UBND TỈNH BÌNH DƯƠNG
TRƯỜNG ĐẠI HỌC THỦ DẦU MỘT

NGUYỄN TRUNG TÍN

XÂY DỰNG HỆ THỐNG HỎI ĐÁP TỰ ĐỘNG HỖ TRỢ CÔNG TÁC TƯ VẤN DỊCH VỤ HÀNH CHÍNH CÔNG TẠI SỞ THÔNG TIN VÀ TRUYỀN THÔNG TỈNH BÌNH DƯƠNG

LUẬN VĂN THẠC SĨ

CHUYÊN NGÀNH: HỆ THỐNG THÔNG TIN
MÃ SỐ: 8480104

BÌNH DƯƠNG - 2019

UBND TỈNH BÌNH DƯƠNG
TRƯỜNG ĐẠI HỌC THỦ DẦU MỘT

NGUYỄN TRUNG TÍN

XÂY DỰNG HỆ THỐNG HỎI ĐÁP TỰ ĐỘNG HỖ TRỢ CÔNG TÁC TƯ VẤN DỊCH VỤ HÀNH CHÍNH CÔNG TẠI SỞ THÔNG TIN VÀ TRUYỀN THÔNG TỈNH BÌNH DƯƠNG

LUẬN VĂN THẠC SĨ
CHUYÊN NGÀNH: HỆ THỐNG THÔNG TIN
MÃ SỐ: 8480104

NGƯỜI HƯỚNG DẪN KHOA HỌC:
TS. BÙI THANH HÙNG

BÌNH DƯƠNG - 2019

LỜI CAM ĐOAN

Tôi là Nguyễn Trung Tín, học viên lớp CH17HT01, ngành Hệ thống thông tin, trường Đại học Thủ Dầu Một. Tôi xin cam đoan luận văn **"Xây dựng hệ thống hỏi đáp tự động hỗ trợ công tác tư vấn dịch vụ hành chính công tại Sở Thông tin và Truyền thông tỉnh Bình Dương"** là do tôi nghiên cứu, tìm hiểu và phát triển dưới sự hướng dẫn của TS. Bùi Thanh Hùng, không phải sự sao chép từ các tài liệu, công trình nghiên cứu của người khác mà không ghi rõ trong tài liệu tham khảo. Tôi xin chịu trách nhiệm về lời cam đoan này.

Bình Dương, ngày 11 tháng 10 năm 2019

Tác giả

Nguyễn Trung Tín

LỜI CẢM ƠN

Để hoàn thành luận văn này, tôi xin gửi lời cảm ơn đến tất cả Quý thầy cô trường Đại học Thủ Dầu Một đã tận tình giảng dạy và truyền đạt cho tôi những kiến thức hữu ích trong suốt quá trình học tập tại trường. Tôi cũng xin chân thành cảm ơn Ban Giám đốc Sở Thông tin và Truyền thông tỉnh Bình Dương cùng Ban Giám đốc Trung tâm Công nghệ Thông tin và Truyền thông đã giúp đỡ, cung cấp nhiều thông tin quý báu và tạo điều kiện cho tôi trong quá trình thu thập dữ liệu, cảm ơn các anh chị em đồng nghiệp đã hỗ trợ cho tôi để tôi có thể thực hiện tốt luận văn của mình.

Hơn hết, tôi xin chân thành cảm ơn thầy hướng dẫn TS. Bùi Thanh Hùng, người đã tận tình truyền đạt, chỉ dạy cho tôi những kiến thức bổ ích về máy học và học tập sâu, cảm ơn thầy đã nhiệt tình hướng dẫn, chỉ bảo cho tôi trong suốt quá trình tôi nghiên cứu, xây dựng và hoàn thiện luận văn này.

Xin gửi lời cảm ơn sâu sắc tới gia đình, các anh chị em học viên lớp CH17HT đã luôn động viên, chia sẻ kinh nghiệm, cung cấp các tài liệu hữu ích cho tôi để tôi thực hiện tốt luận văn của mình.

MỤC LỤC

DANH MỤC THUẬT NGỮ VÀ CÁC TỪ VIẾT TẮT

Từ viết tắt	Từ tiếng Anh	Diễn giải
AI	Artificial Intelligence	Trí tuệ nhân tạo
BiLSTM	Bidirectional Long Short Term Memory	Bộ nhớ dài ngắn song song
Chatbot	Chatbot	Hệ thống trả lời tự động
FSA	Finite State Automaton	Máy tự động trạng thái hữu hạn
LSTM	Long Sort-Term Memory	Bộ nhớ dài ngắn
ML	Machine Learning	Học máy
NLP	Natural Languague Processing	Xử lý ngôn ngữ tự nhiên
NLU	Natural language understanding	Hiểu ngôn ngữ tự nhiên
QA	Question answering system	Hệ thống hỏi đáp
RNN	Recurrent Neural Network	Mạng nơ ron tái phát

DANH MỤC CÁC BẢNG

DANH MỤC HÌNH VẼ, ĐỒ THỊ

TÓM TẮT LUẬN VĂN

Hiện tại việc tiếp nhận, giải quyết và trả lời câu hỏi thắc mắc hoặc yêu cầu của người dùng như (Hệ thống hỏi đáp Q&A và giải quyết thắc mắc): của khách hàng trong hoạt động thương mại, của người dân trong thủ tục hành chính, của học sinh - sinh viên trong hoạt động đào tạo của các trường đại học - cao đẳng ... là rất lớn. Các hoạt động tiếp nhận câu hỏi và trả lời câu hỏi hiện nay đều là hoạt động mang tính thủ công mà chưa có công cụ nào trợ giúp. Việc tiếp nhận và xử lý còn chậm, thiếu chính xác và chưa công khai minh bạch. Các câu hỏi và yêu cầu của người dùng thì đi vào nhiều lĩnh vực và thuộc nhiều đối tượng trả lời khác nhau, việc lựa chọn đúng đối tượng trả lời gây khó khăn và hiểu nhầm cho người dùng dẫn đến các câu hỏi và yêu cầu thường không được trả lời thỏa đáng. Trong đề tài này, chúng tôi sẽ nghiên cứu, xây dựng một mô hình trả lời tự động tiếng Việt, dựa trên phương pháp phân loại câu hỏi bằng phương pháp học sâu từ đó sinh ra câu trả lời từ một chuỗi đầu vào tương ứng. Lợi thế của phương pháp này là đơn giản, nhanh và hiệu quả trong phạm vi dữ liệu thu thập ít. Chúng tôi áp dụng vào xây dựng hệ thống trả lời tự động ở Sở Thông tin và Truyền thông tỉnh Bình Dương. Mô hình đề xuất đã cho kết quả rất tính cực, hỗ trợ giải quyết các vấn đề cần tư vấn một cách nhanh chóng, hiệu quả.

Đề tài luận văn dựa trên những nghiên cứu trước đây để đề xuất nghiên cứu và phát triển một hệ thống trả lời tự động dựa trên hướng tiếp cận phân loại câu hỏi và trích xuất thông tin sử dụng mạng học sâu LSTM để sinh ra câu trả lời tự động từ một chuỗi đầu vào tương ứng. Mô hình phân loại câu hỏi theo hướng mạng bộ nhớ dài ngắn song song được áp dụng để huấn luyện trên bộ dữ liệu chuẩn và bộ dữ liệu tiếng Việt được thu thập, sau đó so sánh kết quả thực nghiệm trên bộ dữ liệu này.

Bộ dữ liệu thu thập sẽ phân tách thành hai bộ câu hỏi và câu trả lời tương ứng, sau đó tiến hành tách từ để tiến hành thiết lập biểu diễn các từ dưới dạng các vector và các bộ từ vựng để tiến hành huấn luyện và kết hợp với các phương pháp đánh giá để cho ra mô hình dự đoán đưa ra các câu trả lời tối ưu. Với bài toán dữ liệu nhỏ, tiếp cận xây dựng hệ thống trả lời tự động bằng phương pháp phân loại câu hỏi sẽ cho kết quả khả quan. Đó chính là lý do chúng tôi áp dụng phương pháp

này để xây dựng hệ thống trả lời tự động. Học máy là hướng tiếp cận chính áp dụng trong giải quyết các bài toán của xử lý ngôn ngữ tự nhiên trong đó có bài toán xây dựng hệ thống trả lời tự động. Chúng tôi áp dụng phương pháp học sâu BiLSTM vì phương pháp này đạt kết quả tốt, và cũng đánh giá so sánh với phương pháp học sâu LSTM từ đó tìm ra giải pháp tối ưu. Luận văn cũng đề xuất xây dựng một ứng dụng web hỗ trợ tư vấn trả lời tự động các câu hỏi của người dùng liên quan đến các dịch vụ hành chính công và các văn bản thường gặp của Sở Thông tin và Truyền thông tỉnh Bình Dương. Ứng dụng hỏi đáp tự động được triển khai thí điểm hỗ trợ công tác tư vấn, giải đáp thắc mắc các thủ tục hành chính tại Sở Thông tin và Truyền thông tỉnh Bình Dương.

CHƯƠNG 1
TỔNG QUAN VỀ LĨNH VỰC NGHIÊN CỨU

1.1. Lí do chọn đề tài

Trí tuệ nhân tạo (AI) và học máy (machine learning - ML) là thành phần chính trong Cuộc cách mạng công nghiệp 4.0 đang bùng nổ và phát triển mạnh mẽ. Xử lí ngôn ngữ tự nhiên Natural Language Processing (NLP) là một trong số những bài toán cơ bản của Trí tuệ nhân tạo với nhiều chủ đề như: Tìm kiếm, Trả lời tự động, Tóm tắt văn bản, Phân loại văn bản, Truy xuất thông tin, ...Chatbot (hay là một hệ thống trả lời tự động) được biết đến là một chương trình máy tính tương tác với người dùng bằng ngôn ngữ tự nhiên dưới một giao diện đơn giản, âm thanh hoặc dưới dạng tin nhắn. Chatbot được ứng dụng rất rộng rãi trong nhiều lĩnh vực như Tài chính ngân hàng, Kinh doanh – Sản xuất, Y tế, Giáo dục,... với mục đích làm trợ lý cá nhân, chăm sóc khách hàng, đặt chỗ, mua hàng, bán hàng tự động, hỗ trợ dạy và học, tư vấn dịch vụ công…

Hệ thống trả lời tự động (Chatbot) là một chương trình mô phỏng cuộc trò chuyện của một con người thông qua văn bản hoặc tương tác bằng giọng nói với máy. Người dùng có thể yêu cầu chatbot một câu hỏi hoặc thực hiện một lệnh và chatbot sẽ trả lời hoặc thực hiện các hành động được yêu cầu. Mức độ chuẩn xác và tự nhiên của câu trả lời phụ thuộc vào khả năng xử lý dữ liệu đầu vào cũng như độ phức tạp của thuật toán lựa chọn đầu ra của hệ thống.

Chatbot được sử dụng hỗ trợ việc trả lời các yêu cầu lặp đi lặp lại. Khi cuộc trò chuyện trở nên quá phức tạp đối với một chatbot, nó sẽ được chuyển đến một nhân viên dịch vụ. Các trợ lý ảo đang ngày càng được sử dụng rộng rãi để xử lý các tác vụ đơn giản, giải phóng tác nhân của con người. Điều này giúp tiết kiệm chi phí và cho phép các công ty cung cấp một dịch vụ tư vấn khách hàng liên tục ngay cả khi không có nhân viên tư vấn trực tiếp.

Với bài toán dữ liệu nhỏ, tiếp cận xây dựng hệ thống trả lời tự động bằng phương pháp phân loại câu hỏi sẽ cho kết quả khả quan. Đó chính là lý do chúng tôi áp dụng phương pháp này để xây dựng hệ thống trả lời tự động. Học máy là hướng tiếp cận chính áp dụng trong giải quyết các bài toán của xử lý ngôn ngữ tự nhiên trong đó có bài toán xây dựng hệ thống trả lời tự động. Chúng tôi áp dụng phương

pháp học sâu BiLSTM vì phương pháp này đạt kết quả tốt, và cũng đánh giá so sánh với phương pháp học sâu LSTM từ đó tìm ra giải pháp tối ưu. Ứng dụng hỏi đáp tự động được triển khai thí điểm hỗ trợ công tác tư vấn, giải đáp thắc mắc các thủ tục hành chính tại Sở Thông tin và Truyền thông tỉnh Bình Dương.

1.2. Mục tiêu nghiên cứu

Phân loại câu hỏi là pha đầu tiên trong kiến trúc chung của một hệ thống hỏi đáp, có nhiệm vụ tìm ra các thông tin cần thiết làm đầu vào cho quá trình xử lý của các pha sau (trích chọn tài liệu, trích xuất câu trả lời, …). Vì vậy phân loại câu hỏi có vai trò hết sức quan trọng, ảnh hưởng trực tiếp đến hoạt động của toàn bộ hệ thống. Nếu phân loại câu hỏi không tốt thì sẽ không thể tìm ra được câu trả lời. Chính vì lý do này mà chúng tôi chọn và nghiên cứu đề tài **"Xây dựng hệ thống hỏi đáp tự động hỗ trợ công tác tư vấn dịch vụ hành chính công tại Sở Thông tin và Truyền thông tỉnh Bình Dương"**.

Luận văn đặt ra mục tiêu nghiên cứu các mô hình có thể phát sinh văn bản, sử dụng các mạng học sâu Long Short Term Memory (Mạng nơ ron bộ nhớ dài ngắn (LSTM)) và mạng Bidirectional LSTM (mạng nơ ron bộ nhớ dài ngắn song song (BiLSTM)) để xử lý các phần khác nhau của câu hỏi, huấn luyện trên tập dữ liệu câu hỏi và trả lời về các thông tin liên quan đến các thủ tục hành chính tại Sở Thông tin và Truyền thông tỉnh Bình Dương. Từ đó xây dựng, cài đặt và vận hành một mô hình trả lời tự động với mục tiêu của đề tài là tiết kiệm được nhân lực và thời gian trong quá trình tiếp nhận, và giải quyết các yêu cầu của người dân, doanh nghiệp trên địa bàn tỉnh.

1.3. Đối tượng, phạm vi nghiên cứu

Nghiên cứu các Mô hình huấn luyện dựa trên nền tảng học sâu Long Short Term Memory để xây dựng hệ thông trả lời tự động.

Lĩnh vực nghiên cứu: xây dựng mô hình trả lời tự động các câu hỏi của người dân liên quan đến những thủ tục hành chính do Sở Thông tin và Truyền thông tỉnh Bình Dương phục trách thông qua một hệ thống câu hỏi và trả lời được xây dựng từ trước. Qua cơ chế huấn luyện từ các phương pháp của DeepLearning như: RNN, CNN, LSTM, BiLSTM sau đó tiến hành dự đoán để trả lời các câu hỏi của người dân.

1.4. Phương pháp nghiên cứu

Luận văn dựa trên phương pháp nghiên cứu lý thuyết và thực nghiệm, vận dụng các lý thuyết về xử lý ngôn ngữ tự nhiên, các nghiên cứu mới trong học máy và lĩnh vực xử lý ngôn ngữ tự nhiên để đề xuất mô hình thích hợp. Luận văn cũng sử dụng phương pháp so sánh, đánh giá để phân tích đánh giá mô hình đề xuất với các mô hình trước.

1.5. Ý nghĩa khoa học và thực tiễn

Ý nghĩa khoa học của luận văn: Luận văn tập trung phân tích dữ liệu thu thập được gồm các thông tin liên quan đến dịch vụ công như hỏi đáp về thủ tục hành chính do Sở Thông tin và Truyền thông tỉnh Bình Dương phụ trách từ đó xây dựng ứng dụng trực quan hóa. Phân tích các yếu tố ảnh hưởng, lựa chọn các phương pháp học sâu phù hợp với bộ dữ liệu có được để hệ thống trả lời tự động đạt được độ chính xác cao nhất cho các câu hỏi của người dùng.

Ý nghĩa thực tiễn: Chúng tôi đã xây dựng được ứng dụng thử nghiệm trên nền tảng Web để trực quan hóa kết quả, từ đó người dùng có thể đặt các câu hỏi liên quan về dịch vụ công và đánh giá ứng dụng của chúng tôi.

1.6. Bố cục luận văn

Luận văn được chia thành 5 chương với các nội dung như sau:

- ✓ Chương 1 – Tổng quan về lĩnh vực nghiên cứu

  Sơ lược tổng quan về vấn đề nghiên cứu trên phương diện tổng quan nhất, nêu ra mục tiêu, phương pháp nghiên cứu và bố cục luận văn.

- ✓ Chương 2 – Cơ sở lý thuyết và các nghiên cứu liên quan

  Giới thiệu tổng quan về xử lí ngôn ngữ tự nhiên, về Word2Vector; giới thiệu về mạng nơ ron nhân tạo, các mô hình mạng nơ ron cải tiến là cơ sở của mạng học sâu. Nghiên cứu các mô hình phát sinh văn bản trong hệ thống đối thoại, giới thiệu về mô hình phân loại câu hỏi và các vấn đề chung có thể gặp phải khi xây dựng mô hình đối thoại; Trình bày cơ bản về hệ thống trả lời tự động, cùng với tình hình nghiên cứu trong nước và ngoài nước.

- ✓ Chương 3 – Mô hình đề xuất:

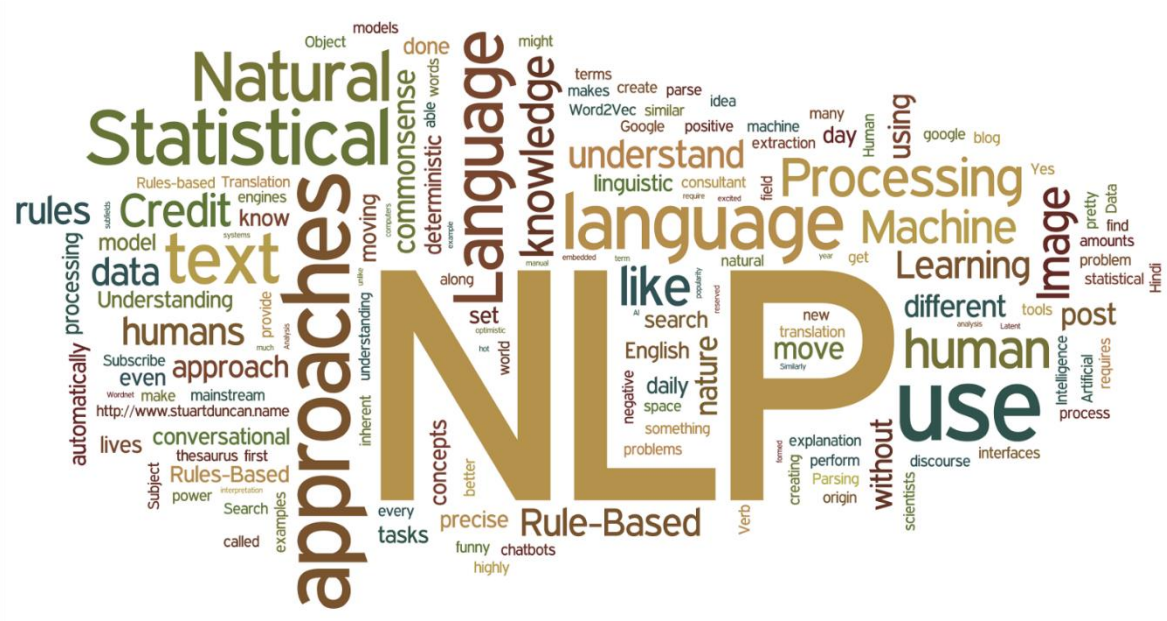
  Chương 3 trình bày tổng quan về mô hình đề xuất và đi sâu phân tích các đặc trưng của mô hình đề xuất.

- ✓ Chương 4 – Thực nghiệm trình bày chi tiết cụ thể các kết quả đạt được và phân tích, đánh giá, so sánh kết quả đạt được với các mô hình trước.
- ✓ Chương 5 – Kết luận và hướng phát triển.

