TẠP CHÍ KHOA HỌC & CÔNG NGHỆ ĐẠI HỌC DUY TÂN
DTU Journal of Science and Technology 02(33) (2019) 03-11

TRƯỜNG ĐẠI HỌC DUY TÂN
DUYTAN UNIVERSITY

Một biến thể mô hình skip-thought và ứng dụng cho bài toán tìm kiếm câu đồng nghĩa trong văn bản tiếng Việt

A variant of skip-thought model and its application in finding semantically similar sentences in Vietnamese texts

Nguyễn Viết Học[a,*], Nguyễn Kim Tuấn[b], Nguyễn Thị Thu Thuỷ[c]
Viet Hoc Nguyen, Kim Tuan Nguyen, Thu Thuy Nguyen

[a]*Học viện Kỹ thuật Quân sự, 236 Hoàng Quốc Việt, Hà Nội, Việt Nam*
Military Technical Academy, 236 Hoang Quoc Viet, Hanoi, Vietnam
[b]*Trường Đại học Duy Tân, 03 Quang Trung, Đà Nẵng, Việt Nam*
Duy Tan University, 03 Quang Trung, Danang, Vietnam
[c]*Trường Cao đẳng Kinh tế Kỹ thuật Quảng Nam, 431 Hùng Vương, Quảng Nam, Việt Nam*
Quang Nam College of Economics and Technology, 431 Hung Vuong, Quangnam, Vietnam

(Ngày nhận bài: 18/02/2019, ngày phản biện xong: 19/02/2019, ngày chấp nhận đăng: 01/04/2019)

Tóm tắt

Đánh giá mức độ đồng nghĩa giữa các câu là nhiệm vụ trọng tâm để thực hiện mục tiêu hiểu ngôn ngữ tự nhiên - một trong những thách thức lớn trong xử lý ngôn ngữ tự nhiên. Sử dụng Deep Learning cho bài toán so khớp ngữ nghĩa của câu đã thay đổi cách tiếp cận, khắc phục được những khó khăn mà các phương pháp truyền thống trước đây gặp phải. Nhiều thuật toán đã được phát triển để có thể biểu diễn câu bằng một vector với số chiều cố định, việc này giúp cho việc xử lý ngôn ngữ tự nhiên dựa trên câu trở nên dễ dàng và hiệu quả hơn. Các phương pháp trên đều trích rút đặc trưng thủ công hoặc sử dụng các thuật toán học có giám sát nhưng với không gian ngữ liệu ngày càng phong phú, các phương pháp này tỏ ra không còn hiệu quả. Điều đó là động lực để ra đời các phương pháp học không giám sát, tận dụng sức mạnh tính toán của thiết bị hiện nay. Skip-thought là một trong những mô hình Deep Learning điển hình cho việc sử dụng thuật toán học không giám sát trong xử lý ngôn ngữ tự nhiên. Song Skip-thought có kiến trúc phức tạp, yêu cầu về phần cứng cao, do đó đã có một vài nghiên cứu nhằm cải thiện hiệu năng của mô hình. Trong nghiên cứu của mình, tác giả thử nghiệm Skip-thought rút gọn, ứng dụng vào bài toán tìm kiếm câu đồng nghĩa trong văn bản tiếng Việt, đánh giá trên bộ ngữ liệu VnPara. Kết quả đạt được vượt trội hơn hẳn so với phương pháp được công bố bởi tác giả của VnPara - Ngô Xuân Bách.

Từ khóa: Xử lý ngôn ngữ tự nghiên, học sâu, mô hình Skip-thought, câu đồng nghĩa.

Abstract

Evaluation of semantic similarity is the central of understanding natural languages - one of the major challenges in natural language processing. Using Deep Learning for the semantic matching has changed the approach and overcome the difficulties that traditional methods had to encounter in the past. Many algorithms have been developed for presenting a sentence as a fixed dimension vector which makes the natural language processing easier and more effective. The above methods have either extracted manually the characteristics or used supervised learning algorithms; however, they would be no longer effective with enormous data. That is the motivation for the using of unsupervised methods to take advantage of the computing power of today's devices. Skip-thought is one of the typical Deep Learning models for using unsupervised learning algorithms in natural language processing. However Skip-thought has a complex architecture and high hardware requirements so there have been some researches to improve the performance of

Email: nkimtuan@gmail.com

the model. This paper presents reduced Skip-thought for finding semantically similar sentences in Vietnamese texts and evaluated it on Vnpara. The results show the better accuracy achievement in comparison with Ngo Xuan Bach's method on the same corpus.

Keywords: Natural Language Processing, Deep Learning, Skip-thought model, Sematic Similarity.

1. Đặt vấn đề

Đánh giá độ tương đồng về ngữ nghĩa trong xử lý ngôn ngữ tự nhiên có rất nhiều ứng dụng trong thực tiễn, ví dụ như xác định quan hệ giữa các câu truy vấn trong máy tìm kiếm, tạo từ khóa cho quảng cáo. Trong y học, có thể kể đến các ứng dụng như phân cụm gen, tìm kiếm gen bệnh, biểu diễn gen. Trong xử lý ngôn ngữ tự nhiên, đánh giá độ tương đồng về ngữ nghĩa rất có ý nghĩa cho các bài toán: tóm tắt văn bản, phân loại văn bản, tìm kiếm thông tin. Bài toán nhóm tác giả đang quan tâm là tìm kiếm câu đồng nghĩa trong văn bản, phục vụ tìm kiếm văn bản theo ngữ nghĩa câu truy vấn.

