là học có giám sát) gồm 2 giai đoạn: huấn luyện và phân lớp. Giai đoạn huấn luyện được trình bày trong Hình 8. Đối với một CVT, mỗi khai triển của nó sẽ có một bộ phân lớp tương ứng cần được ước lượng. Để huấn luyện bộ phân lớp của một khai triển, các ngữ cảnh của CVT (tức là các câu chứa CVT) ứng với khai triển này được dùng làm dữ liệu huấn luyện.



Hình 8. *Huấn luyện bộ phân lớp cho mỗi khai triển của CVT*

Hình 9 mô tả giai đoạn phân lớp. Dữ liệu đầu vào là một ngữ cảnh nào đó của CVT (gọi là dữ liệu kiểm chứng, vì không nằm trong tập dữ liệu huấn luyện). Chúng ta cần tìm khai triển tối ưu cho CVT trong ngữ cảnh này. Khai triển tối ưu được định nghĩa là khai triển có điểm đánh giá (score) cao nhất trong tập các khai triển của CVT này. Điểm đánh giá của mỗi khai triển được xác định nhờ bộ phân lớp của khai triển đó.



Hình 9. *Phân lớp một ngữ cảnh nào đó của CVT vào khai triển tối ưu*

4.2. Bộ phân lớp Naïve Bayes

Bộ phân lớp Naïve Bayes được xây dựng dựa trên xác suất nhờ áp dụng định lý Bayes [4]. Bài toán khử nhập nhằng dùng bộ phân lớp Naïve Bayes được phát biểu như sau: cho dữ liệu đầu vào d gồm CVT và một ngữ cảnh nào đó của nó, khai triển tối ưu của CVT được định nghĩa là khai triển $\hat{c}$ sở hữu xác suất có điều kiện của khai triển đối với dữ liệu đầu vào đạt giá trị cực đại, nghĩa là

$$\hat{c} = \underset{c \in C}{\operatorname{argmax}} P(c|d), \qquad (1)$$

trong đó c là một khai triển trong tập hợp C các khai triển có thể của CVT. Như vậy, điểm đánh giá của khai triển c chính là $P(c|d)$, được tính nhờ bộ phân lớp Naïve Bayes. Trong nghiên cứu này, các thử nghiệm huấn luyện và phân lớp với bộ phân lớp Naïve Bayes sử dụng cài đặt của scikit-learn toolkit [9].

4.3. Chuẩn bị dữ liệu

Trước khi bắt đầu thử nghiệm, chúng tôi loại bỏ các CVT có dữ liệu không thoả mãn các điều kiện để huấn luyện và kiểm chứng mô hình phân lớp như sau:

- Số lượng dữ liệu huấn luyện nhỏ hơn hoặc bằng 5 mẫu. Điều này là do nếu dữ liệu huấn luyện quá ít thì không thể huấn luyện được mô hình phân lớp một cách tin cậy bằng thuật toán học máy.
- Dữ liệu huấn luyện quá thiên lệch về một khai triển nào đó của CVT, cụ thể là một khai triển có số lượng mẫu huấn luyện nhiều gấp 20 lần một khai triển khác. Điều này để đảm bảo kết quả kiểm chứng phản ánh chính xác năng lực khử nhập nhằng của bộ phân lớp.

Sau quá trình lọc dữ liệu, chúng tôi thu được 5 CVT thoả mãn hai điều kiện trên là: "BHYT", "NS", "PTTH", "THA", và "KH". Số lượng này ít hơn hẳn 1.011 CVT đã thu thập được ở phần 2.2. Điều này là do, đối với hầu hết các CVT, lượng dữ liệu ngữ cảnh thu được hoặc rất ít, hoặc phân bố rất không đồng nhất giữa các khai triển. Bảng 1 thống kê số mẫu dữ liệu dùng để huấn luyện bộ phân lớp của các CVT này cho mỗi khai triển. Chú ý là số lượng mẫu dữ liệu dùng để kiểm chứng trong phần 4.4 cũng bằng với số lượng mẫu dữ liệu dùng để huấn luyện bộ phân lớp.