CHƯƠNG 2
CƠ SỞ LÝ THUYẾT VÀ CÁC NGHIÊN CỨU LIÊN QUAN

2.1. Xử lý ngôn ngữ tự nhiên

Xử lý ngôn ngữ tự nhiên (gọi tắt NLP - Natural Language Processing) là các kĩ thuật, phương pháp nhằm hỗ trợ cho các hệ thống máy tính hiểu và xử lý, nhận dạng ngôn ngữ tự nhiên như tiếng Việt, tiếng Anh. Dịch máy, rút trích thông tin, truy hồi thông tin, hỏi đáp tự động,… là các hướng nghiên cứu chính của xử lý ngôn ngữ tự nhiên. Tổng quan các nghiên cứu về xử lý ngôn ngữ tự nhiên được trình bày trong hình 2.1.



Hình 2.1: Tổng quan các nghiên cứu về xử lý ngôn ngữ tự nhiên

2.1.1. Bài toán xác định ý định người dùng (intent detection)

Thông thường, người dùng thường truy cập hệ thống trả lời tự động (chatbot) với mong muốn hệ thống sẽ đưa ra những hành động trợ giúp mình về một vấn đề nào đó. Ví dụ, người dùng của hệ thống chatbot hỗ trợ đặt vé máy bay có thể đưa ra yêu cầu đặt vé của mình khi bắt đầu cuộc hội thoại. Để đưa ra hỗ trợ được chính xác, chatbot cần xác định được ý định (intent) đó của người dùng. Việc xác định ý định của người dùng sẽ quyết định hội thoại tiếp theo giữa người và chatbot sẽ diễn ra như thế nào. Vì thế, nếu xác định sai ý định người dùng, chatbot sẽ đưa ra những phản hồi không đúng, không hợp ngữ cảnh. Khi đó, người dùng có thể thấy chán ghét và không quay lại sử dụng hệ thống. Bài toán xác định ý định người dùng vì thế đóng vai trò rất quan trọng trong hệ thống chatbot.

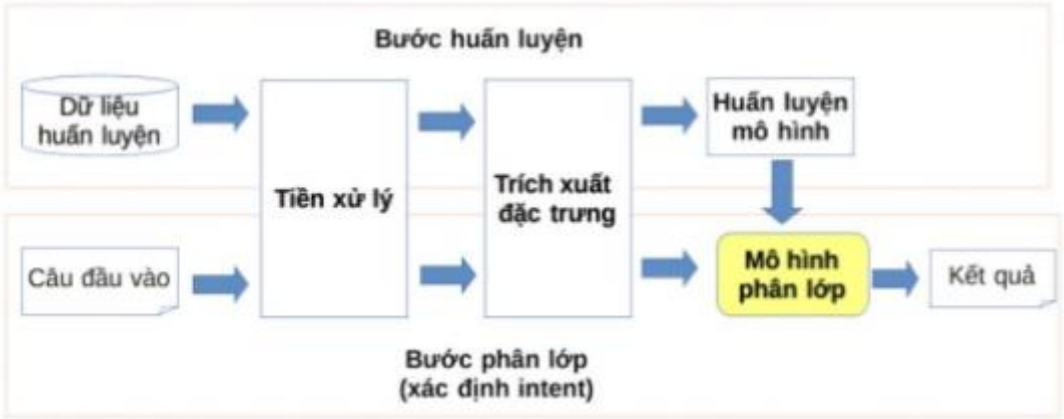
Đối với miền ứng dụng đóng, chúng ta có thể giới hạn rằng số lượng ý định của người dùng nằm trong một tập hữu hạn những intent đã được định nghĩa sẵn, có liên quan đến những nghiệp vụ doanh nghiệp mà chatbot có thể hỗ trợ. Với giới hạn này, bài toán xác định ý định người dùng có thể quy về bài toán phân lớp văn bản. Với đầu vào là một câu giao tiếp của người dùng, hệ thống phân lớp sẽ xác định intent tương ứng với câu đó trong tập các intent đã được định nghĩa.

Để xây dựng một mô hình phân lớp intent, chúng ta cần một tập dữ liệu huấn luyện bao gồm các cách diễn đạt khác nhau cho mỗi intent. Ví dụ, cùng một mục đích hỏi về thời tiết ở Hà Nội trong ngày hôm nay, người dùng có thể dùng những cách diễn đạt sau:

- Thời tiết hôm nay ở Hà Nội thế nào ad?
- Hà Nội hôm nay có mưa không vậy?
- Hà Nội hôm nay bao nhiêu độ vậy?
- Cho mình hỏi, ra ngoài đường hôm nay có phải mang áo mưa không?

Có thể nói, bước tạo dữ liệu huấn luyện cho bài toán phân lớp intent là một trong những công việc quan trọng nhất khi phát triển hệ thống chatbot và ảnh hưởng lớn tới chất lượng sản phẩm của hệ thống chatbot về sau. Công việc này cũng đòi hỏi thời gian, công sức khá lớn khi phát triển chatbot.

Khi đã có dữ liệu huấn luyện cho bài toán phân lớp intent, chúng ta sẽ mô hình bài toán thành bài toán phân lớp văn bản. Bài toán phân lớp văn bản (text categorization) là một bài toán kinh điển trong ngành NLP và khai phá văn bản (Text Mining). Kiến trúc của hệ thống phân lớp intent được minh hoạ trong Hình 2.2.



Hình 2.2: Những thành phần trong hệ phân lớp intent

Hệ thống phân lớp intent có một số thành phần cơ bản:

- Tiền xử lý dữ liệu
- Trích xuất đặc trưng
- Huấn luyện mô hình
- Phân lớp

Trong bước tiền xử lý dữ liệu, chúng ta sẽ thực hiện các thao tác "làm sạch" dữ liệu như: loại bỏ các thông tin dư thừa, chuẩn hoá dữ liệu như chuyển các từ viết sai chính tả thành đúng chính tả, chuẩn hoá các từ viết tắt,… Việc tiền xử lý dữ liệu có vai trò quan trọng trong hệ thống chatbot do đặc thù của ngôn ngữ chat, nói: viết tắt, sai chính tả, hay dùng "teencode".

Sau khi tiền xử lý dữ liệu và thu được dữ liệu đã được làm sạch, chúng ta sẽ trích xuất những đặc trưng từ dữ liệu này. Trong học máy, bước này được gọi là trích xuất đặc trưng (feature extraction hay feature engineering). Trong mô hình học máy truyền thống (trước khi mô hình học sâu được áp dụng rộng rãi), bước trích xuất đặc trưng ảnh hưởng lớn đến độ chính xác của mô hình phân lớp. Để trích xuất được những đặc trưng tốt, chúng ta cần phân tích dữ liệu khá tỉ mỉ và cần cả những tri thức chuyên gia trong từng miền ứng dụng cụ thể.

Bước huấn luyện mô hình nhận đầu vào là các đặc trưng đã được trích xuất và áp dụng các thuật toán học máy để học ra một mô hình phân lớp. Các mô hình phân lớp có thể là các luật phân lớp (nếu sử dụng decision tree) hoặc là các vector trọng số tương ứng với các đặc trưng được trích xuất (như trong các mô hình logistic regression, SVM, hay mạng Neural).

Sau khi có một mô hình phân lớp intent, chúng ta có thể sử dụng nó để phân lớp một câu hội thoại mới. Câu hội thoại này cũng đi qua các bước tiền xử lý và trích xuất đặc trưng, sau đó mô hình phân lớp sẽ xác định "điểm số" cho từng intent trong tập các intent và đưa ra intent có điểm cao nhất.

2.1.2. Bài toán trích xuất thông tin (IE - Information extraction)

Bài toán trích xuất thông tin là một trong những bài toán chính của xử lý ngôn ngữ tự nhiên. Với ví dụ là các câu hội thoại của người dùng, chúng ta cần trích xuất các thông tin cần thiết trong đó. Các thông tin cần trích xuất trong một câu hội thoại thường là các thực thể thuộc về một loại nào đó. Ví dụ, khi một khách hàng muốn đặt vé máy bay, hệ thống cần biết địa điểm xuất phát và địa điểm khách muốn

đến, ngày giờ khách hàng muốn bay,…Thành phần của hệ thống trích xuất thông tin của các hệ thống trả lời tự động thường hỗ trợ các loại thực thể như:

- Vị trí (Location)
- Thời gian (Datetime)
- Số (Number)
- Địa chỉ liên lạc (Contact)
- Khoảng cách (Distance)
- Khoảng thời gian (Duration)

Tôi	muốn	đặt	vé	máy bay	đi	Phú Quốc	sân bay	Nội Bài	vào	8	giờ	tối	ngày	mai
O	O	O	O	O	O	B-LOCATION	O	B-LOCATION	O	B-TIME	I-TIME	I-TIME	I-TIME	I-TIME

Hình 2.3: Gán nhãn từ theo mô hình B-I-O trong trích xuất thông tin

Đầu vào của một module trích xuất thông tin là một câu hội thoại. Module trích xuất thông tin cần xác định vị trí của các thực thể trong câu (vị trí bắt đầu và vị trí kết thúc của thực thể). Ví dụ sau minh hoạ một câu hội thoại và các thực thể được trích xuất từ đó.

• Câu hội thoại: Tôi muốn đặt vé máy bay đi Phú Quốc từ sân bay Nội Bài lúc 8 giờ tối ngày mai.

• Câu có các thực thể được xác định: Tôi muốn đặt vé máy bay đi [Phú Quốc]LOCATION từ sân bay [Nội Bài]LOCATION lúc [8 giờ tối ngày mai]TIME

Trong câu trên có 3 thực thể (nằm trong các dấu [ ]) với các loại thực thể tương ứng (được viết với font chữ nhỏ hơn ở dưới).

Cách tiếp cận phổ biến cho bài toán trích xuất thông tin là mô hình hoá bài toán thành bài toán gán nhãn chuỗi (sequence labeling). Đầu vào của bài toán gán nhãn chuỗi là một dãy các từ, và đầu ra là một dãy các nhãn tương ứng các các từ trong đầu vào. Chúng ta sẽ sử dụng các mô hình học máy để học một mô hình gán nhãn từ một tập dữ liệu đầu vào bao gồm các cặp ($x_1 \ldots x_n$, $y_1 \ldots y_n$), trong đó $x_1 \ldots x_n$ là dãy các từ, $y_1 \ldots y_n$ là dãy các nhãn. Độ dài của các dãy từ trong tập dữ liệu có thể khác nhau.

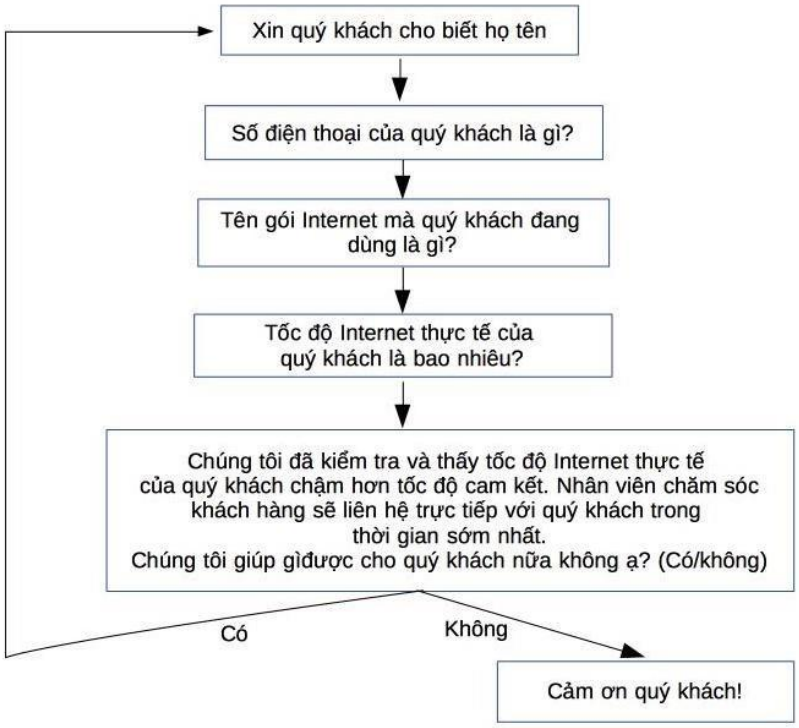
Trong bài toán trích xuất thông tin, tập nhãn cho các từ trong câu đầu vào thường được tạo ra theo mô hình BIO, với B là viết tắt của “Beginning”, I là viết tắt của “Inside”, và O là viết tắt của “Outside”. Khi biết vị trí từ bắt đầu của một thực thể và các từ nằm trong thực thể đó, chúng ta có thể xác định vị trí của thực thể

trong câu. Trong ví dụ ở trên, dãy các nhãn tương ứng với dãy của các từ trong câu hội thoại đầu vào được minh hoạ ở Hình 2.3.

2.1.3. Quản lý hội thoại

Trong các phiên trao đổi dài (long conversation) giữa người và chatbot, chatbot sẽ cần ghi nhớ những thông tin về ngữ cảnh (context) hay quản lý các trạng thái hội thoại (dialog state). Vấn đề quản lý hội thoại (dialoge management) khi đó là quan trọng để đảm bảo việc trao đổi giữa người và máy là thông suốt.

Chức năng của thành phần quản lý hội thoại là nhận đầu vào từ thành phần NLU, quản lý các trạng thái hội thoại (dialogue state), ngữ cảnh hội thoại (dialogue context), và truyền đầu ra cho thành phần sinh ngôn ngữ (Natural Language Generation, viết tắt là NLG). Ví dụ module quản lý dialogue trong một chatbot phục vụ đặt vé máy bay cần biết khi nào người dùng đã cung cấp đủ thông tin cho việc đặt vé để tạo một ticket tới hệ thống hoặc khi nào cần phải xác nhận lại thông tin do người dùng đưa vào. Hiện nay, các sản phẩm chatbot thường dùng mô hình máy trạng thái hữu hạn (Finite State Automata – FSA), mô hình Frame-based (Slot Filling), hoặc kết hợp hai mô hình này.



Hình 2.4: Minh hoạ quản lý hội thoại theo mô hình máy trạng thái hữu hạn FSA

FSA là mô hình quản lý hội thoại đơn giản nhất. Ví dụ, hãy tưởng tượng một hệ thống chăm sóc khách hàng của một công ty viễn thông, phục vụ cho những khách hàng than phiền về vấn đề mạng chậm. Nhiệm vụ của chatbot là hỏi tên khách hàng, số điện thoại, tên gói Internet khách hàng đang dùng, tốc độ Internet thực tế của khách hàng. Hình 2.4 minh hoạ một mô hình quản lý hội thoại cho chatbot chăm sóc khách hàng. Các trạng thái của FSA tương ứng với các câu hỏi mà dialogue manager hỏi người dùng. Các cung nối giữa các trạng thái tương ứng với các hành động của chatbot sẽ thực hiện. Các hành động này phụ thuộc phản hồi của người dùng cho các câu hỏi. Trong mô hình FSA, chatbot là phía định hướng người sử dụng trong cuộc hội thoại.

Ưu điểm của mô hình FSA là đơn giản và chatbot sẽ định trước dạng câu trả lời mong muốn từ phía người dùng. Tuy nhiên, mô hình FSA không thực sự phù hợp cho các hệ thống chatbot phức tạp hoặc khi người dùng đưa ra nhiều thông tin khác nhau trong cùng một câu hội thoại. Trong ví dụ chatbot ở trên, khi người dùng đồng thời cung cấp cả tên và số điện thoại, nếu chatbot tiếp tục hỏi số điện thoại, người dùng có thể cảm thấy khó chịu.

Mô hình Frame-based (hoặc tên khác là Form-based) có thể giải quyết vấn đề mà mô hình FSA gặp phải. Mô hình Frame-based dựa trên các frame định sẵn để định hướng cuộc hội thoại. Mỗi frame sẽ bao gồm các thông tin (slot) cần điền và các câu hỏi tương ứng mà dialogue manager hỏi người dùng. Mô hình này cho phép người dùng điền thông tin vào nhiều slot khác nhau trong frame. Hình vẽ 4 là một ví dụ về một frame cho chatbot ở trên.

Slot	**Câu hỏi**
Họ tên	Xin quý khách cho biết họ tên
Số điện thoại	Số điện thoại của quý khách là gì ạ?
Tên gói Internet	Gói Internet mà quý khách đang dùng là gì ạ?
Tốc độ Internet thực tế	Tốc độ vào Internet của quý khách hiện thời là bao nhiêu ạ?

Hình 2.5: Frame đối thoại thông tin khách hàng (tình huống mạng chậm)

Thành phần quản lý dialogue theo mô hình Frame-based sẽ đưa ra câu hỏi cho khách hàng, điền thông tin vào các slot dựa trên thông tin khách hàng cung cấp cho đến khi có đủ thông tin cần thiết. Khi người dùng trả lời nhiều câu hỏi cùng lúc,

hệ thống sẽ phải điền vào các slot tương ứng và ghi nhớ để không hỏi lại những câu hỏi đã có câu trả lời.

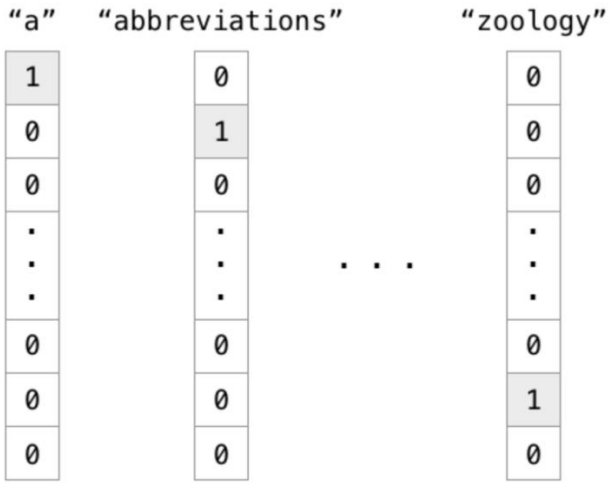
Trong các miền ứng dụng phức tạp, một cuộc hội thoại có thể có nhiều frame khác nhau. Vấn đề đặt ra cho người phát triển chatbot khi đó là làm sao để biết khi nào cần chuyển đổi giữa các frame. Cách tiếp cận thường dùng để quản lý việc chuyển điều khiển giữa các frame là định nghĩa các luật (production rule). Các luật này dựa trên một số các thành tố như câu hội thoại hoặc câu hỏi gần nhất mà người dùng đưa ra.

Các ứng dụng cơ bản của NLP: Dịch máy tự động (ví dụ Google translation), xử lý văn bản và ngôn ngữ, tìm kiếm thông tin, trích suất thông tin, tóm tắt văn bản, phân loại văn bản, data mining, web mining.

2.2. Biểu diễn từ bằng Vector - Word2vector

2.2.1. Biểu diễn One-hot-vector

Thông thường, cách truyền thống để biểu diễn một từ là dùng one-hot vector, khi đó độ lớn vector sẽ đúng bằng số lượng từ vựng có trong văn bản.

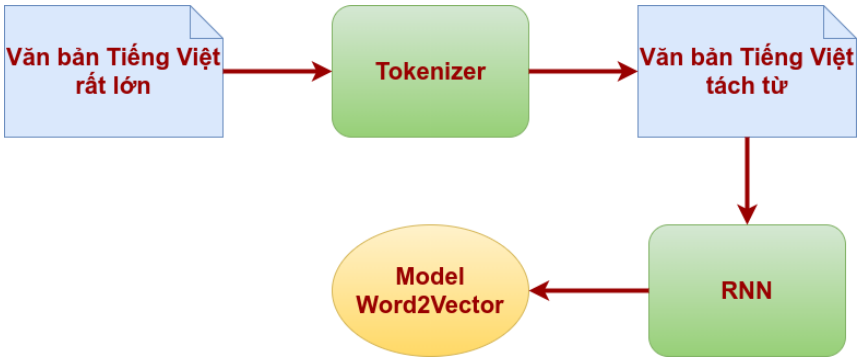


Hình 2.6: Biểu diễn one-hot-vector

Vấn đề ở đây là làm thế nào để thể hiện mối quan hệ giữa các từ và tính tương đồng giữa chúng trong văn bản. Do đó, Word2Vector là giải pháp để giải quyết vấn đề này.

Word2Vector là cách chúng ta biểu diễn 1 từ trong từ điển thành một vector trọng số, có số chiều cụ thể. Word2Vector được giới thiệu bởi một nhóm các nhà nghiên cứu tại Google vào năm 2013. Word2Vector sử dụng các kỹ thuật dựa trên

mạng thần kinh và học tập sâu để chuyển đổi các từ thành các vectơ tương ứng theo về mặt ngữ nghĩa gần nhau trong không gian N chiều.