Đánh giá độ tương đồng ngữ nghĩa theo phương pháp trước đây có thể chia làm các hướng chính như sau: dựa vào kho ngữ liệu, dựa vào tri thức. Các phương pháp này chủ yếu dựa trên sự tương đồng về ngữ nghĩa của các từ trong câu. Tìm kiếm sự đồng nghĩa giữa các từ là một bước quan trọng để thực hiện các nhiệm vụ tiếp theo: tìm kiếm sự đồng nghĩa giữa câu, đoạn văn, văn bản [1]. Từ đồng nghĩa là những từ tương đồng với nhau về nghĩa, khác nhau về âm thanh và có phân biệt với nhau về một vài sắc thái ngữ nghĩa hoặc sắc thái phong cách, hoặc đồng thời cả hai. Dựa vào tri thức là những phương pháp sử dụng thông tin trích xuất từ những mạng từ [2, 3, 4], từ điển bách khoa [5], từ điển đồng nghĩa, cây ngữ nghĩa [6]. Khó khăn gặp phải của nhóm phương pháp này là việc xây dựng mạng từ, từ điển bách khoa, từ điển đồng nghĩa hay cây ngữ nghĩa tốn rất nhiều công sức và chi phí, thêm vào đó là kết quả của phép đo sẽ bị ảnh hưởng nhiều bởi những yếu tố trên. Đồng thời, nếu chỉ so sánh về mặt từ thì sẽ mất ý nghĩa ngữ pháp trong câu. Dựa vào kho ngữ liệu để xác định mức độ đồng nghĩa là những phương pháp sử dụng thông tin thu nhận được từ những kho ngữ liệu lớn từ nhiều nguồn điển hình là thuật toán LSA [7, 8], HAL [9]. Nhược điểm của LSA và HAL là tạo thành không gian vector rất thưa, chưa biểu diễn được câu trực tiếp mà phải qua tính toán thông qua các từ cấu thành câu. Ngoài ra, còn có các cách tiếp cận khác sử dụng các phép đo trên không gian vector của từ để đánh giá độ đồng nghĩa [10]. Những phương pháp kể trên có cùng nhược điểm là chưa tính đến thứ tự từ trong câu. Ví dụ: "con cáo nhảy qua đầu con gà" và "con gà nhảy qua đầu con cáo" sẽ được đánh giá là giống nhau hoàn toàn với các phương pháp cũ.

Trong cộng đồng xử lý ngôn ngữ tiếng Việt, mặc dù vai trò của đánh giá mức độ tương đồng ngữ nghĩa của câu rất quan trọng nhưng các nghiên cứu có tính hệ thống còn rất hạn chế do thiếu sự đầu tư, hạn chế về tài nguyên và tính kế thừa. Cho đến thời điểm này, chỉ duy nhất có phương pháp của tác giả Ngô Xuân Bách và cộng sự đưa ra nhằm xác định câu đồng nghĩa dựa vào kết hợp các độ đo khác nhau [11]. Nhóm tác giả sử dụng 9 độ đo khác nhau: Levenshtein, Jaro-Winkler, Manhattan, Euclidean, cosine, n-gram (n = 3), hệ số so khớp, hệ số Dice và hệ số Jaccard để tính đặc trưng cặp câu đầu vào. Với mỗi cặp câu đầu vào, tác giả xây dựng 7 cặp câu với các mức độ trừu tượng khác nhau:

1- Giữ nguyên các âm tiết và thứ tự trong câu.

2- Âm tiết được thay thế bằng từ

3- Từ được thay thế bằng loại từ

4- Giữ lại từ loại là danh từ, động từ, tính từ, giữ nguyên thứ tự xuất hiện

5- Như 4 nhưng chỉ giữ lại danh từ

6- Như 4 nhưng chỉ giữ lại động từ

7- Như 4 nhưng chỉ giữ lại tính từ

Với mỗi lần áp dụng 9 độ đo cho một trong

những cặp câu trên sẽ cho ra tập đặc trưng mô tả cặp câu. Tác giả kết hợp các tập đặc trưng và đánh giá bằng các thuật toán phân loại: KNN, SVM, Maximum Entropy, Naive Bayes để lựa chọn ra các tập đặc trưng mô tả cặp câu đầu vào tốt nhất. Kết quả trung bình đặt được: Accuracy = 89.10%, F1-score = 86.77%.

