ĐẠI HỌC QUỐC GIA HÀ NỘI

TRƯỜNG ĐẠI HỌC CÔNG NGHỆ

NHỮ BẢO VŨ

XÂY DỰNG MÔ HÌNH ĐỐI THOẠI CHO TIẾNG VIỆT TRÊN MIỀN MỞ DỰA VÀO PHƯƠNG PHÁP HỌC CHUỖI LIÊN TIẾP

Ngành: Công nghệ thông tin

Chuyên ngành: Hệ thống thông tin

Mã số: 60480104

TÓM TẮT LUẬN VĂN THẠC SĨ CÔNG NGHỆ THÔNG TIN

NGƯỜI HƯỚNG DẪN KHOA HỌC: TS. Nguyễn Văn Nam

HÀ NỘI – 2016

LỜI CAM ĐOAN

Tôi là Nhữ Bảo Vũ, học viên khóa K21, ngành Công nghệ thông tin, chuyên ngành Hệ Thống Thông Tin. Tôi xin cam đoan luận văn "Xây dựng mô hình đối thoại cho tiếng Việt trên miền mở dựa vào phương pháp học chuỗi liên tiếp" là do tôi nghiên cứu, tìm hiểu và phát triển dưới sự hướng dẫn của TS. Nguyễn Văn Nam. Luận văn 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.

Hà Nội, ngày tháng năm 2016

MỤC LỤC

DANH MỤC KÝ HIỆU VÀ CÁC CHỮ VIẾT TẮT

Từ viết tắt	Từ chuẩn	Diễn giải
NLP	Natural Languague Processing	Xử lý ngôn ngữ tự nhiên
ANN	Artificial Nerual Network	Mạng nơ ron nhân tạo
RNN	Recurrent Neural Network	Mạng nơ ron tái phát
CNN	Convolutional Neural Networks	Mạng nơ ron tích chập
LSTM	Long short-term memory	Mạng cải tiến để giải quyết vấn đề phụ thuộc quá dài
VNTK	Vietnamese Languague Toolkit	Bộ công cụ xử lý ngôn ngữ tiếng Việt
NLTK	Natural Language Toolkit	Bộ công cụ xử lý ngôn ngữ tự nhiên bằng Python
Python	Python	Ngôn ngữ lập trình python
Nodejs	Nodejs	Nền tảng lập trình phía Server sử dụng ngôn ngữ lập trình javascript
SDK	Support Development Kit	Bộ công cụ hỗ trợ phát triển
CPU	Central Processing Unit	Bộ xử lý trung tâm
GPU	Graphics Processing Unit	Bộ vi xử lý chuyên dụng nhận nhiệm vụ tăng tốc, xử lý đồ họa cho bộ vi xử lý trung tâm CPU
API	Application Programming Interface	Giao diện lập trình ứng dụng
QA	Question Answering	Các cặp câu hỏi đáp
BLEU	Bilingual Evaluation Understudy	Thuật toán để đánh giá chất lượng của một văn bản được sinh ra từ một mô hình ngôn ngữ tự nhiên

DANH MỤC HÌNH VẼ VÀ ĐỒ THỊ

TÓM TẮT

Trong bối cảnh mạng xã hội đã trở lên rất phổ biến như hiện nay, con người kết nối với con người thông qua mạng xã hội, bất cứ thời gian nào và ở bất cứ nơi đâu. Sẽ thật tốt hơn nếu có một hệ thống tự động thông minh hỗ trợ con người bằng cách trò chuyện, có khả năng nhắc nhở, làm trợ lý công việc và có thể theo dõi tình trạng sức khỏe cá nhân mọi lúc, mọi nơi.

Mô hình hóa đối thoại là một nhiệm vụ quan trọng trong bài toán hiểu ngôn ngữ tự nhiên, và máy học thông minh. Các phương pháp tiếp cận trước đây thường giới hạn trong một lĩnh vực cụ thể, ví dụ như đặt vé trực tuyến, tư vấn ghi danh trực tuyến, tìm kiếm thông tin y tế, … và yêu cầu phải thiết kế được các bộ luật học bằng tay, mất nhiều công sức mà hiệu quả đạt được không cao, khó mở rộng mô hình và các ứng dụng có liên quan.

Trong đề tài này, chúng tôi sẽ nghiên cứu, xây dựng một mô hình đối thoại cho tiếng Việt, dựa trên phương pháp học chuỗi liên tiếp, sequence-to-sequence, để sinh ra câu trả lời từ một chuỗi đầu vào tương ứng. Lợi thế của phương pháp này là mô hình có thể được huấn luyện end-to-end trên tập dữ liệu có sẵn, và yêu cầu ít hơn các luật bằng tay. Kết quả chính của chúng tôi đạt được một mô hình đối thoại sử dụng các mạng học sâu để sinh ra câu trả lời bằng tiếng Việt, tương ứng với một câu hỏi chuỗi đầu vào. Mô hình ban đầu đã cho kết quả rất tính cực, có thể giải quyết được những vấn đề cơ bản về ngữ nghĩa, ngữ cảnh và tính cách riêng trong hệ thống đối thoại.