Hình 2.7: Mô hình Word2vector

Hai trong số mô hình Word2vector được áp dụng để biểu diễn các từ là Skip-gram và Continuous Bag of Words (CBOW):

2.2.2. Túi từ liên tục - CBOW

Không giống như các mô hình language model thông thường chỉ có thể dự đoán từ tiếp theo dựa trên thông tin của các từ xuất hiện trước nó, mô hình Word Embedding không bị giới hạn như vậy. Với một mô hình lý tưởng, Word Embedding không chỉ có khả năng dự đoán tốt các từ tiếp theo trong một đoạn văn mà thậm chí có thể hiểu được nghĩa của từ, các từ đồng nghĩa hay trái nghĩa với nhau, hay thậm chí là nội dung tổng thể của cả đoạn văn nếu chúng ta biết cách kết hợp ý nghĩa của các từ cấu thành đoạn văn.

Trong paper của Tomas năm 2013, ông đã sử dụng các từ nằm ở phía trước và các từ ở phía sau của từ cần đoán (target word) để đào tạo mô hình Word Embedding. Nó được gọi với cái tên là Continuous Bag of Words đơn giản vì ông cho rằng, các từ có thể biển diễn một cách liên tục mà thứ tự của các từ trong một đoạn văn không phải là vấn đề.

Một ví dụ cụ thể như sau: Giả sử chúng ta có câu: *Con mèo ngồi trên sàn*

Câu trên được cắt nhỏ thành các từ có nghĩa trong từ điển Tiếng Việt. Bước này có thể dùng các tool tokenizer để làm. Kết quả chúng ta có như sau:

"Con" "mèo" "ngồi" "trên" "sàn"

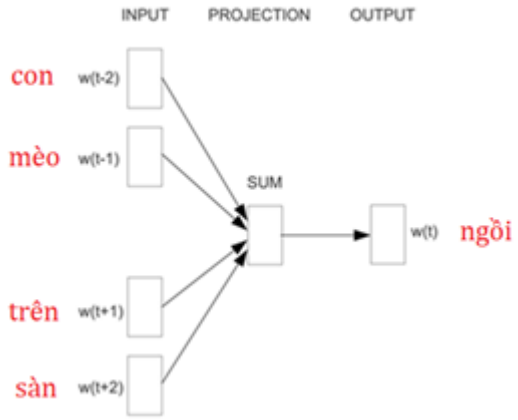
Trong ví dụ đầu tiên, giả sử dùng từ đầu vào (input word) trong mô hình là "*mèo*" và "trên" để dự đoán từ tiếp theo là "*ngồi*" với Window size là 1. Ở đây chú

ý là chúng ta đang cố dự đoán target word là "*ngồi*" dựa vào ngữ cảnh context input word là "*mèo*" và "*trên*".

Các cách biểu diễn trước đây như one-hot-vector sẽ chuyển toàn bộ các từ xuất hiện trong ví dụ trên thành dạng one-hot vector. Vì tổng số các từ có trong ví dụ của chúng ta là 5 (tạm gọi là *V*), vì thế nên one-hot vector có số chiều (dimensions) là 5. Chúng sẽ có dạng như sau:

```
# V = 5
vectors = {
                "con"       :[1, 0, 0, 0, 0]
                "mèo"       :[0, 1, 0, 0, 0]
                "ngồi"      :[0, 0, 1, 0, 0]
                "trên"      :[0, 0, 0, 1, 0]
                "sàn"       :[0, 0, 0, 0, 1]
}
```

One-hot vector là nguyên liệu cần để chúng ta xây dựng CBOW. Mô hình Continuous Bag of Words (CBOW) sử dụng ngữ cảnh để dự đoán mục tiêu. Một ngữ cảnh sẽ được xác định bằng một window size, tức là số từ đứng trước hoặc đứng sau từ cần xét. Ví dụ với widow size =1 thì từ ngữ cảnh "mèo, trên" mô hình CBOW dự đoán được từ đầu ra là "ngồi".



Hình 2.8: Mô hình Continuous Bag of Words

Input hay còn gọi là *context word* của chúng ta là 2 *one-hot vector* có size là V. *Hidden layer* trong kiến trúc CBOW chứa *N neutrals* và *output* lại quay trở lại là một vector có kích thước bằng V.

Diễn giải theo ví dụ ở phía trên của chúng ta theo một cách đơn giản như sau:

- **input layer:** đưa vào 2 vector [0, 1, 0, 0, 0] và [0, 0, 0, 1, 0] —đại diện cho từ "mèo" và "trên". Với một corpus thật lớn, chúng ta sẽ cho mạng CBOW học lần lượt từng từ trong corpus với context tương ứng như minh họa trên. Khi hiện

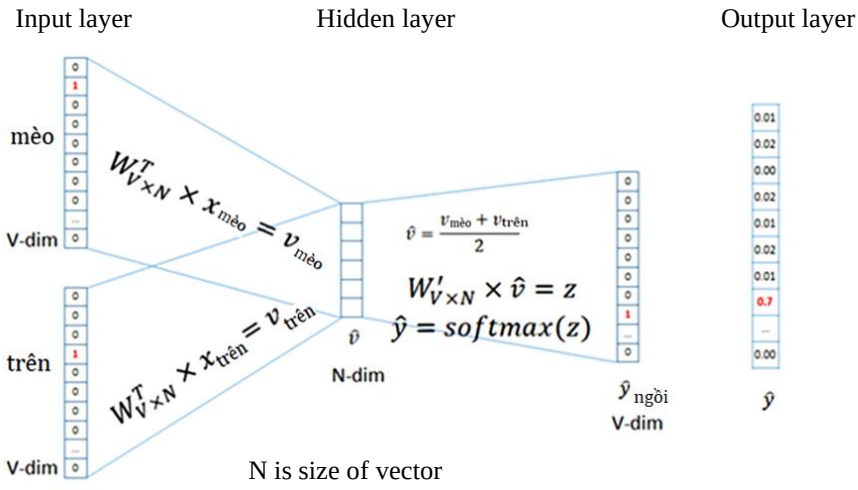
thực, 2 vector one-hot của 2 từ trong context sẽ được cộng lại thành một vector input duy nhất (có 2 giá trị 1) và đưa vào hệ thống như một input duy nhất.

- **hidden layer:** từ mà được truyền vào ở input layer sẽ được embed vào một không gian N chiều bằng cách nhân input vector X với ma trận trọng số W^T_{VxN}.
- **output layer:** kết quả của hidden layer sẽ được map với đầu ra chính là trọng số của từng từ có trong vocab V của chúng ta, trọng số càng cao thì có nghĩa là xác suất từ đó là từ tiếp theo (positive prediction) càng cao. Lưu ý: do trọng số ở output layer có biên độ không cố định nên thường thì người ta dùng softmax ở đoạn này nhằm convert trọng số của output layer thành xác suất probability. Hình 2.7 mô tả chi tiết mô hình CBOW cho ví dụ trên.

Nếu chúng ta chỉ quan tâm đến input và output thôi thì có thể hiểu đơn giản rằng, chúng ta xây dựng một mô hình sao cho khi input vào một từ (word) thì mô hình sẽ đưa ra tập hợp xác suất cho tất cả các từ có trong từ điển (vocab). Chúng ta kỳ vọng từ mà có xác suất cao thì tỉ lệ cao là từ đó sẽ là từ tiếp theo (target word) của từ mà mình đã input (context word). Để máy tính biết được mục tiêu để tối ưu mô hình của mình theo đúng như kỳ vọng trên, theo như bài báo của Mikolov, mục tiêu của CBOW là tối ưu trung bình xác suất [18]:

$$J_\theta = \frac{1}{T}\sum_{t=1}^{T} logp\ (w_t|w_{t-n}, \dots, w_{t-1}, w_{t+1}, \dots, w_{t+n}) \qquad (2.1)$$

Với $w_1, w_2, w_3, \dots, w_T$ là một chuỗi các từ huấn luyện

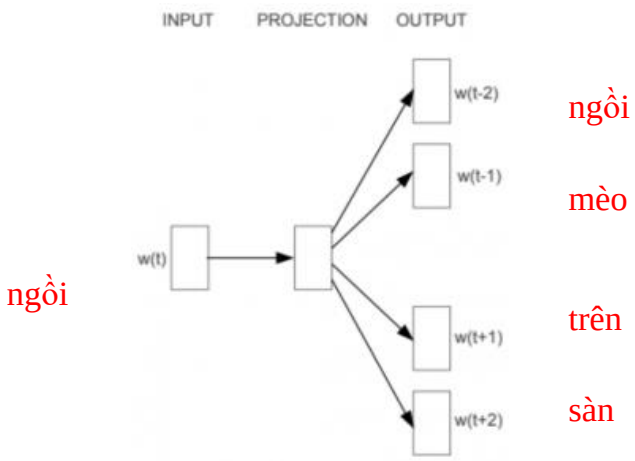


Hình 2.9: Mô hình CBOW chi tiết

Tuy nhiên, đối với tập dữ liệu lớn thì mô hình huấn luyện Word2vector phương pháp Skip gram cho kết quả tốt hơn.

2.2.3. Skip gram

Ngược lại với mô hình CBOW ở trên, trong mô hình Skip gram, đầu vào (input) là từ cần tìm mối quan hệ, đầu ra (output) là các từ có quan hệ gần nhất với từ được đưa ở đầu vào. Lấy ví dụ như trên, với input đầu vào là one-hot vector của từ "ngồi" và context ngữ cảnh với window size =1. Sau khi huấn luyện, chúng ta sẽ có được 1 ma trận W và 2 ma trận chuyển vị W'. Ma trận W sẽ được dùng để tạo ra các embedding vector từ một one-hot vectors.



Hình 2.10: Mô hình Skip gram trong Word2vec

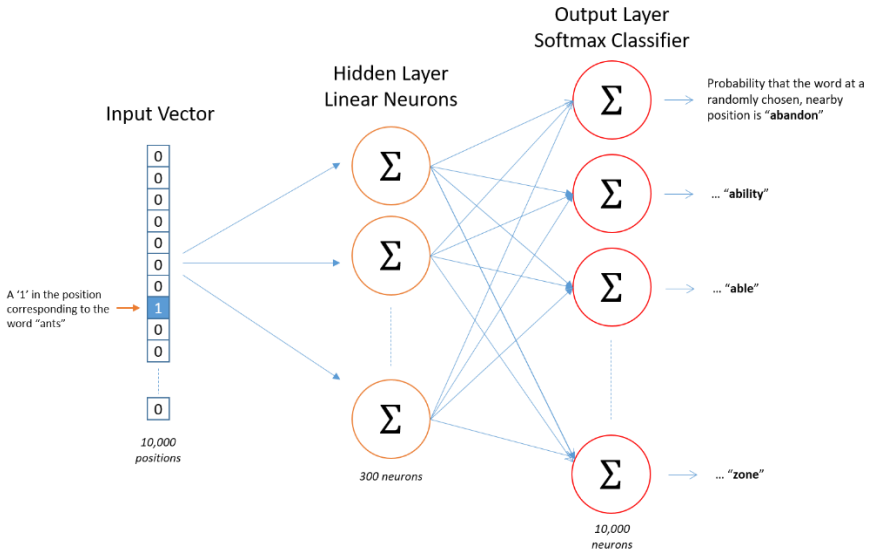
Mục tiêu đào tạo của mô hình Skip-gram là từ đầu vào input để dự đoán các từ xung quanh trong câu hoặc tài liệu. Đơn giản hơn, đưa ra một chuỗi các từ huấn luyện $w_1, w_2, w_3, \ldots, w_T$, mục tiêu của mô hình Skip-gram là tối đa hóa mức trung bình xác suất: [1]

$$J_\theta = \frac{1}{T}\sum_{t=1}^{T} \quad \sum_{-c\leq=j\leq c, j\neq 0}^{n} logp\left(w_{t+j}|w_t\right) \qquad (2.2)$$

Với c là kích cỡ ngữ cảnh huấn luyện, như ví dụ trên thì c=1

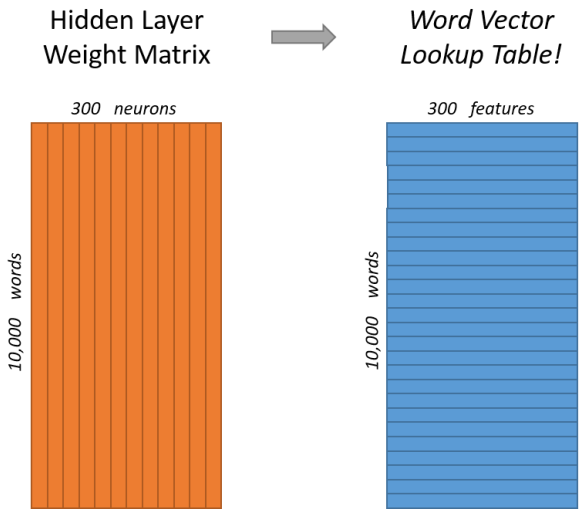
Chi tiết cách thực hiện

Trước hết, chúng ta đưa từ vào mạng neural một lớp ẩn. Để có thể huấn luyện được, từ được vector hóa để cho vào mạng chúng ta có thể xây dựng kho từ điển từ tập dữ liệu văn bản sau đó sử dụng one-hot-vector để diễn tả từng từ trong kho từ điển. Giả sử, nếu có từ điển gồm 10.000 từ riêng biệt, vector one-hot sẽ gồm 10.000 thành phần đại diện cho mỗi từ trong từ điển. Vector one-hot có dạng bao gồm toàn bộ giá trị bằng 0, chỉ có chỉ số tương ứng với vị trí của từ trong từ điển có giá trị bằng 1. Ví dụ từ "ants" sẽ biểu diễn bằng vector 10.000 phần tử gồm toàn số 0, duy nhất số 1 tại vị trí tương ứng với từ "ants" trong từ điển.



Hình 2.11: Mô hình mạng nơ ron 1 lớp ẩn của Word2vec

Lớp ẩn giả sử gồm 300 neuron, thường không sử dụng hàm activation, nhưng đầu ra thì sử dụng hàm softmax. Đầu ra sẽ là vector cũng là một vector có độ lớn 10.000 và giá trị tương ứng với mỗi vị trí là xác suất xuất hiện gần từ đã chọn của từ gần vị trí đó. Kích thước 300 neuron ở lớp ẩn là một hyperparameter của mô hình, nó được gọi là số chiều hay số đặc trưng của Word2vec. Con số 300 được Google sử dụng trong mô hình huấn luyện từ tập ngữ liệu Google News [2]. Giá trị hyperparameter có thể được thay đổi sao cho phù hợp với mô hình, dữ liệu của người nghiên cứu.



Hình 2.12: Ma trận trọng số của lớp ẩn của mô hình Word2vec

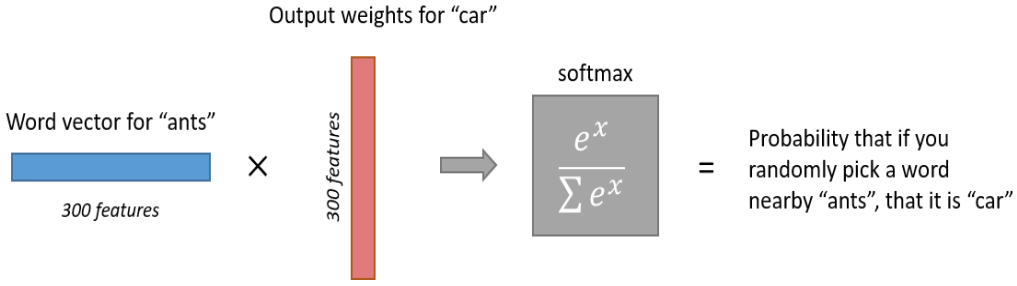
Mục đích cuối cùng của việc huấn luyện trên toàn tập ngữ liệu là tìm ra ma trận trọng số tại lớp ẩn. Nhận thấy đầu vào của mô hình là 1 từ được biểu diễn dưới dạng one-hot vector tức là một vector có các giá trị toàn bằng 0, chỉ có một vị trí

bằng 1 tương ứng với vị trí của từ đầu vào theo thứ tự từ điển. Việc nhân vector one-hot đầu vào với ma trận trọng số bản chất là việc tìm kiếm trên ma trận trọng số một vector đặc trưng có chiều dài bằng số chiều bằng số chiều của ma trận trọng số.

$$[0 \quad 0 \quad 0 \quad 1 \quad 0] \begin{bmatrix} 17 & 24 & 1 \\ 23 & 5 & 7 \\ 4 & 6 & 13 \\ 10 & 12 & 19 \\ 11 & 18 & 25 \end{bmatrix} = [10 \quad 12 \quad 19]$$

Hình 2.13: Lớp ẩn của mô hình hoạt động như một bảng tra cứu

Đầu ra của mô hình Word2vector là một bộ phân loại sử dụng hàm softmax để tính xác suất. Ưu điểm của hàm softmax là luôn tạo giá trị xác suất dương và tổng tất cả các xác suất thành phần là bằng 1. Giả sử tính mối tương quan giữa từ "ants" và từ "car", hai từ này sẽ được vector hóa dựa vào ma trận trọng số của lớp ẩn đã huấn luyện. Đầu ra qua hàm softmax sẽ có ý nghĩa là xác suất từ "car" xuất hiện gần từ được chọn "ants".



Hình 2.14: Mối tương quan giữa từ "ants" và từ "car"

Mục đích của việc biểu diễn từ thành các vector là để tạo ra các đầu vào cho các mô hình học máy xử lí ngôn ngữ tự nhiên, việc số hóa từ thành vector có nhiều cách thực hiện nhưng phương pháp tối ưu hơn cả là Word2vec, Fasttext và GloVe do nó thể hiện được mối tương quan của các từ với nhau trong không gian vector. Có nhiều mô hình học máy để xử lí ngôn ngữ tự nhiên và học sâu Deep learning hiện là hướng tiếp cận tối ưu nhất đặc biệt với dữ liệu huấn luyện lớn. Luận văn của tôi nghiên cứu sử dụng Word2vector được huấn luyện sẵn dùng mô hình Skipgram như trên.

2.3. Học sâu - Deep Learning

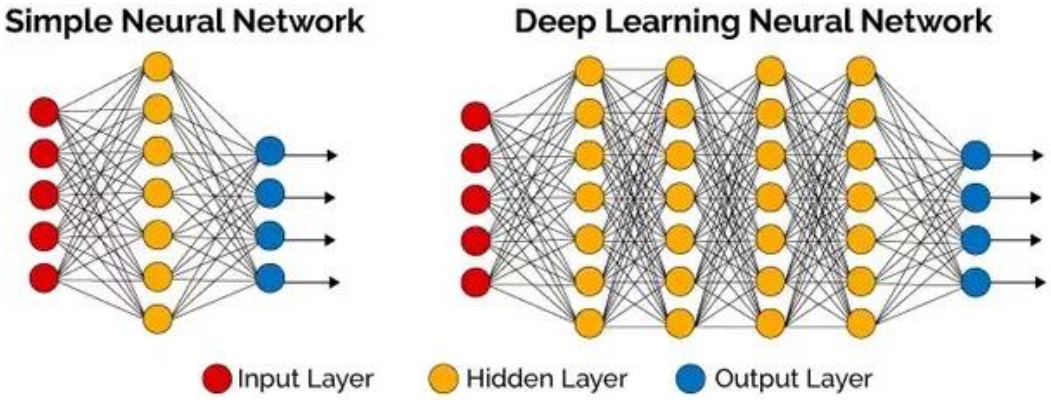
Học máy (Machine Learning) là một lĩnh vực của trí tuệ nhân tạo (Artificial Intelligence - AI). Các thuật toán học máy cho phép máy tính đào tạo đầu vào dữ

liệu và sử dụng phân tích thống kê để đưa ra các giá trị nằm trong một phạm vi cụ thể.

Ngày nay, những người sử dụng công nghệ đều được hưởng lợi từ việc học máy. Công nghệ nhận diện khuôn mặt giúp người dùng gắn thẻ và chia sẻ ảnh của bạn bè. Công nghệ nhận dạng ký tự quang học (OCR) chuyển đổi hình ảnh văn bản sang dạng di chuyển.

Khi mà khả năng tính toán của máy tính được nâng lên một tầm cao mới cùng với lượng dữ liệu khổng lồ được thu thập, Machine Learning đã tiến thêm một bước dài và Deep Learning (DL) một lĩnh vực mới được ra đời.