Deep Learning là công cụ mạnh mẽ để xử lý các bài toán phức tạp mà các phương pháp học máy trước đó chưa thể giải quyết được. Một số mô hình Deep Learning ứng dụng rộng rãi như mạng nhân chập CNN, mạng hồi quy RNN, mạng hồi quy GRU, mạng hồi quy LSTM, mạng nơron sâu DNN. Trong xử lý ngôn ngữ tự nhiên, các mạng hồi quy được dùng nhiều hơn, ứng dụng trong việc mô hình hóa ngôn ngữ, phát hiện đoạn văn cùng nghĩa, sinh văn bản vì mạng hồi quy lưu giữ được sự liên kết của các thành phần trong câu, đồng thời không bị hạn chế bởi số độ dài của câu. Thành công của mạng hồi quy trong xử lý ngôn ngữ tự nhiên phải kể đến mô hình Encoder-Decoder, được nhóm nghiên cứu của Google Brain sử dụng trong dịch máy [12]. Trong mô hình Encoder-Decoder, Encoder và Deocder là hai mạng nơron hồi quy hoạt động độc lập. Encoder nhận đầu vào là một câu với độ dài không cố định, ánh xạ câu sang một biểu diễn vector với số chiều cố định; Decoder ánh xạ biểu diễn của Encoder sang câu mục tiêu. Encoder và Decoder sẽ được huấn luyện cùng nhau để cực đại hóa xác suất điều kiện dự đoán ra câu mục tiêu khi cho một câu đầu vào. Trên ý tưởng đó, kết hợp với kết quả đạt được của mô hình Skip-gram Word2vec [13]. Thay vì sử dụng một bộ Encoder và một bộ Decoder, Skip-thought sử dụng một bộ Encoder nhận đầu vào là một câu cho ra trạng thái ẩn và hai bộ Decoder cùng sử dụng trạng thái ẩn này để dự đoán hai câu mục tiêu đầu ra trước và sau, cùng ngữ cảnh với câu đầu vào Encoder. Vì thế, nếu những câu nằm trong cùng ngữ cảnh thì sẽ có xác suất gần nghĩa với nhau cao.

Mô hình này đã được áp dụng thành công vào ngôn ngữ tiếng Anh và cho kết quả vượt trội. Tuy nhiên, các thử nghiệm cho văn bản tiếng Việt chưa được cộng đồng nghiên cứu quan tâm. Trong bài báo này, chúng tôi sử dụng Skip-thought huấn luyện trên bộ ngữ liệu thu thập từ các nguồn khác nhau trên Internet (vnexpress.net, vnthuquan.net, vietnamnet.vn, vanbanphapluat.net…). Sau đó đánh giá bằng các độ đo, kết hợp với một số mô hình học máy và so sánh với phương pháp của nhóm tác giả Ngô Xuân Bách trên cùng bộ ngữ liệu Vnpara [11].

Các nội dung tiếp theo của bài báo gồm: mục 2 giới thiệu mô hình Skip-thought và ứng dụng trong bài toán tìm câu đồng nghĩa; mục 3 là một số thử nghiệm và đánh giá mô hình qua bộ ngữ liệu được công bố trong [11]; mục 4 là một số kết luận.

2. Mô hình Skip-thought

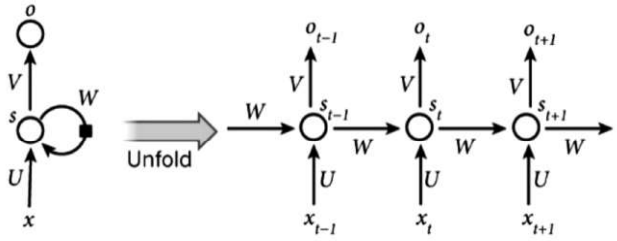
2.1. Mạng hồi quy

Việc phát triển các thuật toán học máy cho lĩnh vực xử lý ngôn ngữ tự nhiên, cụ thể hơn là hiểu ngôn ngữ tự nhiên đã được chú ý và phát triển từ lâu. Những năm gần đây, mạng hồi quy, mạng nhân chập được ứng dụng nhiều để thực hiện việc ánh xạ các vector từ sang vector câu. Những thuật toán này dựa trên các dữ liệu được gán nhãn, tính lỗi lan truyền ngược để cập nhật các trọng số. Trong xử lý ngôn ngữ tự nhiên, các mạng hay được dùng như RNN, GRU, LSTM. Trong phần này, chúng tôi điểm qua về RNN, GRU. LSTM là biến thể của GRU song được cho là tốn kém chi phí tính toán nên chúng tôi không đề cập.

A. Recurrent Neural Network (RNN)

Ý tưởng chính của RNN là sử dụng chuỗi các thông tin liên tục nhau. Trong các mạng nơ-ron truyền thống tất cả các đầu vào và cả đầu ra là độc lập với nhau. Tức là chúng không liên kết thành với nhau về mặt thời gian. Các mô hình như vậy không phù hợp cho các bài toán mà dữ liệu là tuần tự. RNN được gọi là hồi quy (Recurrent) bởi lẽ chúng thực hiện cùng một tác vụ cho tất cả các phần tử của một chuỗi với đầu ra phụ thuộc vào

cả các phép tính trước đó. RNN có khả năng nhớ những trạng thái trước đó của dữ liệu, do đó có thể biểu diễn được mối liên hệ giữa các dữ liệu đầu vào. Mô hình RNN như sau:



Hình 1. Mô hình RNN

Trong đó:

$$s_t = f(Ux_{t-1} + Ws_{t-1})$$

$$o_t = softmax(Vs_t)$$

Hàm f ở đây là một hàm phi tuyến, thông thường là hàm σ hoặc hàm *ReLu*.