CHƯƠNG 1: TỔNG QUAN VỀ HỆ THỐNG TRẢ LỜI TỰ ĐỘNG

1.1 Động lực nghiên cứu và tính cấp thiết của bài toán thực tế

Khái niệm Trợ lý ảo, Chatbot, hay Hệ thống trả lời tự động đang là chủ đề rất nóng từ đầu năm nay 2016, khi chính thức các công ty lớn như Microsoft (Cortana), Google (Google Assistant), Facebook (M), Apple (Siri), Samsung (Viv), WeChat, Slack đã giới thiệu các trợ lý ảo của mình, là các hệ thống trả lời tự động. Chính thức đặt cược lớn vào cuộc chơi chatbot, với mong muốn tạo ra một trợ lý ảo thực sự thông minh tồn tại trong hệ sinh thái các sản phẩm của mình.

Tình hình trong nước, một số công ty như Hồ sơ y tế điện tử ERM.,JSC và Vietcare đã phát triển tạo ra hệ thống trả lời tự động về kiến thức y khoa, hỏi đáp về sức khỏe thông tin y tế, hay RiveHub, Subiz cũng đang cố gắng tạo ra cho mình một hệ thống hỗ trợ, chăm sóc khách hàng và bán hàng tự động. Nhằm trợ giúp người dùng, khách hàng của mình có những trải nghiệm tốt nhất về sản phẩm và cách dịch vụ cung cấp.

1.2 Tình hình nghiên cứu trong và ngoài nước

Hệ thống trả lời tự động đã được các nhà nghiên cứu quan tâm từ rất lâu rồi, 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, … như TREC, CLEF, tại Việt Nam có KSE, RIVF, ATC, …

Với sự ra đời của framework sequence-to-sequence [7], nhiều hệ thống huấn luyện gần đây đã sử dụng các mạng nơ-ron tái phát (RNN) để sinh ra các câu trả lời mới khi đưa vào mạng một câu hỏi hoặc một thông điệp.. Với sự giúp đỡ của các mô hình ngôn ngữ được tiền huấn luyện, chúng mã hóa mỗi tin nhắn vào một vector đại diện. Để loại bỏ sự cần thiết cho một mô hình ngôn ngữ, Serban và cộng sự (2015) [3] đã thử huấn luyện end-to-end trên một mạng RNN. Họ cũng bắt đầu hệ thống của mình với các word embeddings đã được huấn luyện từ trước.

1.3 Phân loại các mô hình trả lời tự động

Mô hình trả lời tự động dựa vào một số kỹ thuật và các tiêu chí khác nhau, như:

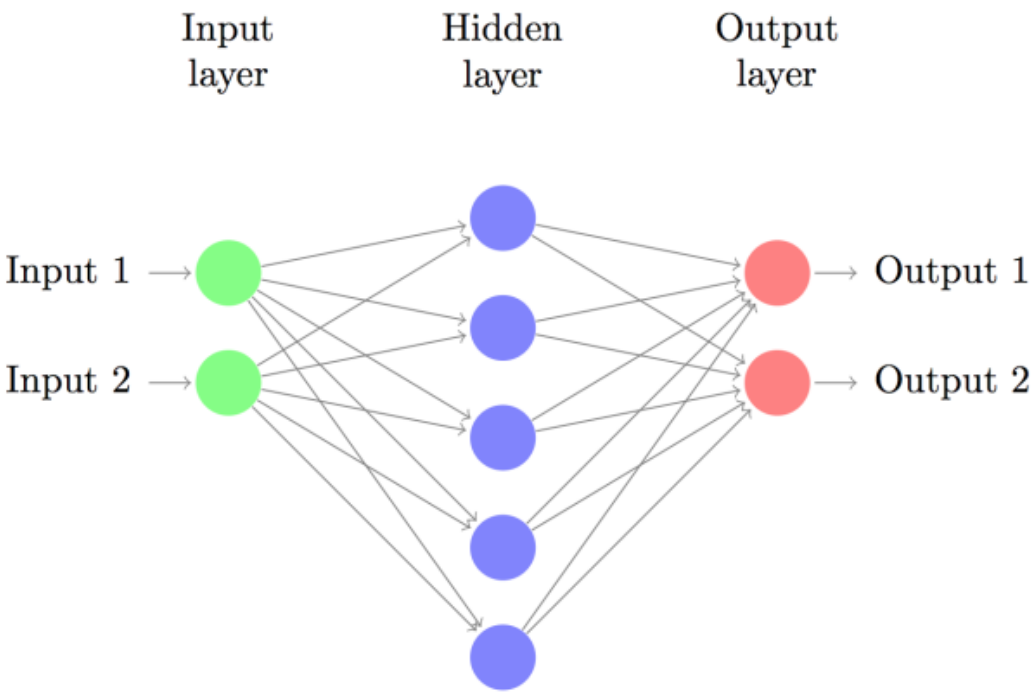
- Phân loại theo miền ứng dụng
- Phân loại theo khả năng trả lời mẫu hỏi
- Phân loại theo mức độ dài, ngắn của đoạn đối thoại
- Phân loại theo hướng tiếp cận

CHƯƠNG 2: CƠ SỞ MẠNG NƠ RON NHÂN TẠO

Chương này giới thiệu về cơ sở lý thuyết về mạng nơ rơn nhân tạo là cơ sở thực hiện xây dựng mô hình đối thoại trong luận văn.

2.1 Kiến trúc mạng nơ ron nhân tạo

Mạng nơ ron nhân tạo (Artificial Neural Network – ANN) là một mô hình xử lý thông tin được mô phỏng dựa trên hoạt động của hệ thống thần kinh của sinh vật, bao gồm số lượng lớn các Nơ-ron được gắn kết để xử lý thông tin. ANN hoạt động giống như bộ não của con người, được học bởi kinh nghiệm (thông qua việc huấn luyện), có khả năng lưu giữ các tri thức và sử dụng các tri thức đó trong việc dự đoán các dữ liệu chưa biết (unseen data).