Deep Learning được xây dựng từ mạng nơ ron sinh học và bao gồm nhiều lớp trong mạng nơ ron nhân tạo được tạo thành từ phần cứng và GPU. Deep Learning sử dụng một tầng các lớp đơn vị xử lý phi tuyến để trích xuất hoặc chuyển đổi các tính năng (hoặc biểu diễn) của dữ liệu. Đầu ra của một lớp phục vụ như là đầu vào của lớp kế tiếp. Deep learning tập trung giải quyết các vấn đề liên quan đến mạng thần kinh nhân tạo nhằm nâng cấp các công nghệ như nhận diện giọng nói, dịch tự động (machine translation), xử lý ngôn ngữ tự nhiên…



Hình 2.15: Mô hình Deep Learning[1]

Trong số các thuật toán học máy hiện đang được sử dụng và phát triển, học sâu thu hút được nhiều nghiên cứu nhất và có thể đánh bại con người trong một số nhiệm vụ nhận thức. Do những đặc tính nổi bật và kết quả tối ưu, học tập sâu đã trở thành phương pháp tiếp cận được nghiên cứu và ứng dụng trong giải quyết nhiều bài toán thuộc lĩnh vực trí tuệ nhân tạo.

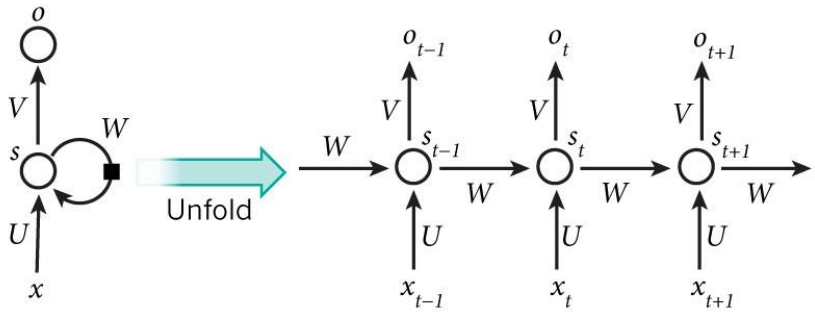
[1] *Nguồn từ researchgate.net*

2.3.1. Mạng nơ ron hồi quy RNN (Recurrent Neural Network)

Mạng nơ ron hồi quy RNN (Recurrent Neural Network) được giới thiệu bởi John Hopfield năm 1982 [3], là một trong những mô hình học sâu - Deep learning**.** Recurrent có nghĩa là thực hiện lặp lại cùng một tác vụ cho mỗi thành phần trong chuỗi. Trong đó, kết quả đầu ra tại thời điểm hiện tại phụ thuộc vào kết quả tính toán của các thành phần ở những thời điểm trước đó.

RNN là một mô hình có trí nhớ **(**memory**)**, có khả năng nhớ được thông tin đã tính toán trước đó. Không như các mô hình Neural Network truyền thống trước đó là thông tin đầu vào **(**input**)** hoàn toàn độc lập với thông tin đầu ra **(**output**).**

Hầu hết RNN được thiết kế như là một chuỗi các module được lặp đi lặp lại, các môdun này thường có cấu trúc đơn giản chỉ có một lớp mạng *tanh*. Huấn luyện RNN tương tự như huấn luyện ANN truyền thống. Giá trị tại mỗi output không chỉ phụ thuộc vào kết quả tính toán của bước hiện tại mà còn phụ thuộc vào kết quả tính toán của các bước trước đó.



Hình 2.16: Quá trình xử lý thông tin trong mạng RNN

RNN có khả năng biểu diễn mối quan hệ phụ thuộc giữa các thành phần trong chuỗi (nếu chuỗi đầu vào có 6 từ thì RNN sẽ dàn ra thành 6 layer, mỗi layer ứng với mỗi từ, chỉ số mỗi từ được đánh từ 0 đến 5. Trong Hình 2.8 ở trên, x_t là input tại thời điểm thứ t, s_t là hidden state(memory) tại thời điểm thứ t, được tính dựa trên các hidden state trước đó kết hợp với input của thời điểm hiện tại với công thức:

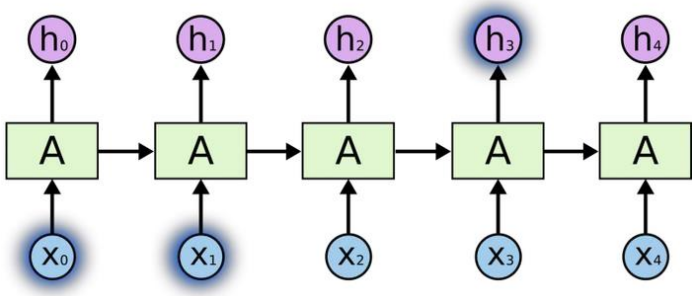
$$S_t = \tanh\left(U_{x_t} + W_{s_{t-1}}\right) \qquad (2.3)$$

S_{t-1} là hidden state được khởi tạo là 1 vector 0. O_t là output tại thời điểm thứ t, là một vector chứa xác xuất của toàn bộ các từ trong từ điển.

$$O_t = \text{softmax}(V_{s_t}) \qquad (2.4)$$

Không như ANN truyền thống, tại mỗi layer cần phải sử dụng một tham số khác, RNNs chỉ sử dụng một bộ parameters(U, V,W) cho toàn bộ các bước.

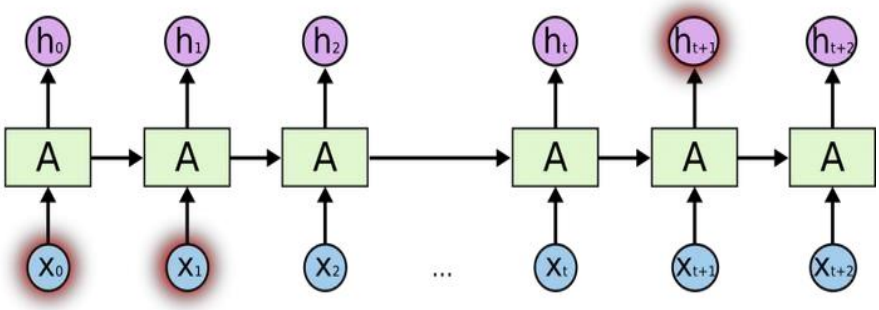
Ý tưởng ban đầu của RNN là kết nối những thông tin trước đó nhằm hỗ trợ cho các xử lý hiện tại. Nhưng đôi khi, chỉ cần dựa vào một số thông tin gần nhất để thực hiện tác vụ hiện tại. Ví dụ, chúng ta dự đoán từ cuối cùng trong câu "chuồn_chuồn bay thấp thì mưa", thì chúng ta không cần truy tìm quá nhiều từ trước đó, ta có thể đoán ngay từ tiếp theo sẽ là "mưa". Trong trường hợp này, khoảng cách tới thông tin liên quan được rút ngắn lại, mạng RNN có thể học và sử dụng các thông tin quá khứ.



Hình 2.17: RNN phụ thuộc short-term

Trường hợp có nhiều thông tin hơn trong một câu, nghĩa là phụ thuộc vào ngữ cảnh. Ví dụ nhưng khi dự đoán từ cuối cùng trong đoạn văn bản **"Tôi sinh ra và lớn lên ở Việt_Nam … Tôi có_thể nói thuần_thục Tiếng_Việt."** Từ thông tin gần nhất cho thấy rằng từ tiếp theo là tên một ngôn ngữ, nhưng khi chúng ta muốn biết cụ thể ngôn ngữ nào, thì cần quay về quá khứ xa hơn, để tìm được ngữ cảnh **Việt_Nam**. Và như vậy, RNN có thể phải tìm những thông tin có liên quan và số lượng các điểm đó trở nên rất lớn.

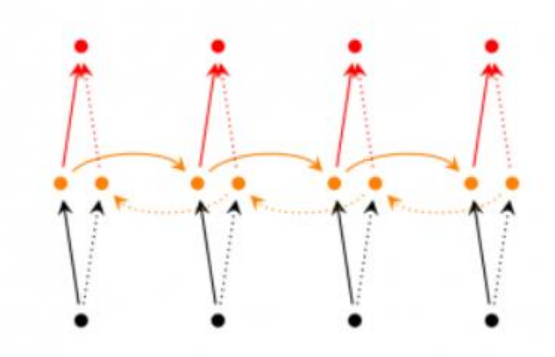
Không được như mong đợi, RNN không thể học để kết nối các thông tin lại với nhau.



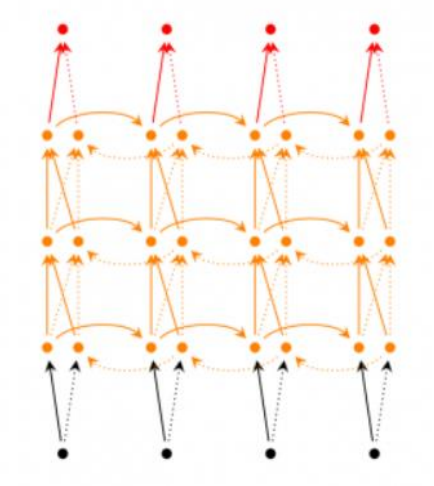
Hình 2.18: RNN phụ thuộc long-term

Về lý thuyết, RNN có thể nhớ được thông tin của chuỗi có chiều dài bất kì, nhưng trong thực tế mô hình này chỉ nhớ được thông tin ở vài bước trước đó[4].

RNN có các phiên bản mở rộng như: Bidirectional RNN **(**RNN hai chiều**),** Deep (Bidirectional) RNN**,** Long short-term memory networks (LSTM)[4][5] [6][7]**.**



Hình 2.19: Bidirectional RNN



Hình 2.20: Deep (Bidirectional) RNN

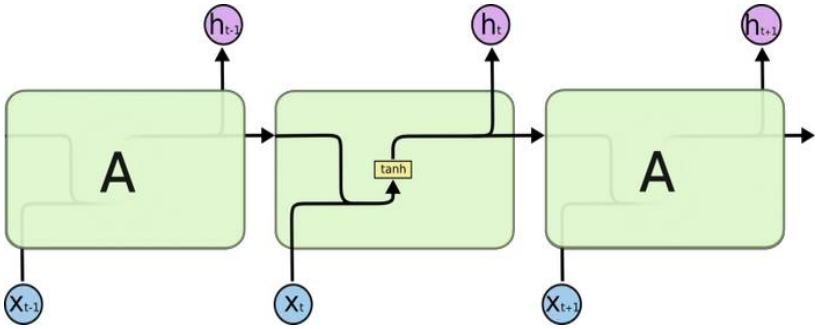
2.3.2. Bộ nhớ dài ngắn LSTM (Long-short term memory)

Là một dạng đặc biệt của mạng Nơron hồi quy, một kĩ thuật dựa trên Gradient. Trong quá trình hoạt động nó cho phép cắt bỏ những Gradient dư thừa. Trong quá trình học LSTM có thể thu hẹp thời gian trễ dư thừa của các bước thực hiện thông qua tập hằng số lỗi (theo Hochreiter & Schmidhuber- 1997 [4]).

LSTM có các thành phần cơ bản sau:

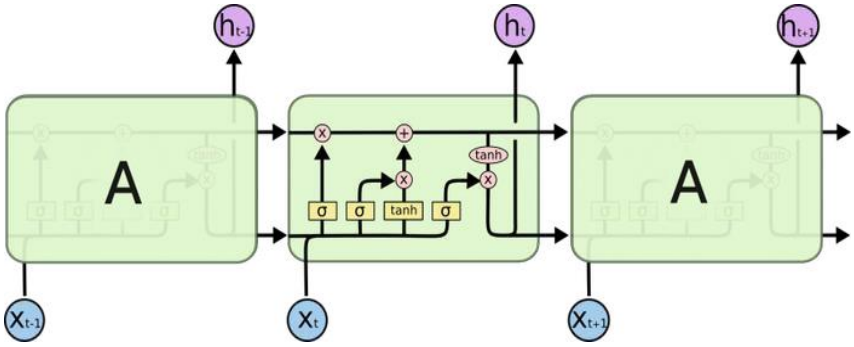
+ Tế bào trạng thái (cell state)
+ Cổng (gates)
+ Sigmoid
+ Tanh

LSTM được thiết kế nhằm loại bỏ vấn đề phụ thuộc quá dài. Ta quan sát lại mô hình RNN bên dưới, các layer đều mắc nối với nhau thành các module neural network. Trong RNN chuẩn, module repeating này có cấu trúc rất đơn giản chỉ gồm một lớp đơn giản **tanh layer**.



Hình 2.21: Các module lặp của mạng RNN chứa một layer

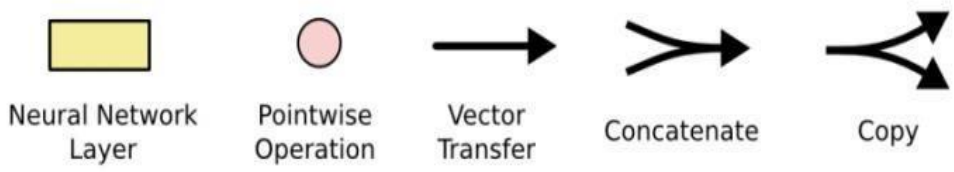
Về kiến trúc mạng LSTM**:** giống như RNN, nó là một chuỗi các module được lặp đi lặp lại. Tuy nhiên, xét về cấu trúc thì LSTM có 4 tầng mạng nơ ron tương tác với nhau, gọi đó là các tầng ẩn (hiden layer). Một số biến thể của LSTM được thực hiện dựa trên việc thay đổi vị trí kết nối giữa các tầng và cổng.



Hình 2.22: Các module lặp của mạng LSTM chứa bốn layer

Trong đó, các ký hiệu sử dụng trong mạng LSTM được giải nghĩa như hình 2.15 sau đây:

- Hình chữ nhật là các lớp ẩn của mạng nơ ron
- Hình tròn biểu diễn toán tử *Pointwise*
- Đường kẻ gộp lại với nhau biểu thị phép nối các toán hạng
- Và đường rẽ nhánh biểu thị cho sự sao chép từ vị trí này sang vị trí khác



Hình 2.23: Các kí hiệu sử dụng trong mạng LSTM

Cho một chuỗi các vectơ (x1, x2, …,xn), σ là hàm sigmoid logistic, trạng thái ẩn h_t, của LSTM tại thời điểm t được tính như sau:

$$h_t = o_t * tanh\,(c_t) \quad (2.5)$$

$$o_t = tanh\,(Wx_0x_t + Wh_0h_{t-1} + Wc_0c_t + b_o) \quad (2.6)$$

$$c_t = f_t * c_{t-1} + i_t * tanh(Wx_cx_t + Wh_ch_{t-1} + b_c) \quad (2.7)$$

$$f_t = \sigma\,(Wx_fx_t + Wh_fh_{t-1} + Wc_fc_{t-1} + b_f) \quad (2.8)$$

$$i_t = \sigma\,(Wx_ix_t + Wh_ih_{t-1} + Wc_ic_{t-1} + b_i) \quad (2.9)$$

Trong đó:

+ tanh là tang hyperbolic

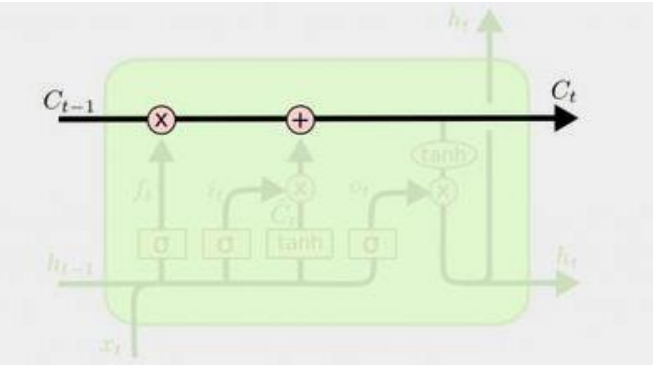
+ σ là hàm signmoid hay còn gọi là hàm logic chuẩn là nghiệm của phương trình sai phân phi tuyến bậc 1:

$$\frac{dP}{dt} = P(1-P)\ trong\ \text{đó}\ P(t) = \frac{1}{1+e^t} \quad (2.10)$$

Phân tích mô hình *LSTM*:

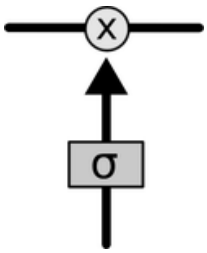
+ LSTM được thiết kế là một bảng mạch số, gồm các mạch logic và các phép toán logic. Thông tin di chuyển trong mạch sẽ được lưu trữ, lan truyền theo cách mà chúng ta thiết kế mạch.

+ Mấu chốt của *LSTM* là *cell state* (tế bào trạng thái), là đường kẻ ngang chạy dọc ở trên *top diagram*. *Nó* giống như băng chuyền, chạy xuyên thẳng toàn bộ mắc xích, chỉ một vài tương tác nhỏ tuyến tính (*minor linear interaction*) được thực hiện, giúp thông tin trong quá trình lan truyền ít bị thay đổi.



Hình 2.24: Tế bào trạng thái LSTM giống như một băng truyền

+ Trong LSTM cấu trúc cổng (*gate*) có khả năng thêm hoặc bớt thông tin vào *cell state*. Các cổng này được tạo bởi hàm *sigmoid* và một toán tử nhân (*pointwise*).

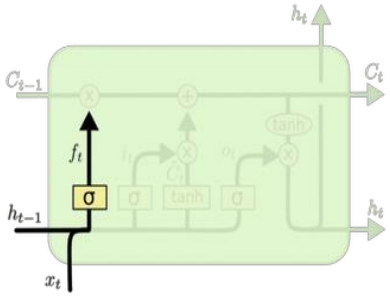


Hình 2.25: Cổng trạng thái LSTM

+ Hàm kích hoạt *Sigmoid* mô tả độ lớn thông tin được phép truyền qua tại mỗi lớp mạng, có giá trị từ 0 – 1. Thu giá trị 0 có nghĩa là "không cho bất kỳ cái gì đi qua", ngược lại nếu thu được giá trị là 1 có nghĩa là "cho phép mọi thứ đi qua". Một *LSTM* có ba cổng như vậy để bảo vệ và điều khiển *cell state*.

Quá trình hoạt động của LSTM được thông qua các bước cơ bản sau:

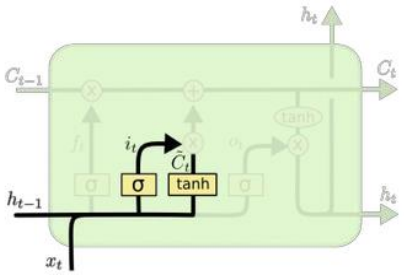
+ Bước đầu tiên của mô hình *LSTM* là quyết định xem thông tin nào chúng ta cần loại bỏ khỏi *cell state*. Một *sigmoid layer* gọi là "*forget gate layer*" – cổng chặn sẽ thực hiện tiến trình này. Đầu vào là h_{t-1} *và* x_t , đầu ra là một giá trị nằm trong khoảng [0, 1] cho *cell state* C_{t-1} (1 tương đương với "giữ lại thông tin", 0 tương đương với "loại bỏ thông tin").



$$f_t = \sigma\left(W_f \cdot [h_{t-1}, x_t] \; + \; b_f\right)$$

Hình 2.26: LSTM focus f

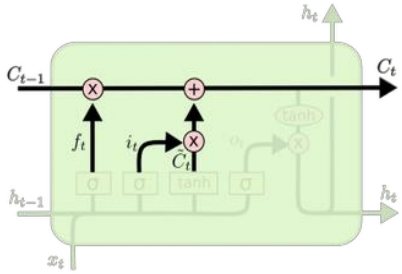
+ Bước tiếp theo, quyết định thông tin nào cần được lưu lại tại *cell state*, được thực hiện bởi *single sigmoid layer* được gọi là "*input gate layer*" quyết định các giá trị chúng ta sẽ cập nhật và một *tanh layer* tạo ra một *vector* ứng viên $\tilde{C}_t$ mới, được thêm vào trong ô trạng thái.



$$i_t = \sigma\left(W_i \cdot [h_{t-1}, x_t] \; + \; b_i\right)$$
$$\tilde{C}_t = \tanh(W_C \cdot [h_{t-1}, x_t] \; + \; b_C)$$

Hình 2.27: LSTM focus i

+ Kế tiếp, kết hợp hai thành phần này lại để cập nhật vào cell state. Lúc cập nhật vào cell state cũ C_{t-1} vào cell state mới C_t. Ta sẽ đưa *state* cũ vào hàm f_t , để quên đi những gì trước đó. Sau đó, ta sẽ thêm i_t*C_t. Đây là giá trị ứng viên mới, co giãn (*scale*) số lượng giá trị mà ta muốn cập nhật cho mỗi *state*.



$$C_t = f_t * C_{t-1} + i_t * \tilde{C}_t$$

Hình 2.28: LSTM focus c

+ Cuối cùng, ta cần quyết định xem thông tin *output* là gì. *Output* này cần dựa trên *cell state* của chúng ta, nhưng sẽ được lọc bớt thông tin. Đầu tiên, ta sẽ áp dụng *single sigmoid layer* để quyết định xem phần nào của *cell state* chúng ta dự định sẽ *output*. Sau đó, ta sẽ đẩy *cell state* qua *tanh* (đẩy giá trị vào khoảng - 1 và 1) và nhân với một *output sigmoid gate*, để giữ lại những phần ta muốn *output* ra ngoài [5].