B. GRU

Vấn đề mà mạng RNN chưa giải quyết được là những phụ thuộc xa của đầu vào và nguy cơ biến mất đạo hàm. GRU cải tiến RNN bằng cách thêm vào cổng điều khiển *cập nhật* và *quên*. Hai cổng này kiểm soát việc có cho phép thông tin của trạng thái trước đi qua hay không. Chính vì thế GRU có thể lưu giữ những thông tin từ những trạng thái ở rất xa trạng thái hiện tại và quên những thông tin trạng thái không quan trọng. Mô hình toán học của GRU như sau:

$$r_t = \sigma(U^{(r)}x_t + W^{(r)}s_{t-1})$$

$$z_t = \sigma(U^{(z)}x_t + W^{(z)}s_{t-1})$$

$$\bar{s}_t = \tanh(Wx_t + U(r_t \odot s_{t-1})$$

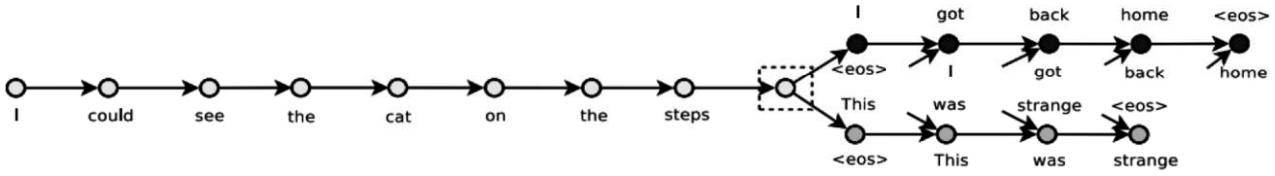
$$s_t = (1 - z_t) \odot s_{t-1} + z_t \odot \bar{s}_t$$

Trong đó, s_{t-1} là trạng thái của mạng tại thời điểm *t-1*. r_t là cổng quên, r_t quyết định sẽ quên thông tin nào trong s_{t-1}. z_t là cổng cập nhật, công thức tương tự với cổng quên, nhưng khác nhau ở trọng số và chức năng. Trạng thái hiện tại sử dụng cổng quên để xác định bao nhiêu thông tin được giữ lại trong $\bar{s}_t$. Thành phần cuối cùng là s_t sẽ quyết định bao nhiêu thông tin của trạng thái hiện tại sẽ truyền cho trạng thái sau. Nếu thành phần vector của z_t tiến về 1, thông tin trạng thái hiện tại sẽ được truyền nhiều hơn, nếu thành phần vector z_t tiến về 0, thông tin trạng thái hiện tại gần như được giữ lại và chỉ truyền thông tin của trạng thái trước đó *t-1*.

2.2. Skip-thought

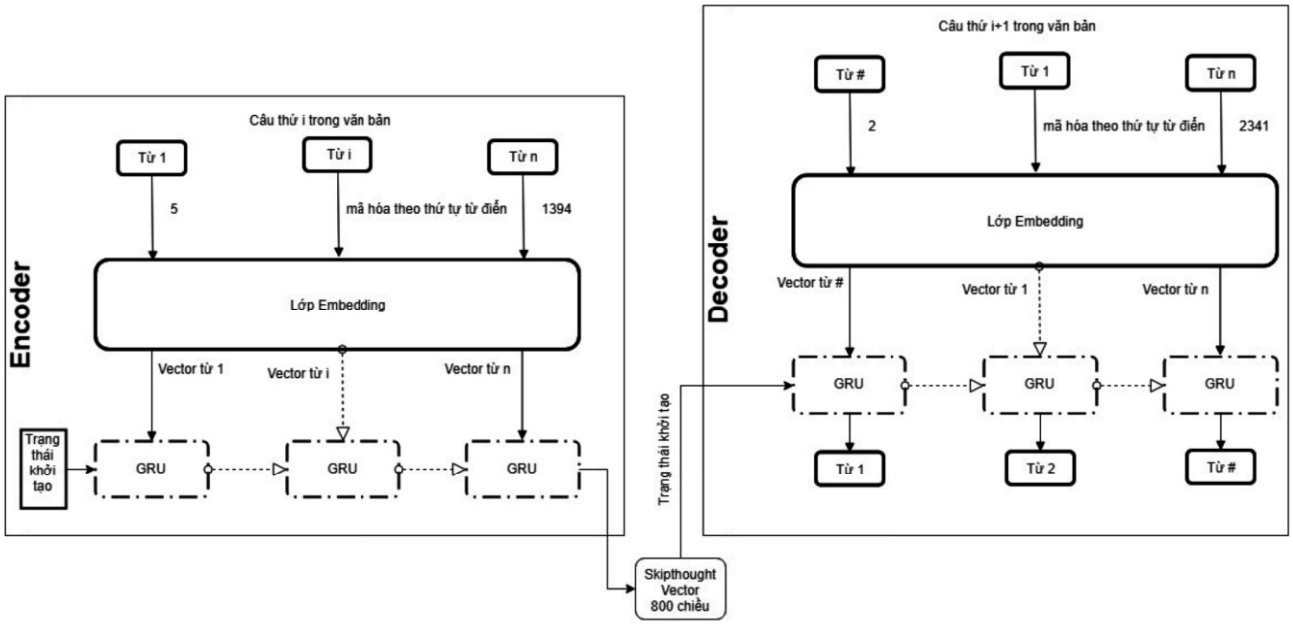
Trong công bố [14], Ryan Kiros đưa ra một mô hình học không giám sát. Lấy ý tưởng từ mô hình Skip-gram của Word2vec, thay vì dùng một từ để dự đoán các từ trong ngữ cảnh, tác giả sử dụng một câu để dự đoán các câu trong ngữ cảnh.