Hình 2.1: Kiến trúc mạng nơ-ron nhân tạo

Kiến trúc chung của một ANN gồm 3 thành phần đó là **Input Layer**, **Hidden Layer** và **Output Layer** (Xem hình trên)

2.3 Mạng nơ-ron tái phát và ứng dụng

Mạng nơ-ron tái phát **Recurrent Neural Network** (RNN) là một trong những mô hình Deep learning được đánh giá có nhiều ưu điểm trong các tác vụ xử lý ngôn ngữ tự nhiên (NLP). Trong phần này, tôi sẽ trình bày các khái niệm, các đặc điểm cũng như những ứng dụng của RNNs trong các bài toán thực tế.

2.3.1 Mạng nơ-ron tái phát

Ý tưởng của RNNs đó là thiết kế một Neural Network sao cho có khả năng xử lý được thông tin dạng chuỗi *(sequential information)*, ví dụ một câu là một chuỗi gồm nhiều từ.

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.

Nói cách khác, 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 đó là thông tin đầu vào (input) hoàn toàn độc lập với thông tin đầu ra (output). Về lý thuyết, RNNs 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.

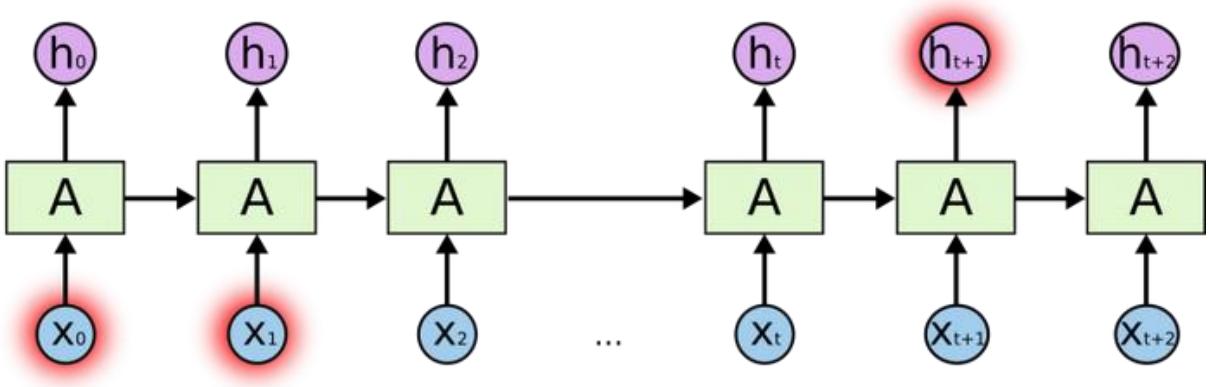
2.3.2 Các ứng dụng của mạng RNN

- **Mô hình ngôn ngữ và phát sinh văn bản (Generating text)**
- **Dịch máy (Machine Translation)**
- **Phát sinh mô tả cho ảnh (Generating Image Descriptions)**

2.4 Mạng Long Short Term Memory (LSTM)

2.4.1 Vấn đề phụ thuộc quá dài

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, trong mô hình hóa ngôn ngữ, chúng ta cố gắng dự đoán từ tiếp theo dựa vào các từ trước đó. Nếu chúng ta dự đoán từ cuối cùng trong câu "đám_mây bay trên bầu_trời", thì chúng ta không cần truy tìm quá nhiều từ trước đó, ta có thể đoán ngay từ tiếp theo sẽ là "bầu_trời".



Hình 2.2: RNN phụ thuộc long-term.

Về lý thuyết, RNN hoàn toàn có khả năng xử lý "long-term dependencies" [14], nghĩa là thông tin hiện tại có được là nhờ vào chuỗi thông tin trước đó. Đáng buồn là, trong thực tế, RNN dường như không có khả năng này. Vấn đề này đã được Hochreiter (1991) [German] and Bengio, et al. (1994) đưa ra như một thách thức cho mô hình RNN.

Rất may là chúng ta đã có mạng LSTM giải quyết được vấn đề này!

CHƯƠNG 3: MÔ HÌNH ĐỐI THOẠI VỚI MẠNG NƠ-RON

3.1 Hệ thống đối thoại người máy

Các hệ thống đối thoại người máy (Dialogue systems), còn được gọi là trợ lý tương tác hội thoại, trợ lý ảo và đôi khi được gọi với thuật ngữ là chatbot, được sử dụng rộng rãi trong các ứng dụng khác nhau, từ các dịch vụ kỹ thuật cho đến các công cụ có thể học ngôn ngữ và giải trí [17]. Các hệ thống đối thoại có thể được chia thành *các hệ thống hướng mục tiêu*, ví dụ như các dịch vụ hỗ trợ kỹ thuật, và *các hệ thống không có định hướng mục tiêu*, ví dụ như các công cụ học ngôn ngữ hoặc các nhân vật trò chơi máy tính [3]. Trong luận văn này, chúng tôi tập trung vào trường hợp thứ hai, là đi xây dựng một mô hình đối thoại cho tiếng Việt trên miền mở do có sẵn nguồn dữ liệu lớn từ các phụ đề Phim tiếng Việt được lấy trên OpenSubtitles năm 2016 [1].