Mô hình LSTM là một bước đột phá đạt được từ mô hình RNN. Mô hình này giải quyết triệt để vấn đề không xử lý được câu hỏi dài mà những mô hình chatbot Skype đang gặp phải.

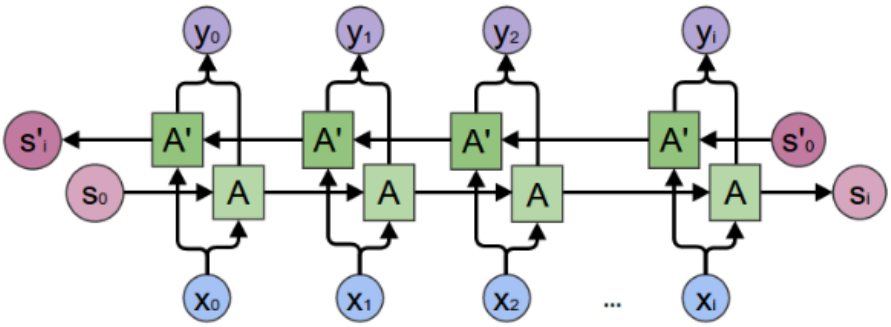
2.3.3. Mạng nơ ron dài ngắn song song (BiLSTM)

2.3.3.1. Giới thiệu sơ về mạng nơ ron dài ngắn 2 chiều

Một hướng tiếp cận với dữ liệu khác nữa, đó là sử dụng hai mạng nơ ron hồi quy theo hai chiều ngược nhau để xử lý (Bidirectional RNN) [7]. Một đơn vị RNN sẽ làm như thường lệ, tức là ta sẽ dùng nó để học các tín hiệu đầu vào từ thời điểm ban đầu tới thời điểm kết thúc (đi xuôi). Còn đơn vị RNN còn lại, ta sẽ đọc theo thứ tự thời điểm từ kết thúc trở lại ban đầu (đi ngược). Sau khi có cả hai kết quả, chúng sẽ được gom lại thành một để có thể dự đoán. Với ý tưởng như vậy, tại một thời điểm t bất kỳ, mạng sẽ có được các thông tin trước và sau thời điểm t ấy.

Do bản chất LSTM là cải tiến của RNN, cho nên ta có thể áp dụng nó và biến nó thành mạng nơ ron dài ngắn song song (BiLSTM). Mỗi LSTM sẽ vẫn có khả năng quên thông tin cũ (cổng quên), lọc thông tin mới (cổng đầu vào), hoặc giấu bớt kết quả (cổng đầu ra) như bình thường. Chính vì vậy, các thông tin từ quá khứ tới

tương lai của mạng BiLSTM đều có thể tự học để tự điều chỉnh. Dẫn tới việc với các bài toán mà ta cần biết nhiều hơn về ngữ cảnh hiện tại của nó, thì mạng BiLSTM cho kết quả tốt hơn [7].



Hình 2.29: Mạng Bi-RNN (có thể thế bằng BiLSTM) sau khi được "bung ra". Ta thấy đơn vị mạng A chính là mạng đi xuôi, và đơn vị mạng A' chính là mạng đi ngược.

2.3.3.2. *Cách dự đoán kết quả của mạng BiLSTM*

Như ý tưởng trên, ta sẽ có kết quả của 2 đơn vị mạng khác nhau (một cái chạy xuôi, và một cái chạy ngược). Chính vì vậy để có kết quả cuối cùng, ta phải gom được hai kết quả trên về một cái thống nhất. Một số cách đơn giản để làm điều này đó là:

- Tổng: Cộng từng tín hiệu đầu ra của hai mạng
- Tích: Nhân từng tín hiệu đầu ra của hai mạng
- Ghép: Hai vectơ tín hiệu đầu ra của hai mạng được ghép tiếp nhau (vectơ kết quả cuối sẽ có số chiều gấp đôi)
- Trung bình cộng: Trung bình cộng từng tín hiệu đầu ra của hai mạng
- Trung bình nhân: Trung bình nhân từng tín hiệu đầu ra

Thông thường, tùy bài toán mà ta dùng cách khác nhau. Tuy vậy, cách ghép hai vectơ là phổ biến nhất.

2.4. Hệ thống trả lời tự động Chatbot

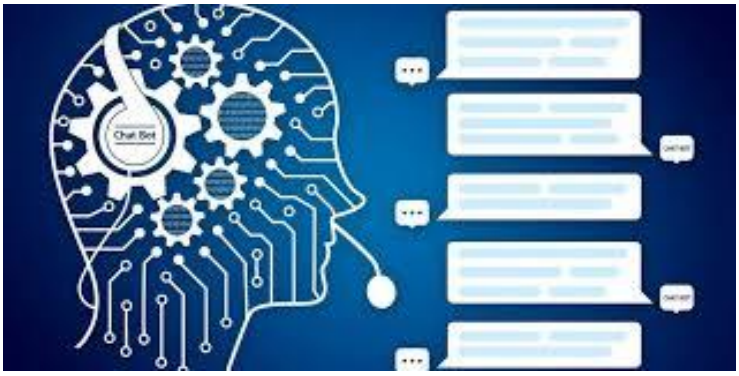
2.4.1. Tổng quan

Chatbot là một hệ thống trả lời tự động thông minh, hay là một chương trình mô phỏng cuộc trò chuyện của con người thông qua văn bản hoặc bằng giọng nói với máy. Người dùng có thể yêu cầu một câu hỏi hoặc thực hiện một lệnh và chatbot sẽ trả lời hoặc thực hiện các hành động được yêu cầu. Mức độ chuẩn xác và

tự nhiên của câu trả lời phụ thuộc vào khả năng xử lý dữ liệu đầu vào cũng như độ phức tạp của thuật toán lựa chọn đầu ra của hệ thống.

Hiện nay nhu cầu sử dụng chatbot đang ngày càng tăng lên, nhất là trong các hệ thống trực tuyến với số lượng lớn người dùng. Các hệ thống chatbot có thể được sử dụng để hỗ trợ hoặc thay thế cho nhân viên chăm sóc khách hàng trong một số tác vụ tự động hoá. Ví dụ, chatbot có thể tự động đưa ra câu trả lời cho khách hàng về các dịch vụ mà doanh nghiệp cung cấp. Sức nóng của chatbot hiện nay phần lớn là do những bước tiến vượt bậc trong ngành trí tuệ nhân tạo, nhất là trong những lĩnh vực học máy, xử lý tiếng nói và xử lý ngôn ngữ tự nhiên.

Nhiều nhà nghiên cứu đã sử dụng các kỹ thuật học máy để xây dựng Chatbot có khả năng hỗ trợ con người trò chuyện, nhắc nhở hay làm trợ lý công việc và có thể theo dõi tình trạng sức khỏe cá nhân mọi lúc, mọi nơi. Rất nhiều công ty lớn đã phát triển các trợ lý ảo có thể hiểu được ngôn ngữ tự nhiên của con người và tương tác được với con người một cách tự nhiên hơn, nhằm làm tăng chất lượng và hiệu quả trong việc chăm sóc khách hàng, giúp khách hàng có những trải nghiệm tốt nhất về sản phẩm và các dịch vụ mà họ được cung cấp.



Hình 2.30: Tổng quan Chatbot

2.4.2. Các hướng tiếp cận

Có 2 hướng tiếp cận chính trong bài toán xây dựng chatbot là: Hướng tiếp cận dựa trên luật (Rule-based) và Hướng tiếp cận dựa trên dữ liệu (Corpus-based)

- Với hướng tiếp cận Rule-based có các nghiên cứu như:
  - Pattern-action rules (Eliza)
  - A mental model (Parry)
- Hướng tiếp cận Corpus-based có các nghiên cứu như:
  - Information Retrieval
  - Deep neural nets

Trong hai hướng tiếp cận này, hướng tiếp cận dựa trên dữ liệu được nghiên cứu, triển khai nhiều trong những năm gần đây và trở thành hướng nghiên cứu chính.

Một số chatbot đã được giới thiệu đầu tiên như: ELIZA (1966); PARRY (1968) (The first system to pass the Turing test); ALICE; CLEVER hay Microsoft XiaoIce 小冰. Đến đầu năm 2016, các công ty lớn như Microsoft (Cortana); Google (Google Assistant); Facebook (M), Apple (Siri), Samsung (Viv), WeChat, Slack, …… cũng đã giới thiệu mô hình trợ lí ảo của mình. Hai trong số chatbot thông minh là Eugene Goostman và SmarterChild đã đẩy làn sóng chatbot lên cao.

2.4.3. Tình hình nghiên cứu

2.4.3.1. Các nghiên cứu ngoài nước

Hệ thống trả lời tự động đã được các nhà nghiên cứu quan tâm từ rất lâu, bao gồm các trường đại học, các viện nghiên cứu và các doanh nghiệp. Việc nghiên cứu về hệ thống trả lời tự động có ý nghĩa trong khoa học và thực tế. Đã có rất nhiều các hội nghị thường niên về xử lý ngôn ngữ tự nhiên, khai phá dữ liệu, xử lý dữ liệu lớn, tương tác người máy.

Trong những năm gần đây, phương pháp Deep Learning đã chứng minh lợi ích đáng kể cho nhiệm vụ xử lý ngôn ngữ tự nhiên; Các mô hình này đã sử dụng hầu hết các mạng thần kinh tái phát như LSTM và CNN để phân loại văn bản. Andreas và các cộng sự với nghiên cứu "Mạng học sâu CNN cho hệ thống trả lời câu hỏi" [8]. Jinfeng Rao và các công sự với nghiên cứu "Ước lượng tương phản nhiễu để lựa chọn câu trả lời với các mạng thần kinh sâu" trình bày trong Kỷ yếu của Quốc tế ACM lần thứ 25 về Hội nghị về Quản lý thông tin và tri thức New York 2016 [9]. Nal Kalchbrenner, Edward Grefenstette và Phil Blunom với nghiên cứu "Một mạng lưới thần kinh tích chập để mô hình hóa câu". Trong Kỷ yếu của Hội nghị thường niên lần thứ 52 của Hiệp hội Ngôn ngữ học tính toán (ACL-14) 2014 [10]. Tom Young và các công sự với nghiên cứu về " Xu hướng gần đây trong việc xử lý ngôn ngữ tự nhiên dựa trên nền tảng học sâu" trên tạp chí IEEE Computational Intelligence, 2018 [11][12]. Yoon Kim với nghiên cứu "Mạng lưới nơ ron chuyển đổi để phân loại câu" [13] sử dụng Mạng thần kinh chuyển đổi (CNN) và coi các câu hỏi là câu chung để đạt được hiệu suất mạnh mẽ đáng kể trong nhiệm vụ phân loại câu hỏi TREC.

2.4.3.2. Tình hình nghiên cứu trong nước

Ở Việt Nam, để hỗ trợ cho việc phát triển chatbot, cách đây hơn 20 năm, các nhà nghiên cứu, khoa học đã kế thừa những thành tựu của thế giới để đưa ra nhiều mô hình lý thuyết và cải tiến làm nền tảng cho việc phát triển các sản phẩm.

Nếu như vài năm trước, chatbot vẫn còn là một khái niệm xa vời thì đến nay nó đã được nhiều doanh nghiệp Việt, thậm chí các cơ quan nhà nước đẩy mạnh ứng dụng. Có thể kể đến như chatbot bán hàng ảo của FPT Shop giúp khách hàng tìm kiếm thông tin sản phẩm, gửi thông báo về các chương trình khuyến mãi và hỗ trợ đặt mua hàng trực tiếp nhanh chóng. VietA Bank sử dụng chatbot để tư vấn khách hàng các thông tin về lãi suất, tỷ giá, sản phẩm, biểu phí, quy trình mở thẻ… EVN Hà Nội ứng dụng chatbot để hỗ trợ khách hàng tra cứu tiền điện, lịch ghi chỉ số, lịch tạm ngừng cung cấp điện, đăng ký cấp điện mới và nhiều dịch vụ hữu ích khác. Công ty VHT ứng dụng công nghệ xử lý ngôn ngữ tự nhiên của FPT mở cho cộng đồng để phát triển hệ thống tự động liên hệ với khách hàng có khả năng liên hệ 15.000 khách hàng trong vòng 1 giờ, tương đương với sức làm việc của 500 người.

Không chỉ trong doanh nghiệp, chatbot hiện cũng đang được đẩy mạnh ứng dụng tại một số cơ quan nhà nước. Ví dụ như Sở Du lịch TP Đà Nẵng đã thí điểm thành công Chatbot Danang Fantasticity của Hakate giúp tra cứu thông tin du lịch tự động trên tin nhắn, Sở Giao thông TP Hồ Chí Minh cũng đã đưa vào sử dụng hệ chatbot do FPT phát triển nhằm cung cấp và giải đáp các thông tin về tình hình giao thông tới người dân. Hiện đã có gần 60 nghìn tài khoản thường xuyên tương tác với hệ thống này trên Zalo…

Để nâng cao hơn nữa hiệu quả hoạt động của chatbot, các nhà phát triển công nghệ đã tăng cường nghiên cứu và ứng dụng trí tuệ nhân tạo để các chatbot có thể thông minh hơn, giao tiếp tốt hơn nhờ cải thiện đáng kể bộ dữ liệu của mình qua các "kinh nghiệm" tích lũy được. Có thể kể đến như nền tảng tạo lập chatbot vừa ra đời với tên gọi QnA Bot Maker của FPT. Nền tảng này được phát triển dựa trên nền tảng trí tuệ nhân tạo FPT.AI cho phép người dùng dễ dàng tạo lập chatbot hỏi – đáp miễn phí. Với giao diện đồ họa người dùng, QnA Bot Maker có thể tích hợp với nhiều ứng dụng khác nhau như Facebook Messenger, Viber...

Hệ thống hỏi đáp tự động áp dụng cho ngôn ngữ Tiếng Việt đã được nhiều tác giả nghiên cứu trong đó có thể kể đến như:

Nguyễn Văn Tú và Lê Anh Cường 2016 [14] đề xuất phương pháp SVM và trích xuất đặc trưng cho hệ thống phân loại câu hỏi. Học viên cao học Nhữ Bảo Vũ (Đại học Quốc gia Hà Nội, 2016 [5]) áp dụng phương pháp học chuỗi liên tiếp xây dựng mô hình đối thoại cho tiếng Việt trên miền mở. Học viên Nguyễn Thị Thanh Hương (Đại học Thủ Dầu Một, 2019 [15]) đề xuất mạng nơ ron nhân tạo xây dựng hệ thống hỏi đáp tự động hỗ trợ giáo dục. Gần đây nhất tác giả Bùi Thanh Hùng với nghiên cứu "Phân loại câu hỏi tiếng Việt dựa trên học sâu" và "Kết hợp giữa phân loại câu hỏi với hệ thống bỏ dấu tự động cho hệ thống hỏi đáp tự động bằng phương pháp học sâu" [16][17].

2.4.3.3. Hướng đề xuất nghiên cứu

Có nhiều hướng xây dựng hệ thống trả lời tự động như: Áp dụng mô hình dịch máy, Xây dựng bộ luật, Truy xuất thông tin, Phân loại câu hỏi… Với hướng tiếp cận Phân loại câu hỏi là từ các câu hỏi và câu trả lời có sẵn, chúng ta cần phân loại câu hỏi đầu vào thuộc loại nào, từ đó sinh ra câu trả lời phù hợp. Có rất nhiều phương pháp giải quyết bài toán này như sử dụng kiến thức về ngôn ngữ, các bộ luật tự xây dựng, học máy SVM, Bayes, Maximum Entropy Models,…trong tất cả các phương pháp, học máy được sử dụng như một kỹ thuật đem lại hiệu quả cao [4][5][7][8]. Trong vấn đề phân loại các cặp câu hỏi - câu trả lời [8][11][13], mỗi cặp câu hỏi - câu trả lời được coi như là một văn bản và được biểu diễn trong mô hình không gian vectơ có số chiều rất lớn, điều này có thể được phân loại tốt bởi phương pháp học sâu – Bộ nhớ dài ngắn song song – Bidirectional Long Short Term Memory (BiLSTM) [15][16]. Chính vì vậy trong nghiên cứu của mình, chúng tôi sử dụng phương pháp học sâu BiLSTM để phân loại câu hỏi dựa trên không gian vector từ Word2vector. Nghiên cứu của chúng tôi khác với các nghiên cứu khác bằng cách phân loại câu hỏi tiếng Việt dựa trên Bộ nhớ dài ngắn song song (BiLSTM) và tích hợp các hệ thống trả lời câu hỏi tự động vào Hệ thống hỗ trợ tư vấn dịch vụ công. Đầu tiên, chúng tôi tạo ra một biểu diễn vectơ cho mỗi câu hỏi, sử dụng các vectơ đó để huấn luyện mô hình của chúng tôi bằng cách tiếp cận học tập sâu BiLSTM và đánh giá hiệu suất của mô hình phân loại của chúng tôi. Cuối cùng, chúng tôi đã tích hợp các hệ thống trả lời câu hỏi tự động vào Hệ thống hỗ trợ tư vấn dịch vụ công dựa trên phân loại Câu hỏi. Mô hình phân loại câu hỏi theo hướng mạng Bộ nhớ dài ngắn song song (BiLSTM) là một loại mạng nơ ron tái

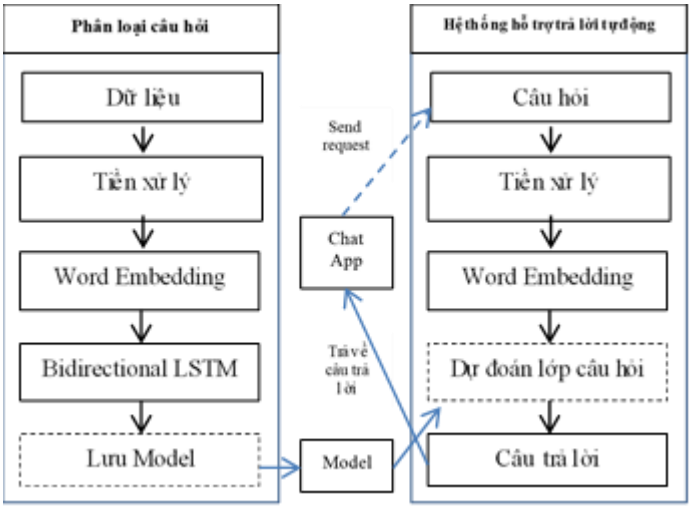
phát (RNN) đặc biệt, có khả năng ghi nhớ và bỏ qua các tính năng theo ngữ cảnh, những đặc điểm này rất hữu ích trong phân loại câu hỏi, được đề xuất nhằm huấn luyện và đánh giá kết quả bộ dữ liệu tiếng Việt được thu thập. Luận văn cũng đưa ra phương pháp đánh giá kết quả huấn luyện và so sánh kết quả thực nghiệm trên bộ dữ liệu thu thập và đồng thời đề xuất mô hình ứng dụng trên nền tảng Web-based để trực quan hóa câu trả lời từ các câu hỏi của người dân theo hướng phân loại câu hỏi theo từng chủ đề.