Skip-thought là một biến thể của mô hình Encoder-Decoder. Trong đó Encoder ánh xạ các từ trong câu vào không gian vector và Decoder tái xây dựng lại câu trong ngữ cảnh. Một số lựa chọn cho Encoder-Decoder như: RNN-RNN, GRU-GRU, LSTM-LSTM. Các mô hình Encoder-Decoder hiện đang được ứng dụng nhiều trong lĩnh vực dịch máy. Skip-thought mở rộng thêm một Decoder cho mô hình. Bây giờ, Encoder ánh xạ một câu vào không gian vector, Decoder thứ nhất (D1) xây dựng lại câu trước đó theo tuần tự câu trong văn bản, Decoder thứ hai (D2) xây dựng lại câu kế tiếp (Hình 2). Như vậy nếu hai câu nằm trong cùng một ngữ cảnh thì khả năng cao sẽ cùng biểu đạt nghĩa tương tự. Đầu ra của Decoder dưới dạng mô hình xác suất: $p(y_1, \ldots, y_T | x_1, \ldots, x_{T'})$.



Hình 2. Mô hình Skip-thought

Skip-thought có thể sử dụng RNN, GRU, LSTM. Như đã nêu trên, RNN gặp vấn đề về



Hình 3. Chi tiết mô hình Skip-thought rút gọn

phụ thuộc xa, sử dụng LSTM lại tốn kém chi phí tính toán, vì vậy sử dụng GRU sẽ cân bằng được hai yếu tố chi phí tính toán và hiệu quả mô hình. GRU giải quyết được vấn đề phụ thuộc xa và tiêu biến đạo hàm của RNN đồng thời đơn giản và hiệu quả gần tương đương với LSTM.

Encoder sẽ đọc từng từ, ánh xạ toàn bộ câu thứ i sang vector có chiều dài cố định s^i. Hai bộ Decoder sẽ sử dụng s^i để dự đoán các câu trong ngữ cảnh. Mô hình xác suất của Decoder D1 sẽ được viết dưới dạng:

$$p(y_1^{i-1}, \dots, y_T^{i-1}|s^i) = \prod_{t=1}^{T} p(y_t^{i-1}|y_1^{i-1}, \dots, y_{t-1}^{i-1}|s^i)$$

$$\mathrm{p}(y_t^{i-1}|y_1^{i-1}, \dots, y_{t-1}^{i-1}|s^i) \propto \exp(v_t s_t^{i-1})$$

Mô hình xác suất của Decoder D2 tương tự như D1, thay chỉ số i-1 bằng i+1. Trong đó v_t là hàng thứ t của ma trận V tương ứng với từ y_t. V là ma trận trọng số trước lớp softmax cuối cùng của Decoder.

Dùng hàm mất mát Cross Entropy đánh giá huấn luyện mô hình, hàm mục tiêu Skip-thought được viết gọn thành:

$$max_\theta \left(\sum_t logp_\theta(y_t^{i+1}|y_{<t}^{i+1}, s^i) + \sum_t logp_\theta(y_t^{i-1}|y_{<t}^{i-1}, s^i) \right)$$

Với θ là bộ các tham số mô hình cần tối ưu.

Vấn đề gặp phải là số lượng tham số mô hình lớn, do đó thời gian huấn luyện mô hình phải mất đến hàng tuần, hàng tháng. Đồng thời yêu cầu phần cứng cao, đòi hỏi phải có hệ thống chuyên dụng. Để khắc phục vấn đề đó, S.Tang cùng cộng sự đã rút gọn một Decoder (D1), kết quả cho thấy mô hình được cải thiện rõ rệt về mặt thời gian đồng thời trong một vài trường hợp độ chính xác đạt được cao hơn so với Skip-thought truyền thống [15]. Chi tiết mô hình như Hình 3. (Dưới đây, khi nhắc đến Skip-thought, tác giả ngầm định đề cập đến Skip-thought rút gọn của S.Tang)

Ví dụ: Ta có văn bản huấn luyện là *"Samsung sẽ bán Galaxy S7 và S7 edge tại Việt Nam vào ngày 18/3 tới đây. Hãng cũng kỳ vọng sẽ có doanh số bán ra bằng hoặc cao hơn so với các thị trường Mỹ và Australia. Theo một số hệ thống cửa hàng điện thoại di động, lượng đặt hàng trước của Galaxy S7 còn ấn tượng hơn nhiều thế hệ Galaxy S6 ra mắt năm ngoái".* Sau các bước tách câu, tách từ, và thay số, email, IP, ngày tháng, từ ít gặp hoặc không có trong từ điển Word2vec bằng ký tự "?", ta có dữ liệu đưa vào huấn luyện được thể hiện trong Bảng 2.

Bảng 2. Dữ liệu đầu vào Skip-thought

Encoder		
Hãng cũng kỳ_vọng sẽ có doanh_số bán ra bằng hoặc cao hơn so với các thị_trường Mỹ và Australia		
Decoder		
1	Input	*# Theo một số hệ_thống cửa_hàng điện_thoại di_động, lượng đặt hàng trước của Galaxy ? còn ấn_tượng hơn nhiều thế_hệ Galaxy ? ra mắt năm ngoái*
	Output	*Theo một số hệ_thống cửa_hàng điện_thoại di_động, lượng đặt hàng trước của Galaxy ? còn ấn_tượng hơn nhiều thế_hệ Galaxy ? ra mắt năm ngoái #*

Sau khi huấn luyện xong mô hình, chỉ giữ lại Encoder. Encoder có nhiệm vụ ánh xạ câu vào không gian ngữ nghĩa [16]. Trên không gian ngữ nghĩa, mỗi câu sẽ là một điểm, đánh giá đồng nghĩa của câu có thể dựa vào các độ đo khoảng cách, các điểm phân bố gần nhau sẽ tương đồng về mặt ngữ nghĩa. Ngoài ra có thể sử dụng các thuật toán học máy khác nhau để phân loại các điểm này như các dữ liệu thông thường khác.