3.2 Mô hình ngôn ngữ

Nền tảng của việc xây dựng mô hình chuỗi tuần tự (ví dụ, mô hình dịch máy) là mô hình ngôn ngữ. Ở mức cao, một mô hình ngôn ngữ đón nhận chuỗi các phần tử đầu vào, nhìn vào từng phần tử của chuỗi và cố gắng để dự đoán các phần tử tiếp theo của chuỗi văn bản. Có thể mô tả quá trình này bằng phương trình hàm số sau đây:

$$Y_t = f(Y_{t-1})$$

Trong đó, Y(t) là phần tử chuỗi ở thời điểm t, Y(t-1) là phần tử chuỗi ở thời điểm trước đó (t – 1), và *f* là hàm ánh xạ các phần tử trước đó của chuỗi sang phần tử tiếp theo của chuỗi. Bởi vì chúng ta đang đề cập đến mô hình chuỗi sử dụng mạng nơ-ron, *f* đại diện cho mạng nơ-ron mà có thể dự đoán được phần tử tiếp theo của một chuỗi, được cho trước bởi một phần tử hiện tại trong chuỗi đó.

Không giống với các mô hình ngôn ngữ đơn giản là chỉ dự đoán xác suất cho từ tiếp theo khi được cho bởi từ hiện tại, mô hình RNN chụp lại toàn bộ bối cảnh của chuỗi đầu vào. Đo đó, RNN dự đoán xác suất tạo ra các từ tiếp theo dựa trên các từ hiện tại, cũng như tất cả các từ trước.

3.3 Mô hình chuỗi liên tiếp seq2seq

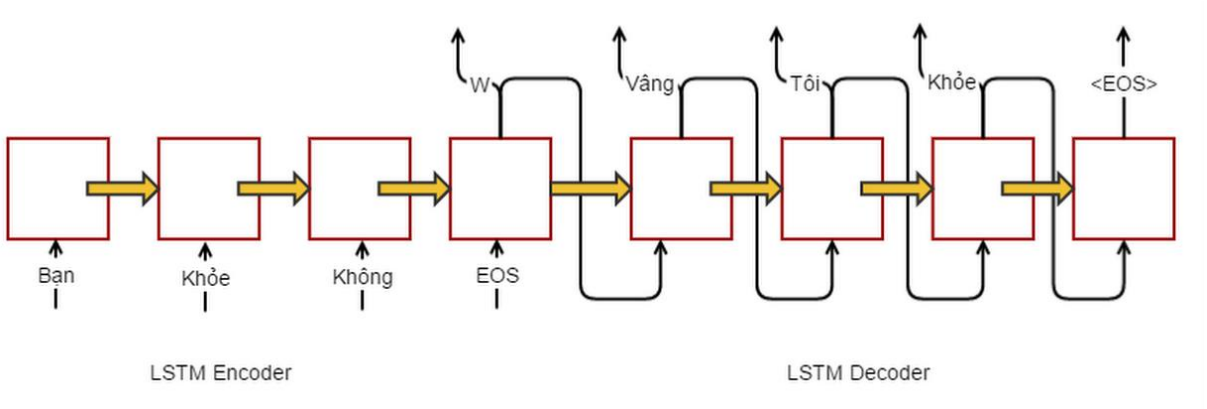
RNN có thể được sử dụng như là mô hình ngôn ngữ cho việc dự đoán các phần tử của một chuỗi khi cho bởi các phần tử trước đó của một chuỗi. Tuy nhiên, chúng ta vẫn còn thiếu các thành phần cần thiết cho việc xây dựng các mô hình đối thoại, hay các mô hình máy dịch, bởi vì chúng ta chỉ có thể thao tác trên một chuỗi đơn, trong khi việc dịch hoạt động trên cả hai chuỗi – chuỗi đầu vào và chuỗi được dịch sang.

Một mô hình ngôn ngữ đơn giản cho phép chúng ta mô hình hóa các chuỗi đơn giản bằng việc dự đoán tiếp theo trong một chuỗi khi cho một từ trước đó trong chuỗi. Thêm nữa là chúng ta đã thấy quá trình xây dựng một mô hình phức tạp có phân tách các bước như mã hóa một chuỗi đầu vào thành một bối cảnh, và sinh một chuỗi đầu ra bằng việc sử dụng một mạng nơ-ron tách biệt.

Mô hình chuỗi sang chuỗi Seq2seq, [5] được giới thiệu trong bài báo "Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation", kể từ đó đã trở thành mô hình cho các hệ thống đối thoại (Dialogue Systems) và Máy dịch (Machine Translation).

3.4 Mô hình đối thoại Seq2seq

Bản thân mô hình seq2seq nó bao gồm hai mạng RNN: Một cho bộ mã hóa, và một cho bộ giải mã. Bộ mã hóa nhận một chuỗi (câu) đầu vào và xử lý một phần tử (từ trong câu) tại mỗi bước. Mục tiêu của nó là chuyển đổi một chuỗi các phần tử vào một vectơ đặc trưng có kích thước cố định mà nó chỉ mã hóa thông tin quan trọng trong chuỗi và bỏ qua các thông tin không cần thiết. Có thể hình dung luồng dữ liệu trong bộ mã hóa dọc theo trục thời gian, giống như dòng chảy thông tin cục bộ từ một phần tử kết thúc của chuỗi sang chuỗi khác.



Hình 3.1: Mô hình đối thoại seq2seq.

Mỗi trạng thái ẩn ảnh hưởng đến trạng thái ẩn tiếp theo và trạng thái ẩn cuối cùng được xem như tích lũy tóm tắt về chuỗi. Trạng thái này được gọi là bối cảnh hay vector suy diễn, vì nó đại diện cho ý định của chuỗi. Từ bối cảnh đó, các bộ giải mã tạo ra một chuỗi, một phần tử (word) tại một thời điểm. Ở đây, tại mỗi bước, các bộ giải mã bị ảnh hưởng bởi bối cảnh và các phần tử được sinh ra trước đó.