4.4. Kết quả thực nghiệm

Chúng tôi đã tiến hành các thử nghiệm huấn luyện và kiểm chứng bộ phân lớp Naïve Bayes với hai phương pháp biểu diễn ngữ cảnh: Bag-of-words và Doc2vec. Bảng 2 thể hiện kết quả độ chính xác khi khai triển CVT. Có thể thấy rằng Bag-of-words cho tỉ lệ khai triển chính xác cao hơn hoặc bằng Doc2vec trong mọi trường hợp. Độ chính xác trung bình của Bag-of-wordslà 86,0% và của Doc2vec là 79,7%.

5. Bàn luận

Trong học máy dựa trên một mô hình xác suất như bộ phân lớp Naïve Bayes thì thông thường, số lượng mẫu huấn luyện càng nhiều thì hiệu năng của mô hình phân lớp càng cao. Từ Bảng 2 có thể thấy rằng, với bài toán khử nhập nhằng trong khai triển CVT bằng tiếp cận học máy thống kê, thì mức độ gần gũi (hay khác nhau) của lĩnh vực kinh tế-xã hội mà các khai triển thuộc về, cũng đóng vai trò quan trọng không kém lượng dữ liệu huấn luyện. Nếu các lĩnh vực không liên quan đến nhau nhiều, ví dụ "bài hát yêu thích" (âm nhạc) và "bảo hiểm y tế" (y tế) đối với CVT "BHYT" hay "thi hành án" (pháp luật) và "tăng huyết áp" (sức khỏe) đối với CVT "THA", thì việc khử nhập nhằng bằng các phương pháp trên cho tỉ lệ khai triển chính xác khá cao (đều trên 90%), cho dù nhiều hay ít dữ liệu huấn luyện, vì ngữ cảnh của CVT thể hiện được vai trò quan trọng của nó trong việc phân lớp. Ngược lại, khi các lĩnh vực gần và liên quan đến nhau nhiều thì ngữ cảnh của CVT không còn thể hiện vai trò lớn trong việc phân lớp nữa, dẫn đến tỉ lệ khai triển chính xác thấp hơn nhiều (đều chỉ trên 70%), cho dù nhiều hay ít dữ liệu huấn luyện (ví dụ "nghệ sĩ" và "nhạc sĩ" đối với CVT "NS" hay "khoa học" và "kế hoạch" đối với CVT

"KH").

Bảng 1. *Thống kê lượng dữ liệu huấn luyện các bộ phân lớp (theo thứ tự giảm dần của cột "Tổng số mẫu huấn luyện")*

STT	CVT	Khai triển	Số mẫu huấn luyện	Tổng số mẫu huấn luyện
1	BHYT	bài hát yêu thích	52	295
		bảo hiểm y tế	243	
2	NS	nghệ sĩ	44	99
		nhạc sĩ	55	
3	PTTH	phát thanh truyền hình	26	49
		phổ thông trung học	23	
4	THA	thi hành án	17	29
		tăng huyết áp	12	
5	KH	khoa học	7	17
		kế hoạch	10	

Bảng 2. *Độ chính xác khi khai triển CVT dùng 2 mô hình biểu diễn ngữ cảnh: Bag-of-words và Doc2vec (số mẫu dữ liệu dùng để kiểm chứng bằng với số mẫu dữ liệu dùng để huấn luyện)*

STT	CVT	Khai triển	Bag-of-words	Doc2vec	Độ chính xác trung bình
1	BHYT	bài hát yêu thích	98,0%	98,0%	**98,0%**
		bảo hiểm y tế			
2	NS	nghệ sĩ	77,5%	74,5%	**76,0%**
		nhạc sĩ			
3	PTTH	phát thanh truyền hình	83,7%	69,4%	**76,5%**
		phổ thông trung học			
4	THA	thi hành án	93,3%	90,0%	**91,7%**
		tăng huyết áp			
5	KH	khoa học	77,8%	66,7%	**72,2%**
		kế hoạch			
Trung bình			**86,0%**	**79,7%**	**82,9%**