3. Thử nghiệm và đánh giá

Nhóm tác giả xây dựng Skip-thought bằng ngôn ngữ lập trình Python sử dụng thư viện Keras, backend TensorFlow; chạy trên GPU Nvidia Geforce Tesla K80 do Google cung cấp miễn phí.

Bảng 3. Ảnh hưởng của số trạng thái ẩn đến mô hình

Mô hình	Số chiều trạng thái ẩn	Số tham số mô hình	Thời gian / epoch	Hàm mất mát sau 5 epochs
Skip-thought (Kiros)	400	13,550,506	20h	1.16
	600	19,895,306	28h	1.20
	800	26,960,106	42h	1.03
	1.200	43,249,706	72h	1.05
Skip-thought rút gọn	400	8,696,903	12p	0.91
	600	12,259,903	16p	0.82
	800	16,302,903	25p	0.76
	1.200	25,828,903	45p	0.71

Bảng 4. Ví dụ về mẫu ngữ liệu trong kho ngữ liệu VnPara

Câu	Nội dung	Nhãn
1	Trả lời câu hỏi này tôi xin nói lên suy nghĩ của mình về bóng đá nhà nghề từ đó suy ra bóng đá của ta hiện nay để các quan chức quản lý bóng đá chuyên nghiệp suy nghĩ rút kinh nghiệm.	0
2	Sự thực 100% đội bóng chuyên nghiệp Việt Nam hiện giờ không thể dùng doanh thu từ bóng đá (gồm tiền thưởng thành tích bản quyền truyền hình bán vé hoạt động thương mại) để tự nuôi sống mình khi thực tế nguồn thu này quá nhỏ và manh mún.	
1	Hơn 10 năm qua công nghệ thông tin đã trở thành ngành kinh tế mũi nhọn có tốc độ tăng trưởng và hiệu quả cao đóng góp trực tiếp gần 7% GDP của đất nước đồng thời có tác động lan tỏa thúc đẩy phát triển nhiều ngành nhiều lĩnh vực kinh tế - xã hội.	1
2	Hơn 10 năm qua công nghệ thông tin đã trở thành ngành kinh tế có tốc độ tăng trưởng cao đóng góp trực tiếp gần 7% GDP và là một lợi thế phát triển đặc biệt của Việt Nam.	

Trước khi huấn luyện Skip-thought, chúng tôi huấn luyện bổ sung mô hình Word2vec. Word2vec sẽ ánh xạ từ vào không gian vector từ (chúng tôi chọn 300 thành phần là gợi ý trong [13]) trước khi đi vào mô hình Skip-thought. Quá trình thu thập dữ liệu và huấn luyện Word2vec được thực hiện tại phòng lab Công nghệ Multimedia và Thông minh - Khoa Công nghệ Thông tin - Học viện Kỹ thuật Quân sự với máy tính CPU Intel Core i7 2600, Ram 8Gb, GPU Nvidia Geforce GT 705.

Kho ngữ liệu huấn luyện Word2vec và Skip-thought ~2Gb được chúng tôi thu thập từ các bài báo, các sách từ internet, tiến hành tiền xử lý: loại bỏ số, địa chỉ email, các đường link, các dấu đặc biệt, địa chỉ IP,… Sau đó văn bản được tách thành các câu bằng các ký tự kết thúc. Các câu sẽ được tách thành các từ, ở đây chúng tôi sử

Bảng 5. So sánh kết quả các phương pháp

TT	Thuật toán	Pha	Accuracy	Precision	Recall	F1-Score
1	Ngưỡng cứng - Cosine (0.45)	Huấn luyện	93.8%	93%	95%	94%
		Kiểm thử	**96.01%**	95%	97%	96%
2	SVM	Huấn luyện	100%	100%	100%	100%
		Kiểm thử	95.33%	94%	96%	95%
3	MLP	Huấn luyện	100%	100%	100%	100%
		Kiểm thử	94.94%	96%	95%	95%
4	K-NN (n=10)	Huấn luyện	86.81%	100%	80%	89%
		Kiểm thử	83.45%	**98%**	76%	86%

Bảng 6. Kết quả 5 câu gần nghĩa nhất với: "Tách MobiFone, VNPT cần có chiến lược gì trước "vạch xuất phát" mới?"

Độ tương tự Cosine	Nội dung	Nhãn
85	Tách MobiFone, VNPT phải làm gì trước "vạch xuất phát" mới?	1
0.4	Có thể xem việc MobiFone - con gà đẻ trứng vàng tách ra - chính là một cơ hội lớn để VNPT phải quyết liệt trước vạch xuất phát mới.	0
0.36	"Nếu tách thì VNPT đương nhiên khó khăn mặc dù có cơ chế để VNPT vẫn chịu đựng nổi về mặt tài chính chứ không bị sốc quá lớn nhưng vẫn có những khó khăn trong 1 - 2 năm đầu", TS Trực nhận định.	0
0.33	MobiFone là công sức của VNPT, là đứa con cả trong nhà của VNPT đang làm ăn tốt, chiếm tới 50-60 % lợi nhuận của VNPT nên vạn bất đắc dĩ mới phải tách ra.	0
0.32	Vấn đề được quan tâm đặt ra hiện nay là những bước đi sắp tới của VNPT sẽ như thế nào sau khi MobiFone đã chính thức tách riêng từ ngày 1/7?	0