Có một vài thách thức trong việc sử dụng mô hình này. Một trong những vấn đề đáng ngại nhất là các mô hình không thể xử lý được các chuỗi dài. Bởi vì hầu như tất cả các ứng dụng chuỗi sang chuỗi, bao gồm cả độ dài các chuỗi. Vấn đề tiếp theo là kích thước từ vựng. Bộ giải mã phải chạy hàm softmax hơn trên một tập rất lớn các từ vựng (khoảng 20,000 từ) cho mỗi một từ xuất ra. Điều này sẽ làm chậm quá trình huấn luyện, cho dù phần cứng của bạn có thể đáp ứng được khả năng xử lý. Đại diện của một từ là rất quan trọng. Làm thế nào để có thể biểu diễn được các từ trong chuỗi ? Sử dụng one-hot vector có nghĩa là chúng ta phải đối mặt với các vector thưa thớt lớn, do kích thước vốn từ vựng lớn mà không có ý nghĩa về mặt ngữ nghĩa của từ được mã hóa bên trong các vector one-hot. Sau đây là cách giải quyết một số vấn đề mà chúng ta sẽ gặp phải.

- **PADDING – Tạo độ dài cố định**
- **BUCKETING – Tránh lu mờ thông tin**
- **Word Embedding – Mật độ dày đặc**

3.5 Những thách thức chung khi xây dựng mô hình đối thoại

Có một số thách thức thể hiện một cách rõ ràng hoặc không thể thấy rõ khi xây dựng một mô hình đối thoại nói chung đang là tâm điểm được chú ý bởi nhiều nhà nghiên cứu.

3.5.1 Phụ thuộc bối cảnh

Để sinh ra các câu trả lời hợp lý, các hệ thống đối thoại cần phải kết hợp với cả hai bối cảnh ngôn ngữ và bối cảnh vật lý. Trong các hội thoại dài, người nói cần theo dõi và nhớ được những gì đã được nói và nhưng thông tin gì đã được trao đổi. Đó là một ví dụ về bối cảnh ngôn ngữ. Phương pháp tiếp cận phổ biến nhất là nhúng cuộc hội thoại vào một Vector, nhưng việc làm này đối với đoạn hội thoại dài là một thách thức lớn. Các thử nghiệm trong các nghiên cứu [3], [15] đều đi theo hướng này. Hướng nghiên cứu này cần kết hợp các loại bối cảnh như: Ngày/giờ, địa điểm, hoặc thông tin về một người.

3.5.2 Kết hợp tính cách

Khi phát sinh các câu trả lời, các hệ thống trợ lý ảo lý tưởng là tạo ra câu trả lời phù hợp với ngữ nghĩa đầu vào cần nhất quán giống nhau. Ví dụ, chúng ta muốn nhận được câu trả lời với mẫu hỏi "Bạn bao nhiêu tuổi?" hay "Tuổi của bạn là mấy?". Điều này nghe có vẻ đơn giản, nhưng việc tổng hợp, tích hợp các kiến thức nhất quán hay "có tính cách" vào trong các mô hình đối thoại là một vấn đề rất khó để nghiên cứu.

message	Where do you live now?
response	I live in Los Angeles.
message	In which city do you live now?
response	I live in Madrid.
message	In which country do you live now?
response	England, you?

Hình 3.2: Thách thức phụ thuộc bối cảnh và tính cách khi xây dựng mô hình đối thoại.

Rất nhiều các hệ thống được huấn luyện để trả lời câu hỏi thỏa đáng với ngôn ngữ, nhưng chúng không được huấn luyện để sinh ra các câu trả lời nhất quán về ngữ nghĩa. Mô hình như thế đang được nghiên cứu trong [10], tạo ra những bước đầu tiên tập trung vào hướng mô hình hóa tính cách.

CHƯƠNG 4: THỰC NGHIỆM XÂY DỰNG MÔ HÌNH ĐỐI THOẠI CHO TIẾNG VIỆT

Chương này tiến hành thực nghiệm xây dựng mô hình đối thoại cho tiếng Viết bằng việc áp dụng mô hình đối thoại Seq2seq trên miền mở.

4.1 Dữ liệu và công cụ thực nghiệm

Chúng tôi thử nghiệm bộ dữ liệu trên miền mở sử dụng bộ dữ liệu phụ đề phim tiếng Việt được lấy từ nguồn mở OpenSubtitles 2016 [1].

Đây là phiên bản sạch được công bố năm 2016, đã được cải thiện các hội thoại, gióng câu, kiểm tra song ngữ, và các siêu dữ liệu khác, gồm:

- 60 ngôn ngữ, 1,689 bitexts
- Tổng số file: 2,815,754
- Tổng số tokens: 17.18G
- Tổng số câu: 2.60G
- Trang chủ: http://www.opensubtitles.org/vi
- Download: http://opus.lingfil.uu.se/OpenSubtitles2016.php

Sau khi tiền xử lý dữ liệu, chúng tôi thu thập được bộ dữ liệu bao gồm 2,078,696 câu văn bản tiếng Việt. Các công đoạn làm sạch xử lý dữ liệu, chúng tôi đã thực hiện qua các bước sau:

- Loại bỏ các ký tự đặc biệt không phải chữ hoặc chữ số (bắt đầu, kết thúc và bên trong một câu tiếng Việt), **ex**: - *Xin chào, các bạn!*, ...
- Xóa bỏ các ký tự phân tách câu không phải dấu chấm, dấu hỏi hoặc dấu chấm than, **ex**: *@#$%^&**, ...
- Xóa bỏ các bình luận, chú thích ý nghĩa các từ, thuật ngữ trong câu, **ex**: *Chatbot (chương trình tự động trả lời)*, ...
- Xóa bỏ các ký tự lặp, ký tự phân tách không có ý nghĩa, **ex**: -,, ..., ...
- Xóa bỏ các thẻ html, **ex**: *Khi mặt trời lói dạng*, ...
- Biến đổi bảng mã html về dạng câu có ý nghĩa, **ex**: *Cho ch#250;ng t#244;i xem c#225;i c#242;n l#7841;i l#224; g#236; n#224;o.*
- Biến đổi bảng mã Unicode tổ hợp về dạng unicode dựng sẵn, **ex**:

```
Phaʾi. Vaˋ noˊ se˜ đuʿoʾ.c thuʿˋa nha.ˆn boʿʾi nhuʿ˜ng gia điˋnh coˊ danh tieˆˊn
```

- Loại bỏ các cặp câu không có ý nghĩa, ex: *Phụ_đề dịch bởi Unknow Subteam 2pi, ...*

Công cụ sử dụng:

- **NLTK**: Công cụ xử lý ngôn ngữ tự nhiên mã nguồn mở.
- **VNTK**: Vietnamese languague toolkit, do chúng tôi xây dựng và phát triển để xử lý các vấn đề cơ bản của tiếng Việt.
- **Subsent**: Công cụ hỗ trợ bóc tách dữ liệu từ các file phụ đề, do chúng tôi xây dựng và phát triển
- **Dongdu**: Thư viện hỗ trợ tách từ tiếng Việt [11], của tác giả Lưu Tuấn Anh
- **Tensorflow**: Framework machine learning
- **Python**: Ngôn ngữ lập trình để xây dựng mô hình đối thoại tiếng Việt.

4.2 Tách từ tập dữ liệu tiếng Việt

Tách từ là một quá trình xử lý nhằm mục đích xác định ranh giới của các từ trong câu văn, cũng có thể hiểu đơn giản rằng tách từ là quá trình xác định các từ đơn, từ ghép… có trong câu. Đối với xử lý ngôn ngữ, để có thể xác định cấu trúc ngữ pháp của câu, xác định từ loại của một từ trong câu, yêu cầu nhất thiết đặt ra là phải xác định được đâu là từ trong câu. Vấn đề này tưởng chừng đơn giản với con người nhưng đối với máy tính, đây là bài toán rất khó giải quyết.

Chính vì lý do đó tách từ được xem là bước xử lý quan trọng đối với các hệ thống Xử Lý Ngôn Ngữ Tự Nhiên, đặc biệt là đối với các ngôn ngữ thuộc vùng Đông Á theo loại hình ngôn ngữ đơn lập, ví dụ: tiếng Trung Quốc, tiếng Nhật, tiếng Thái, và tiếng Việt. Với các ngôn ngữ thuộc loại hình này, ranh giới từ không chỉ đơn giản là những khoảng trắng như trong các ngôn ngữ thuộc loại hình hòa kết như tiếng Anh…, mà có sự liên hệ chặt chẽ giữa các tiếng với nhau, một từ có thể cấu tạo bởi một hoặc nhiều tiếng. Vì vậy đối với các ngôn ngữ thuộc vùng Đông Á, vấn đề của bài toán tách từ là khử được sự nhập nhằng trong ranh giới từ.

Bởi vì các lý do trên, trước khi đưa vào mô hình huấn luyện và trả lời câu hỏi chúng tôi đã thực hiện tách từ tiếng Việt và sử dụng công cụ **DongDu** của tác giả Lưu Tuấn Anh trong [11] với độ chính xác tới 98% tính theo từ.

Kết quả sau khi chúng tôi làm sạch dữ liệu, tiền xử lý, tách từ, lựa chọn tập dữ liệu học chúng tôi thu được bộ dữ liệu, như sau:

- 120,885 words
- 362,655 tokens
- 1,824,063 QA

4.3 Thực nghiệm xây dựng mô hình đối thoại tiếng Việt

Do dữ liệu thu thập được khá lớn, chúng tôi đã chia làm 4 tập nhỏ mỗi tập 500,000 câu đối thoại phụ đề phim tiếng Việt. Và thử nghiệm trên 1 tập dữ liệu với các bộ tham số khác nhau, gồm:

Profile 1: • 26,746 words • 74,870 QA • 5 Max length • Thời gian huấn luyện: 25 giờ • RAM: 4GB • CPU: 2 Core x4 @ 2.4GHz	Profile 2: • 37,693 words • 180,700 QA • 7 Max length • Thời gian huấn luyện: 125 giờ • RAM: 4GB • CPU: 2 Core x4 @ 2.4GHz
Profile 3: • 40,348 words • 320,419 QA • 10 Max length • Thời gian huấn luyện: 270 giờ • RAM: 4GB • CPU: 2 Core x4 @ 2.4GHz	Profile 4: • 43,033 words • 381,474 QA • 12 Max length • Thời gian huấn luyện: 1,000 giờ • RAM: 4GB • CPU: 2 Core x4 @ 2.4GHz

Trong các lần thử nghiệm này chúng tôi đã huấn luyện mô hình mạng sử dụng hai lớp mạng LSTM (kích thước lớp ẩn 256) và kích thước embeding là 25.