CHƯƠNG 3

MÔ HÌNH ĐỀ XUẤT

3.1. Tổng quan mô hình đề xuất

Mô hình trả lời tự động đề xuất trong luận văn được áp dụng dựa trên mô hình phân loại câu hỏi theo hướng mạng Bộ nhớ dài ngắn song song (BiLSTM) để huấn luyện dữ liệu và kết hợp các phương pháp đánh giá dựa trên đánh giá độ chính xác (Accuracy) để đưa ra mô hình dự đoán tối ưu nhất nhằm mục đích trả lời các câu hỏi của người dùng.



Hình 3.1: Đề xuất mô hình xây dựng chatbot

Mô hình đề xuất của chúng tôi được mô tả ở Hình 3.1 gồm 2 khối chính: Phân loại câu hỏi và xây dựng ứng dụng hỗ trợ trả lời tự động tại Sở Thông tin và Truyền thông tỉnh Bình Dương.

Trong module Phân loại câu hỏi, từ bộ dữ liệu thu thập: câu hỏi – câu trả lời đã được gán nhãn theo từng mục khác nhau sẽ được tiền xử lý tiền xử lý chuyển thành Word2vector sau đó đưa qua bộ phân loại sử dụng phương pháp học sâu BiLSTM. Model được lưu để sử dụng cho module Hệ thống hỗ trợ trả lời tự động. Hệ thống hỗ trợ trả lời tự động xây dựng trên nền tảng ứng dụng Web nhận câu hỏi từ người sử dụng, câu hỏi sẽ được tiền xử lý và sau đó chuyển thành Word2vector. Kết quả sẽ được đưa qua mô hình đã huấn luyện để dự đoán thuộc bộ câu hỏi nào, từ đó đưa ra câu trả lời phù hợp và trả về kết quả cho người sử dụng. Mô hình đề xuất được mô tả chi tiết thông qua các phần trình bày sau đây.

3.1.1. Mô hình huấn luyện dữ liệu tổng quát

Như mọi phương pháp ứng dụng của Machine Learning, việc đầu tiên là thu thập dữ liệu rồi sau đó qua bước tiền xử lý dữ liệu thô để tiến hành huấn luyện dữ liệu từ các phương pháp học máy như phân lớp dữ liệu, phân cụm, học sâu,… để sinh ra mô hình dự đoán kết quả nhằm trả lời các câu hỏi tương ứng của người dân.

Phương pháp huấn luyện được lựa chọn dựa trên các phương pháp học sâu để huấn luyện bộ dữ liệu, kết quả cuối cùng của việc huấn luyện là cho ra mô hình huấn luyện được lưu lại để thực hiện việc dự đoán kết quả. Quá trình huấn luyện sẽ được thực hiện liên tục nhằm mục đích tìm kiếm mô hình tối ưu nhất thông qua việc thay đổi bộ tham số huấn luyện đầu vào và phương pháp đánh giá kết quả dự đoán dựa trên bộ tham số đầu vào.

+ Bước 1: Thu thập dữ liệu

Thực hiện việc thu thập thông tin dữ liệu là các câu hỏi của các người dân liên quan đến Sở Thông tin và Truyền thông như: thông tin thủ tục hành chính do Sở Thông tin và Truyền thông phụ trách, chương trình đào tạo công nghệ thông tin cho cán bộ công chức viên chức trong tỉnh, thông tin về lãnh đạo Sở và các phòng ban, đơn vị trực thuộc,…

+ Bước 2: Chuẩn hóa dữ liệu

Sau khi thu thập dữ liệu hoàn thành, việc chuẩn hóa dữ liệu sẽ được thực hiện nhằm loại bỏ những giá trị không phù hợp, ví dụ như các từ dừng (stop word), các ký hiệu dấu câu,….; khôi phục các từ lỗi,….

+Bước 3: Lựa chọn thuật toán huấn luyện

Tiến hành lựa chọn một trong các giải thuật của Machine Learning để tiến hành huấn luyện bộ dữ liệu huấn luyện.

+ Bước 4: Phân chia bộ dữ liệu huấn luyện và dự đoán

Phân chia bộ dữ liệu thu thập đã được chuẩn hóa theo tỷ lệ n:m, n phần dùng để thực hiện việc huấn luyện, m phần dùng để dự đoán.

+ Bước 5: Huấn luyện bộ dữ liệu huấn luyện

Với phương pháp huấn luyện đã được chọn lựa, tiến hành huấn luyện dữ liệu huấn luyện với các tham số huấn luyện phù hợp.

+ Bước 6: Lưu trữ mô hình dự đoán

Sau khi thực hiện việc huấn luyện dữ liệu hoàn tất, tiến hành lưu trữ mô hình huấn luyện để phục vụ cho việc dự đoán sau này.

3.1.2. Mô hình dự đoán kết quả

Để tiến hành việc dự đoán kết quả, từ mô hình huấn luyện đã lưu trữ trước đó, tiến hành dự đoán kết quả trên dữ liệu dự đoán, sau đó tiến hành đánh giá kết quả dự đoán.

3.1.3. Mô hình huấn luyện dữ liệu - dự đoán kết quả

Quá trình huấn luyện và dự đoán kết quả sẽ được lặp đi lặp lại cho đến khi mô hình dự đoán kết quả là tối ưu nhất thông qua việc đánh giá kết quả dự đoán. Mỗi quá trình huấn luyện dữ liệu, các tham số huấn luyện sẽ được thay đổi để tìm ra kết mô hình huấn luyện tối ưu nhất. Mô hình huấn luyện này sẽ được sử dụng để dự đoán kết quả cho các bộ dữ liệu mà người dùng nhập vào.

+ **Bước 1:** Thu thập dữ liệu.

+ **Bước 2:** Chuẩn hóa dữ liệu.

+ **Bước 3:** Lựa chọn thuật toán huấn luyện.

+ **Bước 4**: Chia bộ dữ liệu theo tỷ lệ n:m.

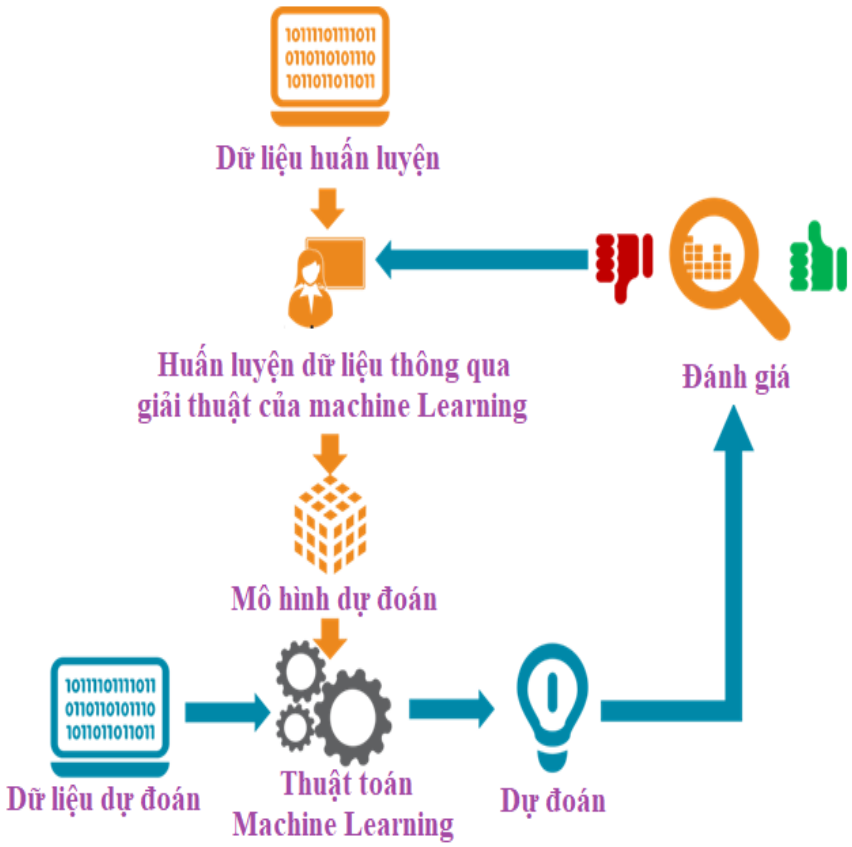
+ **Bước 5**: Xây dựng mô hình dự đoán bằng cách đưa vào các tham số huấn luyện phù hợp, sau đó tiến hành huấn luyện trên bộ dữ liệu huấn luyện.

+ **Bước 6:** Lưu trữ mô hình dự đoán để phục vụ cho việc dự đoán kết quả sau này.

+ **Bước 7:** Từ thuật toán huấn luyện được chọn lựa ở Bước 3, kết hợp với mô hình dự đoán đã lưu trữ ở Bước 6, tiến hành dự đoán kết quả với bộ dữ liệu dự đoán.

+ **Bước 8:** Đánh giá kết quả dự đoán ở Bước 7.

- Nếu kết quả dự đoán thỏa một tiêu chí đánh giá thì kết thúc, mô hình dự đoán sẽ được sử dụng để thực hiện việc dự đoán kết quả từ bộ dữ liệu do người dùng đưa vào.
- Nếu kết quả chưa đạt thì tiến hành quay lại Bước 4.



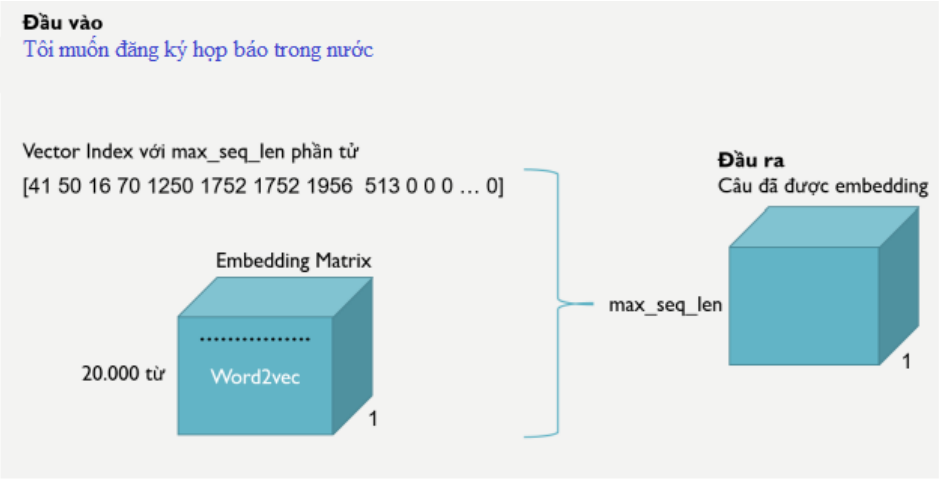
Hình 3.2: Quy trình huấn luyện dữ liệu - dự đoán kết quả

3.2. Các đặc trưng của mô hình đề xuất

3.2.1. Từ nhúng – Word embedding

Word Embedding là quá trình đưa các từ trong câu về dạng để mô hình toán có thể hiểu được. Cụ thể là từ dạng text, các từ sẽ được chuyển về dạng vector đặc trưng để đưa vào mô hình LSTM. Trước khi đưa về dạng vector các câu cần được chuẩn hóa về độ dài. Trong nghiên cứu này chúng tôi sử dụng mô hình Word2vector – skipgram từ bộ dữ liệu đã được huấn luyện sẵn của Facebook AI research – fastText.

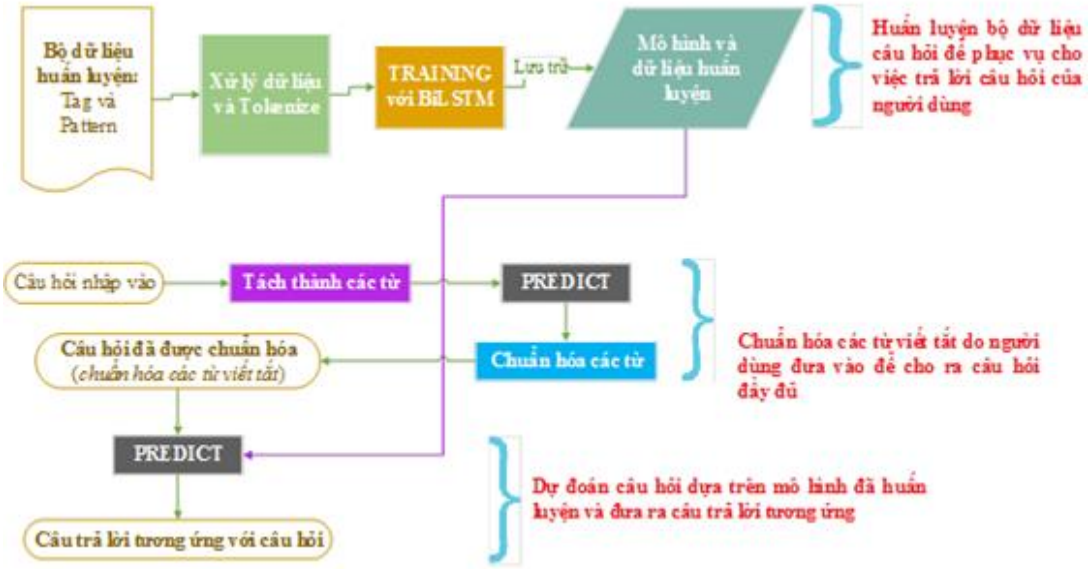
Khi một câu được đưa vào, trước tiên nó sẽ được embedding theo số index tương ứng của nó trong từ điển. Sau đó, dựa trên từ điển và kết quả Word2vector thu được tôi embedding toàn bộ câu dưới dạng ma trận như Hình 3.3 dưới đây. Trong mô hình này max_seq_len là độ dài của câu, tất cả các câu trong tập huấn luyện đều được cắt hoặc nối để có độ dài max_seq_len.



Hình 3.3: Quá trình embedding của một câu

3.2.2. Mô hình học sâu BiLSTM xây dựng hệ thống hỏi đáp tự động

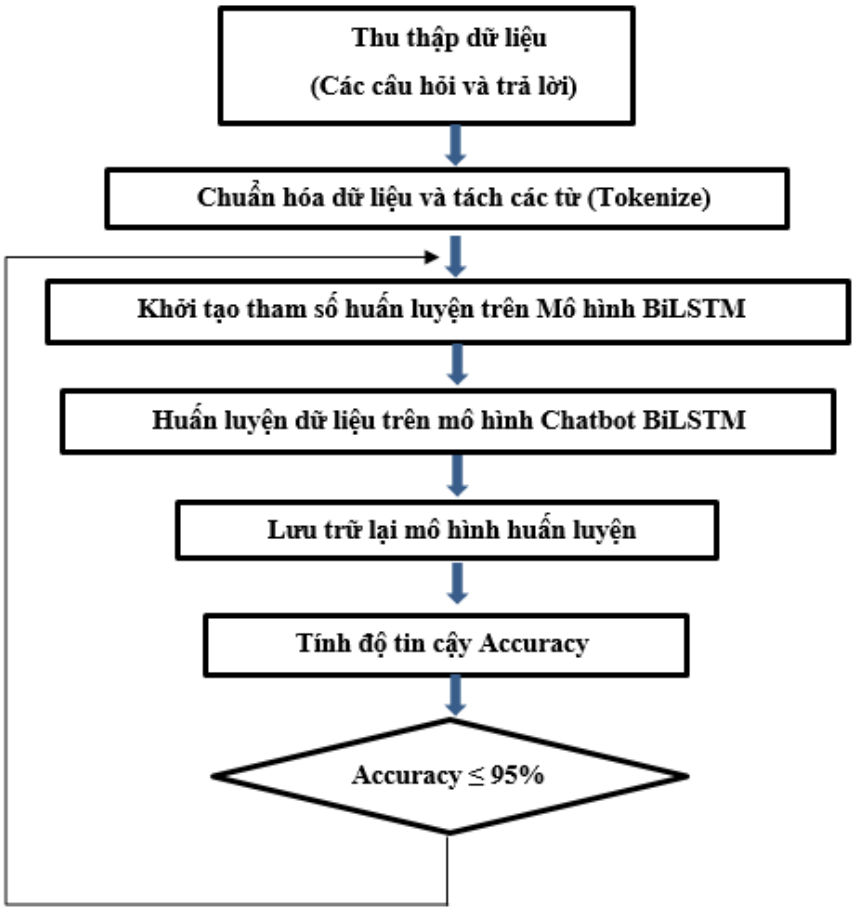
Mô hình này được trình bày tổng quát trong Hình 3.4.



Hình 3.4: Mô hình học sâu BiLSTM xây dựng hệ thống hỏi đáp tự động

3.2.2.1. Mô hình huấn luyện dữ liệu với BiLSTM

Đầu tiên dữ liệu được thu thập rồi sau đó qua bước tiền xử lý dữ liệu thô để tiến hành huấn luyện dữ liệu bằng phương pháp học sâu BiLSTM từ đó sinh ra mô hình dự đoán kết quả nhằm trả lời các câu hỏi tương ứng. Mô hình này được trình bày chi tiết ở Hình 3.5.



Hình 3.5: Mô hình huấn luyện dữ liệu với BiLSTM

3.2.2.2. Mô hình dự đoán kết quả

Mô hình trả lời các câu hỏi của người dùng được xây dựng dựa trên mô hình phân loại câu hỏi bằng phương pháp học sâu. Mô hình này được trình bày ở Hình 3.6.

Để trả lời các câu hỏi của người dùng, việc đầu tiên đó là quá trình huấn luyện dữ liệu đã được chuẩn hóa sau quá trình thu thập. Tuy nhiên, trong quá trình đưa thông tin đầu vào là các câu hỏi, một số người dùng có thể nhập các từ viết tắt không theo một quy luật nào, do đó việc huấn luyện dữ liệu là các từ viết tắt được thực hiện song song với việc huấn luyện dữ liệu thu thập là các câu hỏi và trả lời. Quá trình huấn luyện hoàn tất sẽ được lưu trữ thành để thực hiện cho việc dự đoán các câu hỏi của người dùng.

Quá trình trả lời các câu hỏi của người dùng được thực hiện như sau:

+ Nhận dữ liệu là câu hỏi của người dùng.

+ Chuẩn hóa các từ viết tắt trong câu hỏi bằng cách tách các từ trong câu hỏi và tiến hành dự đoán từ đầy đủ của từ (nếu có).

+ Ghép các từ lại thành câu hoàn chỉnh theo trình tự.

+ Tiến hành dự đoán câu trả lời dựa trên câu hoàn chỉnh để trả lời cho người dùng.

Hình 3.6: Mô hình dự đoán kết quả

3.3. Đánh giá quá trình huấn luyện và dự đoán kết quả

Để đánh giá hiệu quả của mô hình huấn luyện, một tham số được đề xuất nhằm đánh giá hiệu năng của mô hình đề xuất đó là thông số Accuracy hoặc F1-Score. Quá trình huấn luyện được thực hiện theo tuần tự các bước như sau:

- **Bước 1**: Chuẩn hóa dữ liệu đầu vào.

Với dữ liệu đầu vào là các câu hỏi, sử dụng kỹ thuật tokenize và loại bỏ các từ dừng (stop word).

- **Bước 2**: Khởi tạo các tham số huấn luyện.

Khởi tạo các tham số huấn luyện: activation, optimizer, batch_size, loss,…

- **Bước 3**: Huấn luyện dữ liệu.

Sử dụng mô hình mạng nơ ron sâu để tiến hành huấn luyện với bộ tham số huấn luyện đã khởi tạo.

- **Bước 4**: Lưu trữ mô hình dự đoán.

Tiến hành lưu trữ mô hình dự đoán với các tham số huấn luyện đã được khởi tạo trước đó để thực hiện cho việc dự đoán sau này.

- **Bước 5**: Đánh giá hiệu quả của việc huấn luyện.

Sử dụng thông số accuracy để đánh giá mô hình huấn luyện, nếu tham số accuracy đạt trên 95% thì tiến hành kết thúc quá trình huấn luyện, nếu ngược lại thì quay lại Bước 2.