Bảng 7. Một số trường hợp đúng chương trình chưa truy vấn được

Nội dung	Nhãn
Giấy tờ liên quan đến việc gửi tiền ở ngân hàng do bà Phát giữ.	1
Bà Phát giữ những giấy tờ liên quan đến việc gửi tiền ở ngân hàng.	
Trong đơn ông Hà Xuân trình bày trước đây ông có nhờ bà Phát gửi giùm số tiền 90.000.	1
Mọi người thống nhất giao cho ông và con gái nuôi hợp pháp của bà Phát đồng đứng ra quản lý số tài sản đã được kiểm kê.	
Làm người phải giữ chữ tín.	1
Chữ tín phải được đặt lên hàng đầu.	
Giá USD tự do duy trì xu thế giảm mạnh còn 21.250 đồng được cho là nguyên nhân kéo giá vàng xuống.	1
Các ngân hàng cho biết một trong những nguyên nhân khiến giá USD giảm là do cung cầu trên thị trường khá dồi dào.	

dụng bộ tách câu ViTokenizer của nhóm tác giả Lê Hồng Phương.

Qua thử nghiệm, sử dụng Word2vec huấn luyện bổ sung sẽ cho mô hình hội tụ nhanh hơn. Ngoài ra, khi xây dựng từ điển cho Skip-thought, số lượng từ có thể lên đến hơn 400.000 từ. Sẽ có những từ có tần suất xuất hiện rất thấp, điều này sẽ làm tốn chi phí tính toán không cần thiết. Chính vì thế chúng tôi chỉ sử dụng lớp Embedding gồm 10.000 từ có tần suất xuất hiện cao nhất. Đối với những từ bị bỏ đi, chúng tôi thay bằng ký tự đặc biệt UNK = '?' để không làm mất bố cục câu. Chúng tôi gán thêm hai ký tự đặc biệt để đánh dấu bắt đầu câu và kết thúc câu. Sau pha huấn luyện, lớp Embedding 10.000 từ sẽ được thay thế bằng lớp Embedding gồm hơn 400.000 từ, điều

này không ảnh hưởng đến kết quả huấn luyện mà mở rộng khả năng xử lý các từ chưa được gặp trong huấn luyện.

Trong thử nghiệm của mình, nhóm tác giả chọn GRU cho cả Encoder và Decoder. Số chiều của trạng thái ẩn được đặt là 800. Tác giả Kiros đề xuất số chiều trạng thái ẩn là 1200 nhưng qua quá trình huấn luyện, chúng tôi thử với số chiều tăng dần là 400, 600, 800, 1200 thì giá trị của hàm mất mát không thay đổi nhiều nhưng mô hình thì chậm hơn đáng kể (Bảng 3). Với số chiều là 400 và 600 thì mô hình bị underfit, độ chính xác chỉ đạt ~50% sau quá trình huấn luyện. Do đó chúng tôi chọn số chiều trạng thái ẩn là 800, độ chính xác mô hình sau huấn luyện là 85%. Thay vì sử dụng GRU truyền thống chỉ xử lý một chiều từ đầu câu đến cuối câu, chúng tôi sử dụng GRU-2 chiều, điều này làm tăng độ chính xác của mô hình.

Pha huấn luyện sử dụng kỹ thuật teacher-forcing nhằm tăng tốc độ và độ chính xác của mô hình, tránh được việc tính đạo hàm lan truyền ngược.

Trong thực nghiệm, mô hình dừng lại sau 10 epochs, chi phí thời gian cho mỗi epochs là ~4 tiếng, giá trị hàm mất mát ~0.7.

Để đánh giá mô hình chúng tôi sử dụng bộ ngữ liệu do tác giả Ngô Xuân Bách công bố và sử dụng trong [11]. Bộ ngữ liệu gồm có 3000 cặp câu, được gán nhãn sẵn. Bộ ngữ liệu được xây dựng trên các tin mạng (dantri.com.vn, vnexpress.net, thanhnien.com.vn, .v.v.). Tác giả lấy ra hai câu từ hai tin có cùng chủ đề, sau đó xác định hai câu đó có tương đồng về nghĩa hay không. Việc xác định câu có tương đồng về nghĩa hay không do hai người thực hiện độc lập. Hệ số tin cậy Kappa đạt được là 0.9. Kết quả là 1500 được gán nhãn là tương đồng ngữ nghĩa (nhãn 1), 1500 gán nhãn là không tương đồng (nhãn 0).

Chúng tôi dùng Encoder của mô hình để tính 6.000 vector của 3.000 cặp trong câu bộ ngữ liệu Vnpara. Sau đó chia tập ngữ liệu thành 2 phần với tỷ lệ 70% dùng cho huấn luyện và 30% dùng cho kiểm thử. Chúng tôi xác định hai câu đồng nghĩa bằng cách thiết lập ngưỡng cứng cho độ đo tương tự cosine của hai vector v_1, v_2 là vector biểu diễn hai câu trong cùng một mẫu Vnpara. Đồng thời, theo thử nghiệm của Kiros, chúng tôi kết hợp hai vector v_1, v_2 bằng các nối $v_1 \odot v_2$ ($\odot$ là phép nhân từng thành phần) và $|v_1 - v_2|$, vector cuối cùng sẽ là đầu vào cho một số phương pháp học máy khác (2, 3, 4) được mô tả trong Bảng 5.