Kết quả khi sử dụng huấn luyện với **profile 1**, tức là mẫu hỏi / câu trả lời có độ dài tối đa là 5. Và mạng này cho kết quả khá tốt !

Q: Xin chào **A: chào**	**Q: Tạm_biệt** **A: tạm_biệt**

Q: Xin chào ! **A: xin chào !** **Q: Chào** **A: chào** **Q: Chào bạn** **A: này ward !** **Q: Anh đấy à ?** **A: ừ** **Q: Ồ vậy à** **A: được mà** **Q: Thật không ?** **A: thật chứ** **Q: Có tin được không ?** **A: uh** **Q: Ai vậy ?** **A: anh st .** **Q: Anh đang ở đâu ?** **A: ở tư_gia**	**Q: Tạm_biệt anh** **A: tạm_biệt** **Q: Tạm_biệt cô** **A: tạm_biệt các em** **Q: Tên tôi là Vũ** **A: chào mừng** **Q: Gặp anh sau nhé** **A: tạm_biệt** **Q: Gặp anh sau nhé !** **A: chúc may_mắn !** **Q: Cá có_thể bơi không ?** **A: không** **Q: Anh bao_nhiêu tuổi ?** **A: 20** **Q: Anh ấy bao_nhiêu tuổi ?** **A: 19** **Q: Cô ấy bao_nhiêu tuổi ?** **A: 19**

Một số trường hợp fail, kết quả chưa được tốt đối với profile 1.

Q: Hôm_nay là tháng mấy **A: okay** **Q: Anh là ai vậy** **A: em ổn rồi** **Q: Anh vẫn khỏe chứ ?** **A: chào anh gray** **Q: Con chó đâu ?** **A: thanh_tra montgomery** **Q: Tên anh là gì ?** **A: jonesy_._jonesy** **Q: Anh còn sống không ?** **A: kyle , tương_lai**	**Q: Ai là John ?** **A: tao làm_việc giải_thích** **Q: 2000** **A: miriam !** **Q: Tình_yêu là cái gì ?** **A: qua nhân_sự thùng** **Q: Cô ghét ai ?** **A: bài bông** **Q: Chim có_thể bay không ?** **A: quan_tâm đang buồn_cười** **Q: Ai có_thể hát được ?** **A: không có họ**

Các thử nghiệm với câu dài hơn so với input huấn luyện.

Kết quả khi sử dụng huấn luyện với **profile 3**, tức là mẫu hỏi / câu trả lời có độ dài tối đa là 10 với dữ liệu 40,348 words và 320,419 QA. Đúng như mong

đợi, kết quả cho lần huấn luyện này đã đạt được rất tốt về mặt ngữ nghĩa. Các câu trả lời đã sát nghĩa hơn, bám sát với ý của câu hỏi, cùng với cặp hội thoại được đưa vào kiểm nghiệm với kết quả như sau:

Human: Xin chào **Machine: chào** **Human: Xin chào !** **Machine: xin chào !** **Human: Chào** **Machine: chào** **Human: Chào bạn** **Machine: chào** **Human: Anh đấy à ?** **Machine: yeah** **Human: Ồ vậy à** **Machine: phải** **Human: Thật không ?** **Machine: thật** **Human: Có tin được không ?** **Machine: không** **Human: Ai vậy ?** **Machine: tôi không biết** **Human: Anh đang ở đâu ?** **Machine: tôi không biết**	**Human: Tạm_biệt** **Machine: tạm_biệt** **Human: Tạm_biệt anh** **Machine: tạm_biệt** **Human: Tạm_biệt cô** **Machine: tạm_biệt** **Human: Tên tôi là Vũ** **Machine: tôi là gwen** **Human: Gặp anh sau nhé** **Machine: tạm_biệt** **Human: Gặp anh sau nhé !** **Machine: chúc may_mắn !** **Human: Cá có_thể bơi không ?** **Machine: tôi không biết** **Human: Anh bao_nhiêu tuổi ?** **Machine: tôi không biết** **Human: Anh ấy bao_nhiêu tuổi ?** **Machine: 19** **Human: Cô ấy bao_nhiêu tuổi ?** **Machine: 19**

Một số kết quả fail trong **profile 1**, đã có chất lượng được cải thiện đáng kể như bảng dưới đây, thay vì là phớt lờ thông tin thì nó đã trả lời một cách thẳng thắn hơn bằng câu „Tôi không biết“.

Human: Hôm_nay là tháng mấy **Machine: tôi không biết** **Human: Anh là ai vậy** **Machine: anh không biết** **Human: Anh vẫn khỏe chứ ?** **Machine: ừ**	**Human: Ai là John ?** **Machine: tôi không biết** **Human: 2000** **Machine: khoản của tôi** **Human: Tình_yêu là cái gì ?** **Machine: cháu không biết**

KẾT LUẬN

Luận văn này đã đưa ra các lý thuyết và vấn đề trong quá trình thiết lập, huấn luyện và xây dựng một hệ thống đối thoại cho tiếng Việt trên miền mở. Từ đó, đã xây dựng được mô hình đối thoại tự động cho tiếng Việt trên miền dữ liệu mở được lấy từ kho phụ đề mã mở OpenSubtitles2016 [1]. Kết quả ban đầu đạt được là tiền đề để tạo ra các trợ lý ảo, xây dựng các ứng dụng thông minh có thể hiểu được ngôn ngữ tiếng Việt. Có khả năng áp dụng vào các bài toán thực tế, ví dụ như các hệ thống hỗ trợ hỏi đáp về y khoa, tư vấn mua hàng, hỗ trợ giải đáp kỹ thuật cho khách hàng, các dịch vụ khác, … Đặc biệt, có thể tạo ra một trợ lý ảo mà có thể theo dõi sức khỏe và tương tác với cá nhân mà chúng tôi đang hướng tới.