Hình 3.7: Quy trình đánh giá quá trình huấn luyện và dự đoán kết quả

CHƯƠNG 4

THỰC NGHIỆM

4.1. Dữ liệu

Tập dữ liệu cho bài toán "Xây dựng hệ thống hỏi đáp tự động hỗ trợ công tác tư vấn dịch vụ hành chính công tại Sở Thông tin và Truyền thông tỉnh Bình Dương" được xây dựng dựa trên văn bản về các thủ tục hành chính do sở Thông tin và Truyền thông quản lý và các câu hỏi liên quan đến văn bản này. Dữ liệu này bao gồm 4 lĩnh vực với 37 thủ tục hành chính cấp tỉnh [18] [19] và những thông tin chung liên quan đến Sở Thông tin và Truyền thông. Quy trình thực hiện thu thập dữ liệu được trình bày như ở phần tiếp sau đây.

4.1.1. Quy trình thực hiện

Bước 1: Chuẩn bị dữ liệu.

Với dữ liệu thu thập, các câu hỏi được chuyển vào JSON theo cấu trúc như sau: các câu hỏi được đưa vào trường patterns, các câu trả lời được đưa vào trường responses, lớp câu hỏi và trả lời được đưa vào trường tag.

Bước 2: Từ file JSON, tiến hành loại bỏ các từ dừng và tiến hành tokenize dữ liệu ở tag **Pattern** bằng cách sử dụng thư viện **nltk** vector hóa các từ sau đó lưu vào bộ từ điển.

Bước 3: Khởi tạo các tham số và tiến hành huấn luyện dữ liệu. Sau đó thực hiện lưu mô hình để thực hiện việc dự đoán sau này

Bước 4: Tiến hành đánh giá độ chính xác accuracy, nếu độ chính xác ≤ 95% thì tiếp tục thực hiện lại bước 3. Ngược lại thì chuyển sang bước 5.

Bước 5: Với câu được nhập vào từ người sử dụng, tiến hành loại bỏ các từ dừng và tách từ. Sau đó tiến hành dự đoán câu trả lời dựa trên mô hình đã lưu ở bước 4. Kết quả dự đoán là lớp tương ứng với câu hỏi được đưa vào, khi đó sẽ chọn lựa câu trả lời ngẫu nhiên ứng với lớp đã được dự đoán.

4.1.2. Dữ liệu thực nghiệm

Dữ liệu thực nghiệm là dữ liệu được thu thập là những câu hỏi thường gặp của người dân liên quan đến các thủ tục hành chính do sở Thông tin và Truyền thông quản lý. Dữ liệu này bao gồm 4 lĩnh vực với 37 thủ tục hành chính cấp tỉnh [18] [19] và những thông tin chung liên quan đến Sở Thông tin và Truyền thông. Dữ liệu sau đó được chuyển thành các file JSON để thực hiện việc chuẩn hóa dữ

liệu. Đây là bộ dữ liệu phục vụ cho việc huấn luyện để trả lời các câu hỏi của người dân, dữ liệu trên file JSON được mô tả chi tiết trong Bảng 4.1.

Phân lớp câu hỏi	**Số lớp con**
Báo chí	31
Bưu chính	26
Phát thanh truyền hình - Thông tin điện tử	53
Xuất bản	48
Thông tin chung	24

Bảng 4.1 Bộ dữ liệu thu thập về thông tin của Sở Thông tin và Truyền thông

Bộ dữ liệu thu thập gồm 540 câu hỏi với 200 câu trả lời được chia thành hai tập huấn luyện và kiểm tra theo tỷ lệ 4:1, tức là 4 phần để huấn luyện (432 câu) và một phần để kiểm tra (108 câu) được mô tả trong Bảng 4.2.

Tên bộ dữ liệu	**Số câu**
Số câu hỏi	540
Số câu trả lời	200
Số lớp con	179
Dữ liệu huấn luyện	432
Dữ liệu kiểm tra	108

Bảng 4.2 Dữ liệu huấn luyện

Dữ liệu được tổ chức trong Excel với cấu trúc các mục như sau:

	A	B	C	D	E	F	G
	stt	question	answer	entity_faq_keyword	sample	entity_faq_subject	entity_faq_detail
232		Cách thực hiện TTHC Đăng ký hoạt động co	Trả lời: Hồ sơ nộp và trả kết quả trực tiếp tại Bộ phận tiếp nhậ	41101	Làm sao thực hiện TTHC Đăng ký hoạt động cơ sở in.. ?		
233		Thời gian giải quyết TTHC Đăng ký hoạt động	Trả lời: 03 ngày làm việc, kể từ ngày nhận đủ hồ sơ hợp lệ. .	41102	TTHC Đăng ký hoạt động cơ sở in.. mất bao lâu ?		
234		TTHC Đăng ký hoạt động cơ sở in.. yêu cầu t	Trả lời: Không. (Nếu thực hiện theo dịch vụ bưu chính công íc	41103	TTHC Đăng ký hoạt động cơ sở in.. có tính phí không ?		
235		Các bước thực hiện TTHC Đăng ký hoạt độn	Trả lời: Cơ sở in thực hiện chế bản, in, gia công sau in sản ph	41104	TTHC Đăng ký hoạt động cơ sở in.. thực hiện như thế nào ?		
236		Hồ sơ cần nộp TTHC Đăng ký hoạt động cơ	Trả lời: 1 Tờ khai đăng ký hoạt động cơ sở in (theo mẫu) 2 S	41105	TTHC Đăng ký hoạt động cơ sở in.. cần nộp những gì ?		
237		Yêu cầu đối với TTHC Đăng ký hoạt động co	Trả lời: 1 Có thiết bị phù hợp để thực hiện một hoặc nhiều cô	41106	TTHC Đăng ký hoạt động cơ sở in.. có yêu cầu gì không ?		
238		Căn cứ pháp lý của TTHC Đăng ký hoạt động	Trả lời: Nghị Định 60/2014/NĐ-CP Quy định về hoạt động in.N	41107	Tính pháp lý của TTHC Đăng ký hoạt động cơ sở in.. ?		
239		Cách thực hiện TTHC Đăng ký hoạt động ph	Trả lời: Hồ sơ nộp và trả kết quả trực tiếp tại Bộ phận tiếp nhậ	41201	Làm sao thực hiện TTHC Đăng ký hoạt động phát hành xuất bản phẩm.. ?		
240		Thời gian giải quyết TTHC Đăng ký hoạt động	Trả lời: 07 ngày làm việc, kể từ ngày nhận đủ hồ sơ hợp lệ .	41202	TTHC Đăng ký hoạt động phát hành xuất bản phẩm.. mất bao lâu ?		
241		TTHC Đăng ký hoạt động phát hành xuất bản	Trả lời: Không. (Nếu thực hiện theo dịch vụ bưu chính công íc	41203	TTHC Đăng ký hoạt động phát hành xuất bản phẩm.. có tính phí không ?		
242		Các bước thực hiện TTHC Đăng ký hoạt độn	Trả lời: Trước khi hoạt động 15 ngày, cơ sở phát hành xuất bả	41204	TTHC Đăng ký hoạt động phát hành xuất bản phẩm.. thực hiện như thế nào ?		
243		Hồ sơ cần nộp TTHC Đăng ký hoạt động phá	Trả lời: 1 Đơn đăng ký hoạt động phát hành xuất bản phẩm (tl	41205	TTHC Đăng ký hoạt động phát hành xuất bản phẩm.. cần nộp những gì ?		
244		Yêu cầu đối với TTHC Đăng ký hoạt động ph	Trả lời: 1 Người đứng đầu cơ sở phát hành phải thường trú tạ	41206	TTHC Đăng ký hoạt động phát hành xuất bản phẩm.. có yêu cầu gì không ?		
245		Căn cứ pháp lý của TTHC Đăng ký hoạt động	Trả lời: Luật Xuất bản 19/2012/QH13 ngày 20/11/2012Nghị đị	41207	Tính pháp lý của TTHC Đăng ký hoạt động phát hành xuất bản phẩm.. ?		
246		Cách thực hiện TTHC Đăng ký sử dụng máy	Trả lời: Hồ sơ nộp và trả kết quả trực tiếp tại Bộ phận tiếp nhậ	41301	Làm sao thực hiện TTHC Đăng ký sử dụng máy photocopy màu, máy in có chức năng photocopy màu.. ?		
247		Thời gian giải quyết TTHC Đăng ký sử dụng m	Trả lời: 05 ngày làm việc, kể từ ngày nhận đủ hồ sơ hợp lệ. .	41302	TTHC Đăng ký sử dụng máy photocopy màu, máy in có chức năng photocopy màu.. mất bao lâu ?		
248		TTHC Đăng ký sử dụng máy photocopy màu,	Trả lời: Không. (Nếu thực hiện theo dịch vụ bưu chính công íc	41303	TTHC Đăng ký sử dụng máy photocopy màu, máy in có chức năng photocopy màu.. có tính phí không ?		
249		Các bước thực hiện TTHC Đăng ký sử dụng	Trả lời: Trước khi sử dụng máy photocopy màu, máy in có chứ	41304	TTHC Đăng ký sử dụng máy photocopy màu, máy in có chức năng photocopy màu.. thực hiện như thế nào ?		
250		Hồ sơ cần nộp TTHC Đăng ký sử dụng máy p	Trả lời: 1 Đơn đăng ký sử dụng máy photocopy màu, máy in c	41305	TTHC Đăng ký sử dụng máy photocopy màu, máy in có chức năng photocopy màu.. cần nộp những gì ?		
251		Yêu cầu đối với TTHC Đăng ký sử dụng máy	Trả lời: Không .	41306	TTHC Đăng ký sử dụng máy photocopy màu, máy in có chức năng photocopy màu.. có yêu cầu gì không ?		

Hình 4.1: Mô tả về bộ dữ liệu được lưu trữ trên Excel

Một câu hỏi gồm 1 hoặc nhiều câu trả lời, các câu hỏi được gán nhãn vào các lớp khác nhau và được lưu trữ theo từng dòng, trong đó:

+ Phân lớp (Tag): lấy từ cột **entity_faq_keyword** của file Excel.

⇨ Phân lớp các câu hỏi khi đưa vào huấn luyện.

+ Câu hỏi huấn luyện (Pattern): lấy từ cột **question** và cột **sample**.

⇨ Dữ liệu dùng để training

+ Câu trả lời (Response): lấy từ cột **answer**.

⇨ Dùng để trả lời các câu hỏi tương ứng của người dùng.

4.2. Xử lý dữ liệu

Dữ liệu sau khi thu thập tiến hành xử lý theo các bước như sau:

- Bước 1: Đọc dữ liệu từ file Json và loại bỏ các từ dừng.

Dữ liệu từ file excel đã thu thập được chuyển sang dạng file Json tiến hành tách từ, loại bỏ các từ dừng và lưu trữ vào dữ liệu theo hai bộ câu hỏi và trả lời theo quy tắc:

+ Cột question, sample được đưa vào bộ dữ liệu câu hỏi, cột answer được đưa vào bộ dữ liệu trả lời.

+ Các câu hỏi và câu trả lời phải tương ứng theo chỉ mục với trên file json tương ứng với các thẻ patterns.

Phân chia tập huấn luyện (train) và kiểm tra (test).

Chia bộ dữ liệu thành hai tập huấn luyện và kiểm tra theo tỷ lệ 4:1, tức là 4 phần để huấn luyện và một phần để kiểm tra, quá trình phân chia được thực hiện theo nguyên tắc sau:

+ Dữ liệu được chọn vào tập huấn luyện, kiểm tra được chọn một cách ngẫu nhiên.

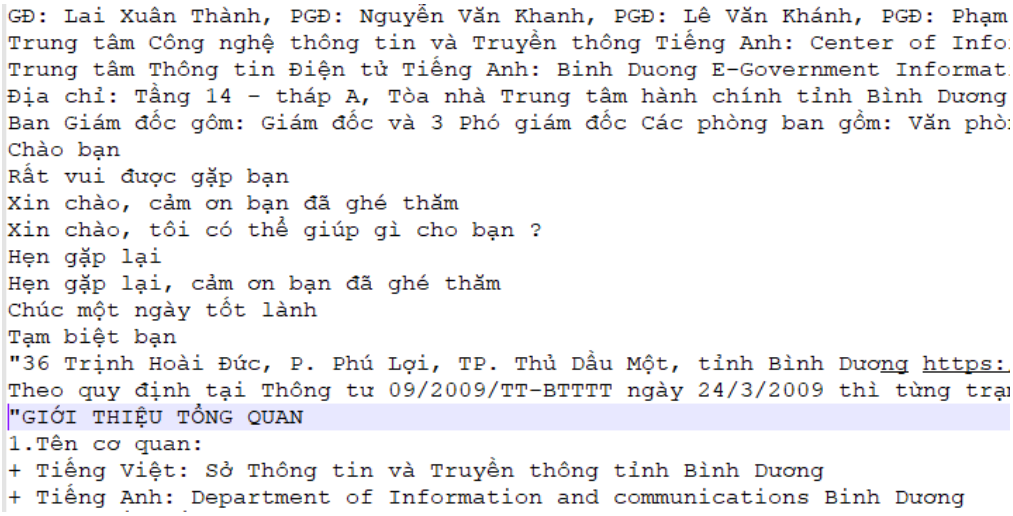
+ Các câu hỏi và câu trả lời phải tương ứng theo chỉ mục với trên file json tương ứng với các thẻ patterns.

+ Dữ liệu huấn luyện và kiểm tra được lưu trữ vào 4 file được trình bày trong các Hình 4.2, 4.3:

- Bộ huấn luyện: 1 file chứa bộ câu hỏi, 1 file chứa bộ câu trả lời.
- Bộ kiểm tra: 1 file chứa bộ câu hỏi, 1 file chứa bộ câu trả lời.

```
chào bạn ?
xin chào ?
Có ai ở đó không ?
Cho mình hỏi xíu ?
tạm biệt bạn ?
bye
Goodbye
Hẹn gặp lại
Địa chỉ Trung tâm Công nghệ Thông tin và truyền thông ?
Học bồi dưỡng CNTT ở đâu ?
Xin hỏi Sở TTTT có thể cung cấp thông tin chính thức về ảnh hu
Trạm BTS đến sức khỏe con người không ?
Chức năng nhiện vụ của Sở Thông tin và Truyền thông ?
Giới thiệu thổng quan về Sở Thông tin và Truyền thông ?
Thông tin  lãnh đạo  Sở  TTTT ?
Giới thiệu ban giám đốc Sở  TTTT ?
Thông tin liên hệ phòng Công nghệ thông tin ?
Thông tin liên hệ lĩnh vực  Công nghệ thông tin ?
Thông tin liên hệ phòng Báo chí – Xuất bản  ?
Thông tin liên hệ lĩnh vực Báo chí  ?
```

Hình 4.2: Bộ câu hỏi – training



Hình 4.3: Bộ câu trả lời – training

- **Bước 2**: Tách từ (tokenize) và lưu bộ từ điển.

Từ bộ dữ tiến hành tách các từ trong bộ câu hỏi trên file json tương ứng với các thẻ patterns.:

Các từ lưu trong file được sắp xếp theo số lượng từ xuất hiện trong bộ câu hỏi và trả lời từ cao đến thấp.

- **Bước 3**: Word2vector sử dụng bộ từ đã được huấn luyện sẵn fastText của Facebook AI research.

4.3. Huấn luyện

Thực hiện việc huấn luyện dữ liệu trên mô hình với bộ dữ liệu huấn luyện. Đối với tập dữ liệu tiền xử lý, chúng tôi đã sử dụng Tokenize tiếng Việt của Pyvi (0.0.0.9 - Tran Viet Trung 2016), Pre-train word embeddings tiếng Việt của fastText. Từ bộ dữ liệu đem đi huấn luyện, chúng tôi lọc lấy các từ Word2vector trong bộ fastText, với các từ ghép không có trong bộ từ điển chúng tôi sẽ tìm từ các từ đơn rồi ghép lại, các từ không có sẽ được khởi tạo ngẫu nhiên. Chúng tôi sử dụng Tensorflow framework và các thư viện học sâu của Keras trong mô hình huấn luyện của mình.

Các tham số được sử dụng trong mô hình LSTM và BiLSTM:

Số nút ẩn: 128,

Drop out: 0.2,

Kích hoạt chức năng trong lớp đầu ra: Sigmoid,

Số vòng lặp huấn luyện (epochs): 300,

Batch size: 500,

Tối ưu hóa: Adam,

Loss function: Categories cross entropy.

Chúng tôi xây dựng ứng dụng bằng HTML, CSS và Flask trong ngôn ngữ lập trình Python.

4.4. Đánh giá

Trên bộ dữ liệu thu thập với 5 nhãn bao gồm 179 lớp con với 88 câu hỏi và 35 câu trả lời tương ứng. Hiệu suất trong phân loại câu hỏi được đánh giá bởi độ chính xác của trình phân loại cho tất cả các lớp theo công thức:

$$\text{Độ chính xác (Accuracy)} = \frac{\#\text{số câu hỏi phân lớp đúng}}{\#\text{số câu hỏi}} \qquad (4.1)$$

Trong trường hợp một câu hỏi chỉ có một lớp, một câu hỏi được phân loại chính xác nếu nhãn dự đoán giống với nhãn thật. Nếu một câu hỏi được phân loại thành nhiều lớp, chúng tôi sẽ lấy lớp có kết quả cao nhất.

Để đánh giá mô hình đề xuất của chúng tôi, chúng tôi đã so sánh kết quả với mô hình LSTM một cách riêng biệt như trong Bảng 4.3.

Phương pháp	**Độ chính xác**
LSTM	95.24
BiLSTM	**97.36**

Bảng 4.3 Kết quả trong phân loại câu hỏi

Kết quả này cho thấy mô hình đề xuất của chúng tôi BiLSTM có kết quả tốt nhất. Khi có kết quả từ Phân loại câu hỏi, chúng tôi sẽ tích hợp nó trong Hệ thống hỗ trợ tư vấn thực hiện thủ tục hành chính. ICTbot là tên của ứng dụng của chúng tôi. ICTbot mới được triển khai thí nghiệm, chúng tôi sẽ triển khai hệ thống vào thực tế và đánh giá kết quả dựa trên bảng khảo sát phân tích ý kiến người sử dụng.

Khi chúng tôi phân tích kết quả của các hệ thống trả lời câu hỏi tự động, chúng tôi thấy rằng một trong những nguồn lỗi lớn nhất có xu hướng gắn thẻ thực thể không chính xác. Trong trường hợp một câu trả lời của ứng viên có quá nhiều thực thể thuộc loại yêu cầu, nó thường bị phân loại sai.

Có một số lỗi do những câu hỏi mơ hồ làm cho việc phân lớp câu hỏi sai. Ngoài ra còn một số lỗi do câu hỏi và câu trả lời được gắn thẻ tương quan không chính xác trong bộ dữ liệu. Chúng tôi đã chỉnh sửa các lỗi trên và tiến hành huấn luyện lại. Ứng dụng ICTBot được xây dựng trên bộ dữ liệu chuẩn để có được kết quả tốt nhất. ICTBot đã hoạt động hiệu quả trong việc hỗ trợ công tác tư vấn dịch

vụ hành chính công tại Sở Thông tin và Truyền thông tỉnh Bình Dương sau một thời gian thử nghiệm. Kết quả thử nghiệm được đánh giá qua hệ thống phản hồi của người sử dụng ứng dụng này được mô tả chi tiết trong Bảng 4.4 và Bảng 4.5.

Thời gian thử nghiệm	14 ngày
Số lượt người tham gia	50
Số ý kiến phản hồi	50

Bảng 4.4 Tổng hợp khảo sát ứng dụng ICTBot

Xếp loại	Số lượng đánh giá
Tốt	40
Cần cải thiện	5
Không ý kiến	5

Bảng 4.5 Bảng Kết quả đánh giá ứng dụng ICTBot

4.5. Xây dựng ứng dụng Chatbot trên nền tảng web

Ứng dụng Web trực quan hóa kết quả gồm 3 menu chính:

- ICTbot.
- Phân tích dữ liệu.
- Đánh giá kết quả.