So với kết quả đạt được của Ngô Xuân Bách: Accuracy = 89.10%, F1-Score = 86.77%, kết quả chúng tôi vượt trội hơn hẳn nếu sử dụng ngưỡng 0.45 để phân lớp: Accuracy = 96.01% và F1-Score = 96%. Trong bài toán đặt ra, tìm kiếm câu đồng nghĩa trong văn bản là tìm kiếm không chính xác. Với mỗi cặp câu được gán nhãn 1 truy vấn, lấy câu thứ nhất để truy vấn n câu gần nghĩa nhất trong tập câu thứ hai, nếu trong tập n kết quả trả về được sắp xếp theo giá trị độ đo tương tự cosine có chứa câu thứ hai cùng mẫu thì truy vấn được coi là chính xác. Với cách đánh giá trên, kết quả đạt thể hiện ở Bảng 8.

Ví dụ một truy vấn trên VnPara: "*Tách MobiFone, VNPT cần có chiến lược gì trước "vạch xuất phát" mới?"* cho kết quả với 5 câu gần nhất được thể hiện trong Bảng 6.

Bảng 8. Độ chính xác truy vấn n câu gần nghĩa nhất

n câu gần nhất	Độ chính xác	Số trường hợp đúng không truy vấn được
5	97.65%	37
10	98.35%	26
15	98.79%	19

Một số trường hợp đúng mà chương trình chưa truy vấn được thể hiện trong Bảng 7.

4. Kết luận

Nghiên cứu và kết quả thử nghiệm cho thấy mô hình Skip-thought đã khắc phục được nhược điểm của các phương pháp cũ, đồng thời phù hợp với bài toán tìm kiếm câu đồng nghĩa trong văn bản tiếng Việt. Qua đánh giá mô hình với bộ ngữ

liệu Vnpara, kết quả thử nghiệm khi sử dụng Skip-thought đạt độ chính xác lên đến 96.01% vượt trội so với phương pháp của nhóm Ngô Xuân Bách (89.1%). Áp dụng cho bài toán tìm kiếm câu đồng nghĩa trong văn bản, kiểm thử trên bộ ngữ liệu Vnpara cho kết quả 96.9% với cách đánh giá trên mục 3.

Tài liệu tham khảo

[1] Wael H. Gomaa and Aly A. Fahmy, "A Survey of Text Similarity Approaches," *Int. J. Comput. Appl.*, vol. 68, no. 13, pp. 13-18, 2013.

[2] D. Kazakov and S. Dobnik, "Inductive learning of lexical semantics with typed unification grammars," no. May, 2014.

[3] W. Zhibiao and M. Palmer, "VERB SEMANTICS AND LEXICAL SELECTION," *32nd Annu. Meet. Assoc. Comput. Linguist.*, pp. 133-138, 1994.

[4] H. Liu and P. Wang, "Assessing sentence similarity using WordNet based word similarity," *J. Softw.*, vol. 8, no. 6, pp. 1451-1458, 2013.

[5] H. T. Nguyen, P. H. Duong, and V. T. Vo, "Vietnamese sentence similarity based on concepts," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 8838, 2014.

[6] M. C. Lee, J. W. Chang, and T. C. Hsieh, "A Grammar-Based Semantic Similarity Algorithm for Natural Language Sentences," vol. 2014, 2014.

[7] T. K. Landauer, P. W. Foltz, and D. Laham, "An Introduction to Latent Semantic Analysis," *Discourse Process.*, vol. 25, pp. 259-284, 1998.

[8] S. Simmons and Z. Estes, "Using latent semantic analysis to estimate similarity," *Proc. Cogn. Sci. Soc.*, pp. 2169-2173, 2006.

[9] K. Lund and C. Burgess, "Producing high-dimensional semantic spaces from lexical co-occurrence," *Behav. Res. Methods, Instruments, Comput.*, vol. 28, no. 2, pp. 203-208, 1996.

[10] C. Exposure and D. Ed, "From Word Embeddings To Document Distances," no. September, 2009.

[11] N. X. Bach, T. T. Oanh, N. T. Hai, and T. M. Phuong, "Paraphrase Identification in Vietnamese Documents," *Proc. - 2015 IEEE Int. Conf. Knowl. Syst. Eng. KSE 2015*, pp. 174-179, 2015.

[12] Y. Wu *et al.*, "Google's Neural Machine Translation System: Bridging the Gap between Human and Machine Translation," pp. 1-23, 2016.

[13] T. Mikolov, I. Sutskever, K. Chen, G. Corrado, and J. Dean, "Distributed Representations of Words and Phrases and their Compositionality," *IEEE Trans. neural networks*, vol. 14, no. 6, pp. 1569-72, Oct. 2013.

[14] R. Kiros *et al.*, "Skip-Thought Vectors," no. 786, pp. 1-11, 2015.

[15] S. Tang, H. Jin, C. Fang, Z. Wang, and V. R. de Sa, "Rethinking Skip-thought: A Neighborhood based Approach," 2017.

[16] Y. Bengio, R. Ducharme, V. Pascal, and J. Christian, "A Neural Probabilistic Language Model," *J. Mach. Learn. Res.*, vol. 3, pp. 1137-1155, 2003.