Từ kết quả thực nghiệm của luận văn này, chúng tôi có một số nhận xét: Với các chuỗi câu dài thì mạng huấn luyện mất nhiều thời gian hơn. Sau khoảng 300,000 lần lặp với độ dài 10 từ thì mạng vẫn cung cấp những câu trả lời lảng tránh, phớt lờ câu hỏi (bằng việc trả lời bằng các câu **“Tôi không biết”**, nhưng nó đã hiểu và cần tích hợp một số ngữ nghĩa cơ bản. Bằng việc thay đổi mô hình bằng cách điều chỉnh độ dài của mạng hoặc tối ưu cục bộ các cặp câu hỏi-đáp thì cho kết quả với chất lượng tốt hơn rất nhiều, bám sát ngữ nghĩa hơn.

Qua những kết quả đạt được ban đầu, chúng nhận thấy còn rất nhiều việc phải làm, cần phải tối ưu. Nhưng cách tiếp cận này ban đầu đã cho những kết quả rất tích cực và đúng đắn, có thể giải quyết được những vấn đề ngữ nghĩa, ngữ cảnh và tính cách trong hệ thống đối thoại. Định hướng nghiên cứu tiếp theo, chúng tôi tiếp tục làm mượt dữ liệu, để tạo ra các mô hình mới có khả năng trả lời sát với ngữ cảnh, đạt chất lượng cao hơn, giảm khả năng lảng tránh và đưa tính cá nhân vào trong đoạn hội thoại.

TÀI LIỆU THAM KHẢO

1. Pierre Lison and Jörg Tiedemann, 2016, *OpenSubtitles2016: Extracting Large Parallel Corpora from Movie and TV Subtitles.* In Proceedings of the 10th International Conference on Language Resources and Evaluation (LREC 2016)
2. Ryan Lowe, Nissan Pow, Iulian Serban, Joelle Pineau, 4 Feb 2016. "The Ubuntu Dialogue Corpus: A Large Dataset for Research in Unstructured Multi-Turn Dialogue Systems".
3. Iulian V. Serban, Alessandro Sordoni, Yoshua Bengio, Aaron Courville, Joelle Pineau, 6 Apr 2016. "Building End-To-End Dialogue Systems Using Generative Hierarchical Neural Network Models".
4. Wojciech Zaremba, Ilya Sutskever, Oriol Vinyals, 19 Feb 2015. "Recurrent Neural Network Regularization".
5. Kyunghyun Cho, Bart van Merrienboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, Yoshua Bengio, Sep 2014. "Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation".
6. Oriol Vinyals, Quoc Le, 22 Jul 2015. "A Neural Conversational Model".
7. Ilya Sutskever, Oriol Vinyals, Quoc V. Le, 14 Dec 2014. "Sequence to Sequence Learning with Neural Networks" pp. 1–9.
8. Lifeng Shang, Zhengdong Lu, Hang Li, 27 Apr 2015. "Neural Responding Machine for Short-Text Conversation".
9. Alessandro Sordoni, Michel Galley, Michael Auli, Chris Brockett, Yangfeng Ji, Margaret Mitchell, Jian-Yun Nie, Jianfeng Gao, Bill Dolan, 22 Jun 2015. "A Neural Network Approach to Context-Sensitive Generation of Conversational Responses".
10. Jiwei Li, Michel Galley, Chris Brockett, Georgios P. Spithourakis, Jianfeng Gao, Bill Dolan, 8 Jun 2016. "A Persona-Based Neural Conversation Model".
11. Lưu Tuấn Anh, Yamamoto Kazuhide, 16 Feb 2013. "Pointwise for Vietnamese Word Segmentation".
12. S. Hochreiter and J. Schmidhuber, 1997. "Long Short-Term Memory" Neural Computation, vol. 9, pp. 1735–1780.
13. S. Sukhbaatar, A. Szlam, J. Weston, and R. Fergus, 31 Mar 2015. "End-To-End Memory Networks" pp. 1–11.
14. Christopher Olah, 27 Aug 2015. "Understanding LSTM Networks".

15. Kaisheng Yao, Geoffrey Zweig, Baolin Peng, 29 Oct 2015. “Attention with Intention for a Neural Network Conversation Model”.
16. Jacob Andreas, Marcus Rohrbach, Trevor Darrell, Dan Klein, 7 Jan 2016. “Learning to Compose Neural Networks for Question Answering”.
17. Young, M. Gasic, B. Thomson, and J. D. Williams, 2013. “POMDP-based statistical spoken dialog systems: A review. Proceedings of the IEEE”, 101(5):1160–1179.
18. Williams, A. Raux, D. Ramachandran, and A. Black. The dialog state tracking challenge. In Special Interest Group on Discourse and Dialogue (SIGDIAL), 2013.
19. S. Kim, L. F. DHaro, R. E. Banchs, J. Williams, and M. Henderson. Dialog state tracking challenge 4. 2015.
20. Wen, M. Gasic, D. Kim, N. Mrksic, P. Su, D. Vandyke, and S. Young. Stochastic language generation in dialogue using recurrent neural networks with convolutional sentence reranking. Special Interest Group on Discourse and Dialogue (SIGDIAL), 2015.