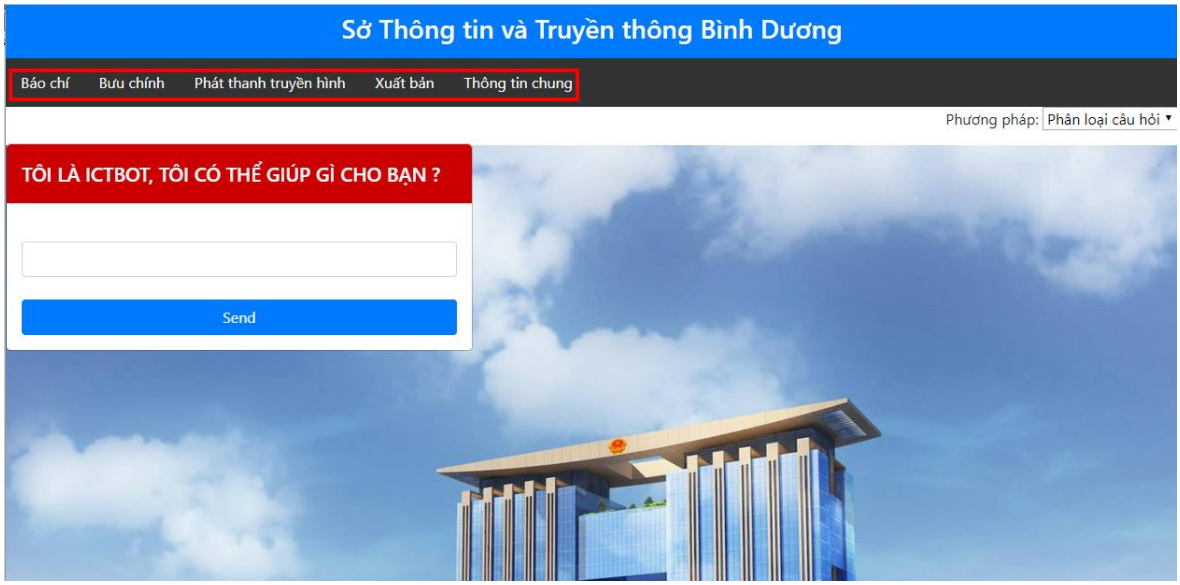
Ứng dụng được xây dựng dựa trên nền tảng Flask kết nối với Python Server, và các công cụ xây dựng web như: HTML, CSS, Bootstrap, Javascript.

Giao diện chính của chương trình được trình bày như Hình 4.4 dưới đây.

Hình 4.4: Giao diện Web - Chọn lựa chức năng của chương trình

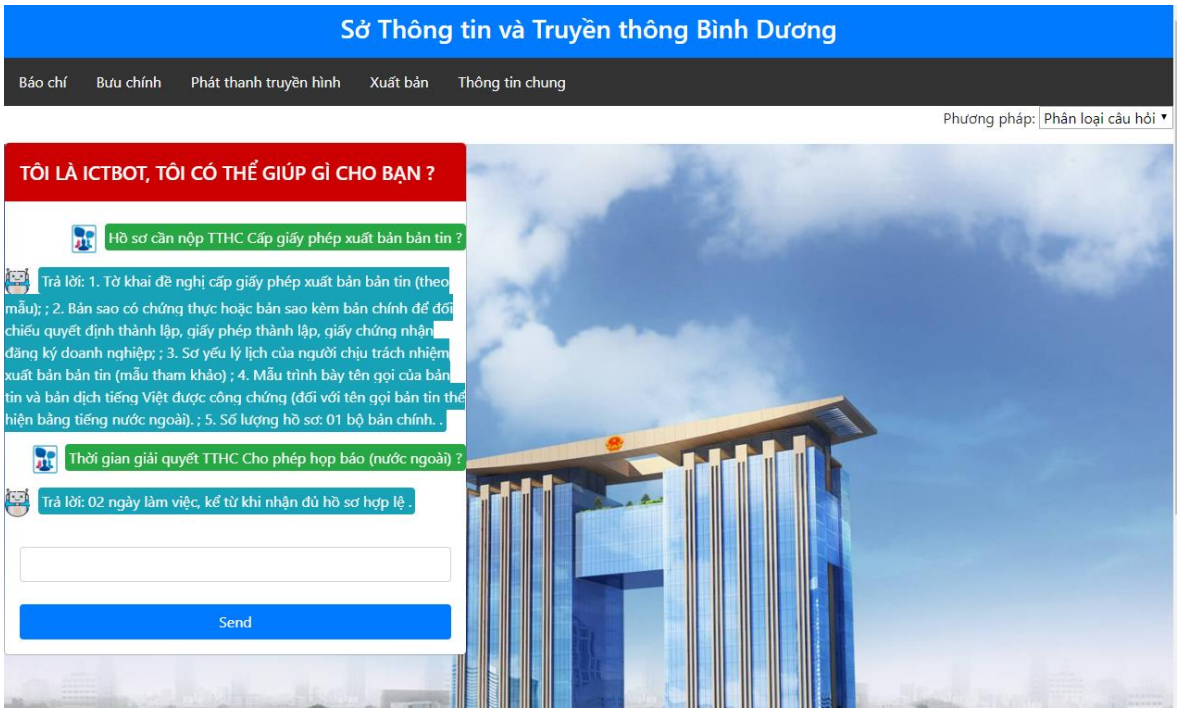
Từ màn hình chính người dùng chọn ICTBot để vào ứng dụng trả lời tự động.

Người dùng có thể chọn lựa một trong các danh mục để hỏi như Báo chí, Bưu chính, Phát thanh truyền hình, Xuất bản và thông tin chung của Sở Thông tin và Truyền thông.



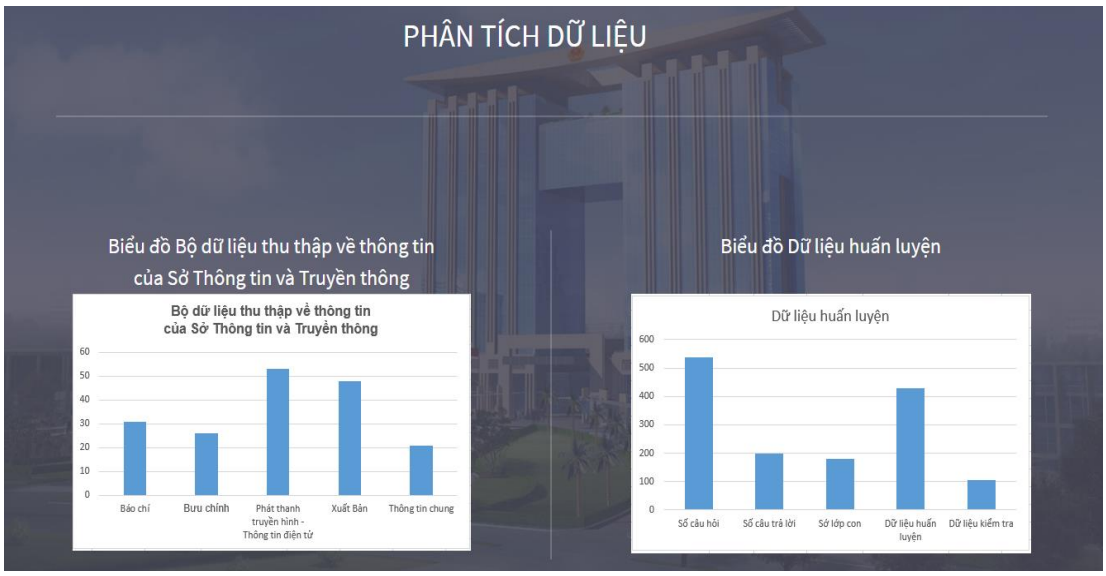
Hình 4.5: Giao diện Web - Chọn lựa mục để hỏi

Người dùng nhập các câu hỏi để nhận câu trả lời.



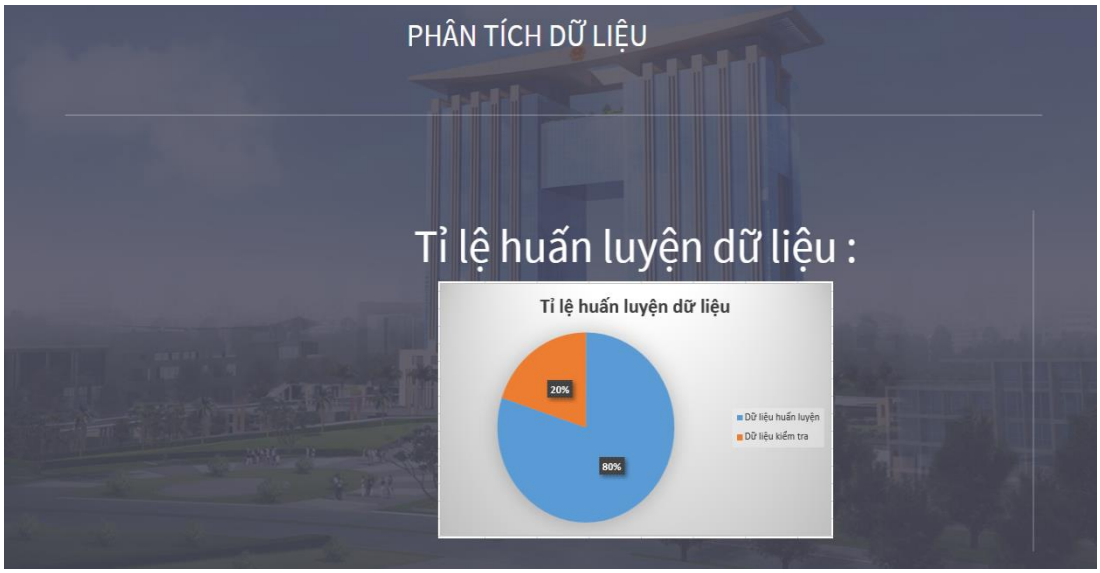
Hình 4.6: Giao diện Web - Hỏi và trả lời tự động

Người dùng chọn chức năng Phân tích dữ liệu ở màn hình chính của ứng dụng để xem biểu đồ về thông tin bộ dữ liệu thu thập.



Hình 4.7: Giao diện phân tích dữ liệu

Chức năng phân tích dữ liệu của ứng dụng cũng thể hiện tỉ lệ giữa bộ dữ liệu huấn luyện và bộ dữ liệu kiểm tra.

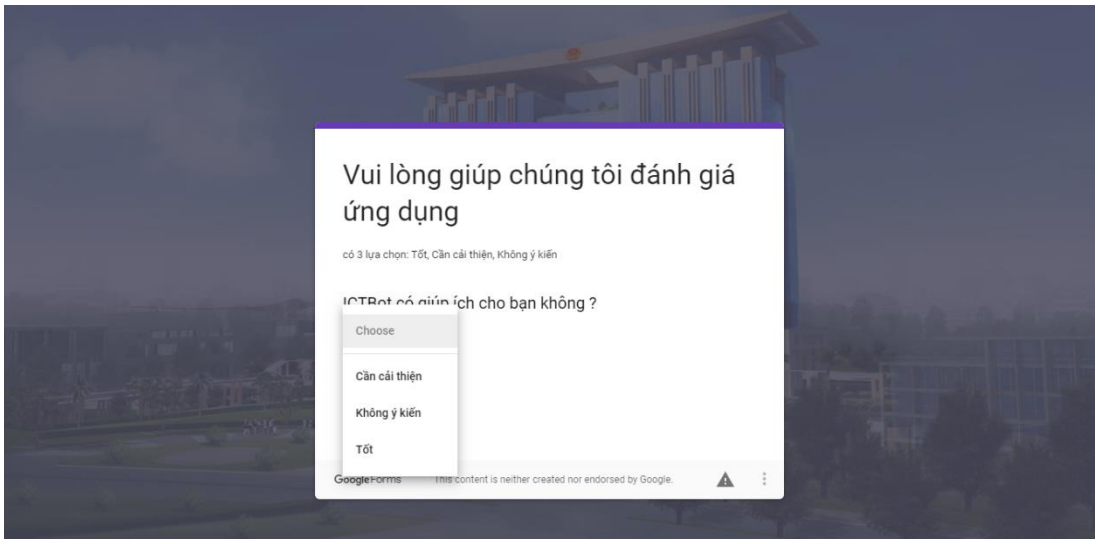


Hình 4.8: Giao diện phân tích tỉ lệ huấn luyện dữ liệu

Người dùng chọn chức năng Đánh giá kết quả ở màn hình chính của ứng dụng để xem đánh giá về mô hình đề xuất.

Hình 4.9: Giao diện kết quả đánh giá mô hình

Trong thời gian thử nghiệm chúng tôi sử dụng Google Biểu mẫu để thu thập đánh giá của người dùng về ứng dụng ICTBot, để đảm bảo tính khách quan chúng tôi yêu cầu người dùng đăng nhập tài khoản thư điện tử của Google để đánh giá ứng dụng và giới hạn mỗi tài khoản chỉ được phép đánh giá một lần.



Hình 4.10: Giao diện đánh giá ứng dụng

Chức năng đánh giá kết quả cũng thể hiện đánh giá ứng dụng từ người dùng trong thời gian ứng dụng được thử nghiệm tại Sở Thông tin và Truyền thông.

Hình 4.11: Kết quả phản hồi của người dùng

CHƯƠNG 5

KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

5.1. Kết quả đạt được

Hệ thống trả lời tự động là một trong những hướng phát triển nhằm hỗ trợ con người trong việc giảm tải nguồn nhân lực để chăm sóc khách hàng hoặc tư vấn dịch vụ cụ thể nào đó. Trong bối cảnh hiện nay, máy học là một trong các hướng tiếp cận để xây dựng hiệu quả này. Do đó, luận văn đã kế thừa những kết quả các nghiên cứu trước để xây dựng hệ thống trả lời tự động nhằm giúp Sở Thông tin và Truyền thông trả lời các câu hỏi của người dân về các thông tin liên quan đến những thủ tục hành chính do Sở phụ trách. Kết quả đã đạt của luận văn gồm:

+ Xây dựng hệ thống trả lời tự động dựa trên mô hình phân loại câu hỏi theo hướng mạng nơ ron Bộ nhớ dài ngắn song song BiLSTM: Bộ dữ liệu đầu vào là các câu hỏi và câu trả lời được phân lớp sau khi loại bỏ các từ dừng sẽ được tách từ bằng phương pháp Word2vector. Quá trình huấn luyện được tiến hành dựa trên kỹ thuật mạng nơ ron sâu thông qua hàm softmax để thể hiện xác suất của lớp và Entropy chéo được định nghĩa để đánh giá mục tiêu của đầu ra để dự đoán các câu hỏi được đưa vào của người sử dụng. Phương pháp đánh giá dựa trên độ đo chính xác được sử dụng trong mô hình này nhằm đánh giá kết quả để đưa ra mô hình dự đoán tối ưu.

+ Xây dựng ứng dụng dựa trên nền tảng Web-based: Luận văn cũng đã xây dựng một giao diện dựa trên nền tảng Web-based nhằm trực quan kết quả trả lời tự động các câu hỏi của người dân liên quan đến các thủ tục hành chính và các văn bản thường gặp của Sở Thông tin và Truyền thông tỉnh Bình Dương.

Với độ đo chính xác (Accuracy) đã giải quyết mặt hạn chế do dữ liệu thu thập đầu vào không được phong phú, số lượng câu hỏi ít nên sự khác biệt giữa các mô hình huấn luyện cho ra độ chính xác không chênh lệch nhiều. Mô hình BiLSTM tốn nhiều thời gian huấn luyện hơn nhưng cho kết quả tốt hơn mô hình LSTM.

5.2. Hướng phát triển

Tiếp tục kế thừa những nghiên cứu trước đây và phát triển mô hình chatbot mới có khả năng trả lời sát với ngữ cảnh, nhằm làm cho hệ thống trả lời tự động của mình đạt chất lượng tốt hơn. Tiếp tục xây dựng bộ dữ liệu liên quan đến Sở Thông

tin và Truyền thông nhiều hơn đặc biệt là các câu hỏi có cùng ngữ nghĩa, các câu hỏi theo văn nói mà người dùng có thể đặt câu hỏi cho chương trình liên quan các từ khóa thuộc chức năng nhiệm vụ của Sở Thông tin và Truyền thông.

Áp dụng các phương pháp học sâu khác để cải thiện độ chính xác của chương trình được cao hơn.

Mở rộng mô hình chatbot trong các lĩnh vực khác, thu thập bộ dữ liệu tối ưu nhất nhằm gia tăng tốc độ huấn luyện và tăng độ chính xác cho câu trả lời. Phát triển chương trình có thể áp dụng cho nhiều lĩnh vực khác nhằm phục vụ cho tất cả các Sở, ngành trong tỉnh Bình Dương.

Xây dựng một hệ thống tự động thu thập câu hỏi từ người dùng và có khả năng tự động cập nhật thông tin vào bộ dữ liệu đã có.

CÔNG TRÌNH CÔNG BỐ

Nguyễn Trung Tín, Bùi Thanh Hùng. (2019) **"Xây dựng hệ thống trả lời tự động áp dụng ở Trung tâm Công nghệ Thông tin và Truyền thông tỉnh Bình Dương"**. Kỷ yếu Ngày hội Khoa học Cán bộ, Giảng viên trẻ và Học viên cao học lần thứ III – năm 2019. Đại học Thủ Dầu Một. 6.2019

TÀI LIỆU THAM KHẢO

[1] L. Vergeest, "Using N-grams and Word Embeddings for Twitter Hashtag Suggestion", 2014, Tilburg University (School of Humanities).

[2] https://cs224d.stanford.edu/lecture_notes/notes1.pdf

[3] https://en.wikipedia.org/wiki/Hopfield_network

[4] S. Hochreiter and J. Schmidhuber, "Long Short-Term Memory", Neural Computation, vol. 9, pp. 1735–1780, 1997.

[5] Nhữ Bảo Vũ, "Xây dựng mô hình đối thoại cho tiếng việt trên miền mở dựa vào phương pháp học chuỗi liên tiếp", đại học quốc gia Hà Nội, trường Đại học Công Nghệ 2016.

[6] Zichao Yang, Diyi Yang, Chris Dyer, Xiaodong He, Alex Smola, and Eduard Hovy, "Hierarchical attention networks for document classification", In *Proc ACL,* 2016

[7] Wang P, Qian Y, Soong F K, He L, Zhao H, "Part-of-Speech Tagging with Bidirectional Long Short-Term Memory Recurrent Neural Network", Cornell University, 2015

[8] Andreas, J., Rohrbach, M., Darrell, T., and Klein, Deep Learning to Compose Neural Networks for Question Answering. arXiv preprint arXiv:1601.01705. 2016.

[9] Jinfeng Rao, Hua He, and Jimmy Lin. Noise-contrastive estimation for answer selection with deep neural networks. In Proceedings of the 25th ACM International on Conference on Information and Knowledge Management, CIKM '16, pages 1913–1916, New York, NY, USA. ACM. 2016.

[10] Nal Kalchbrenner, Edward Grefenstette and Phil Blunsom. "A convolutional neural network for modelling sentences". In Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (ACL-14), pages 655-665. 2014.

[11] Tom Young, Devamanyu Hazarika, Soujanya Poria, Erik Cambria. "Recent Trends in Deep Learning Based Natural Language Processing". IEEE Computational Intelligence Magazine, 2018.

[12] Tom Young, Devamanyu Hazarika, Soujanya Poria, Erik Cambria, "Recent Trends in Deep Learning Based Natural Language Processing, IEEE Computational Intelligence Magazine, 2018.

[13] Yoon Kim. Convolutional neural networks for sentence classification. CoRR, abs/1408.5882. 2014.

[14] Nguyen Van-Tu and Le Anh-Cuong. Improving question classification by feature extraction and selection. *Indian Journal of Science and Technology*, 9(17). 2016

[15] Nguyễn Thị Thanh Hương, "Xây dựng hệ thống trả lời tự động Chatbot bằng tiếng Việt sử dụng phương pháp học sâu", đại học Thủ Dầu Một, 2019.

[16] Bui Thanh Hung. (2019). "Vietnamese Question Classification based on Deep Learning for Educational Support System". The 19th International Symposium on Communications and Information Technologies, ISCIT 9.2019.

[17] Bui Thanh Hung. (2019). "Integrating Diacritics Restoration and Question Classification into Vietnamese Question Answering System". Special Issue on Advancement in Engineering and Computer Science Journal - ASTESJ, Volumn 4, Issue 5, Page No 207-212,, October 2019. ISSN: 2415-6698

[18] Cơ sở dữ liệu Quốc gia về thủ tục hành chính: http://csdl.thutuchanhchinh.vn/Pages/trang-chu.aspx

[19] Quyết định số 1284/QĐ-UBND ngày 17/5/2019 về việc công bố thủ tục hành chính thuộc thẩm quyền giải quyết của Sở Thông tin và Truyền thông/Ủy ban nhân dân cấp huyện trên địa bàn tỉnh Bình Dương.