Việc Bag-of-words cho tỉ lệ khai triển chính xác cao hơn hoặc bằng Doc2vec trong cả 5 thử nghiệm khai triển CVT là một kết quả trái với dự đoán ban đầu của các tác giả nếu xem xét các ưu điểm của Doc2vec so với Bag-of-words đã trình bày ở Phần 3. Do hạn chế cả về thời gian tiến hành thực nghiệm cũng như kích thước dữ liệu dùng trong các thử nghiệm, nên chúng tôi chưa thể đưa ra giải thích thỏa đáng cho kết quả này. Điều này sẽ được xem xét trong một nghiên cứu tiếp theo.

6. Kết luận

Chúng tôi đã trình bày hai phương pháp biểu diễn ngữ cảnh dùng để khử nhập nhằng trong khai triển CVT, một phương pháp truyền thống là Bag-of-words và một phương pháp mới được đề xuất gần đây là Doc2vec. Tiếp cận học máy thống kê dùng bộ phân lớp Naïve Bayes cũng được mô tả và thử nghiệm để kiểm chứng hiệu quả của hai phương pháp biểu diễn ngữ cảnh này. Kết quả thực nghiệm cho thấy Bag-of-words cho tỉ lệ khai triển chính xác cao hơn Doc2vec trung bình khoảng 6%. Độ chính xác trung bình của các thử nghiệm khai triển CVT sử dụng bộ phân lớp Naïve Bayes là 82,9%. Trong tương lai, chúng tôi sẽ thử nghiệm trên các bộ dữ liệu CVT lớn hơn, cũng như thử nghiệm trên các mô hình phân lớp khác để có đánh giá toàn diện hơn về hiệu quả của Doc2vec trong bài toán biểu diễn ngữ cảnh của CVT.

TÀI LIỆU THAM KHẢO

[1] Richard Sproat, Alan Black, Stanley Chen, Shankar Kumar, MariOstendorf, and Christopher Richards, "Normalization of Non-Standard Words", *Computer Speech and Language*, 15(3), 2001, pp. 287-333.

[2] Thu-Trang Thi Nguyen, Thanh Thi Pham, Do-Dat Tran, *A Method for Vietnamese Text Normalization to Improve the Quality of Speech Synthesis*, Proceedings of International Symposium on Information and Communication Technology (SoICT), Vietnam, 2010.

[3] Dinh Anh Tuan, Phi Tung Lam, Phan Dang Hung, *A Study of Text Normalization in Vietnamese for Text-to-Speech System*, Proceedings of Oriental COCOSDA Conference, China, 2012.

[4] Daniel Jurafsky, James H. Martin, *Speech and Language Processing, 2nd edition*, Prentice Hall, 2008.

[5] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, and Jeffrey Dean, *Distributed Representations of Words and Phrases and their Compositionality*, Proceedings of Conference on Neural Information Processing Systems (NIPS), USA, 2013.

[6] Quoc Le, Tomas Mikolov, *Distributed Representations of Sentences and Documents*, Proceedings of the 31st International Conference on Machine Learning, Beijing, China, 2014.

[7] Nguyen Nho Tuy, Phan Huy Khanh, *Developing Database of Vietnamese Abbreviations and Some Applications*, Proceedings of Second International Conference on Nature of Computation and Communication, RachGia, Vietnam, 2016.

[8] Paul Taylor, *Text-to-Speech Synthesis*, Cambridge University Press, 2009.

[9] Fabian Pedregosa et al., "Scikit-learn: Machine Learning in Python", *Journal of Machine Learning Research*, 12(Oct), 2011, pp. 2825-2830.

(BBT nhận bài: 02/02/2017, hoàn tất thủ tục phản biện: 02/03/2017)