TỔNG LIÊN ĐOÀN LAO ĐỘNG VIỆT NAM

**TRƯỜNG ĐẠI HỌC TÔN ĐỨC THẮNG**

**KHOA CÔNG NGHỆ THÔNG TIN**

****

**MAI HOÀNG MINH - 51703130**

**NGUYỄN VĂN TINH - 51703199**

**XÂY DỰNG HỆ THỐNG CHATBOT DỰA TRÊN PHƯƠNG PHÁP TRUY VẤN THÔNG TIN**

**(RETRIEVAL - BASED)**

**DỰ ÁN CÔNG NGHỆ THÔNG TIN 2**

**HỆ THỐNG THÔNG TIN**

**THÀNH PHỐ HỒ CHÍ MINH, NĂM 2022**

TỔNG LIÊN ĐOÀN LAO ĐỘNG VIỆT NAM

**TRƯỜNG ĐẠI HỌC TÔN ĐỨC THẮNG**

**KHOA CÔNG NGHỆ THÔNG TIN**

****

**MAI HOÀNG MINH - 51703130**

**NGUYỄN VĂN TINH - 51703199**

**XÂY DỰNG HỆ THỐNG CHATBOT DỰA TRÊN PHƯƠNG PHÁP TRUY VẤN THÔNG TIN**

**(RETRIEVAL - BASED)**

**DỰ ÁN CÔNG NGHỆ THÔNG TIN 2**

**HỆ THỐNG THÔNG TIN**

Người hướng dẫn

**TRẦN LƯƠNG QUỐC ĐẠI**

**THÀNH PHỐ HỒ CHÍ MINH, NĂM 2022**

**LỜI CẢM ƠN**

Để hoàn thành bộ môn dự án CNTT 2 lần này chúng tôi xin gửi lời cảm ơn tới Thầy Trần Lương Quốc Đại đã tận tình hướng dẫn giúp đỡ, đóng góp ý kiến và chỉ bảo nhiệt tình trong quá trình thực hiện đề tài.

Đồng thời, chúng tôi xin chân thành bày tỏ lòng biết ơn đến toàn thể quý thầy cô giáo trong trường đã tận tình hỗ trợ tôi trong suốt quá trình làm bài. Nhờ vào các kiến thức, đặc biệt là kiến thức chuyên môn, được truyền dạy lại sẽ là cơ sở lý thuyết vững vàng và tạo điều kiện giúp đỡ tôi trong suốt quá trình học tập và làm việc hiện tại cũng như sau này.

Chúng tôi đã cố gắng hoàn thành đồ án một cách tốt nhất nhưng sẽ có những thiếu sót mà chúng tôi không nhận thấy. Chúng tôi mong nhận được những phản hồi và ý kiến của các thầy cô để bài đồ án sau chúng tôi có thể hoàn thành tốt hơn, chúng tôi xin chân thành cảm ơn.

**ĐỒ ÁN ĐƯỢC HOÀN THÀNH**

**TẠI TRƯỜNG ĐẠI HỌC TÔN ĐỨC THẮNG**

Tôi xin cam đoan đây là sản phẩm đồ án của riêng chúng tôi và được sự hướng dẫn của Thầy Trần Lương Quốc Đại. Các nội dung nghiên cứu, kết quả trong đề tài này là trung thực và chưa công bố dưới bất kỳ hình thức nào trước đây. Những số liệu trong các bảng biểu phục vụ cho việc phân tích, nhận xét, đánh giá được chính tác giả thu thập từ các nguồn khác nhau có ghi rõ trong phần tài liệu tham khảo.

Ngoài ra, trong đồ án còn sử dụng một số nhận xét, đánh giá cũng như số liệu của các tác giả khác, cơ quan tổ chức khác đều có trích dẫn và chú thích nguồn gốc.

**Nếu phát hiện có bất kỳ sự gian lận nào tôi xin hoàn toàn chịu trách nhiệm về nội dung đồ án của mình.** Trường đại học Tôn Đức Thắng không liên quan đến những vi phạm tác quyền, bản quyền do tôi gây ra trong quá trình thực hiện (nếu có).

*TP. Hồ Chí Minh, ngày tháng năm*

*Tác giả*

*(ký tên và ghi rõ họ tên)*

*Mai Hoàng Minh Nguyễn Văn Tinh*

**PHẦN XÁC NHẬN VÀ ĐÁNH GIÁ CỦA GIẢNG VIÊN**

**Phần xác nhận của GV hướng dẫn**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Tp. Hồ Chí Minh, ngày tháng năm

(kí và ghi họ tên)

**Phần đánh giá của GV chấm bài**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Tp. Hồ Chí Minh, ngày tháng năm

(kí và ghi họ tên)

**TÓM TẮT**

Chatbot là một chương trình kết hợp với trí tuệ nhân tạo (AI) để tương tác với con người, Ngày nay, hơn 30% công việc sử dụng công nghệ Chatbot. Các công ty ngày càng sử dụng nhiều hơn những con bot này vào các công việc như tư vấn khách hàng, tạo nguồn thông tin,...Ngoài ra chatbot còn giúp chúng ta tiết kiệm được nguồn nhân lực và phần lớn thời gian để có thể đầu tư vào những hoạt động khác.

Chatbot được phân thành 2 loại chính: Retrieval-Based vs Generative Models. Trong bài báo cáo này sẽ tập trung tìm hiểu về một Retrieval-Based Chatbot.

Báo cáo được hoàn thành trong 3 tháng với các nội dung từ tiếp cận đề tài qua các cơ sở lý thuyết cho đến các thực nghiệm thực tiễn. Nội dung báo cáo sẽ gồm 5 chương:

Chương 1: Phân tích bài toán

Chương 2: Cơ sở lý thuyết

Chương 3: Mô hình chatbot bằng phương pháp truy vấn thông tin

Chương 4: Thực nghiệm

Chương 5: Kết luận

**MỤC LỤC**

[CHƯƠNG 1: GIỚI THIỆU 5](#_heading=h.28h4qwu)

[1.1 Giới thiệu về đề tài 5](#_heading=h.3l18frh)

[1.2 Phân tích yêu cầu bài toán 5](#_heading=h.206ipza)

[1.3 Hướng giải quyết 6](#_heading=h.4k668n3)

[1.4 Các hướng tiếp cận 6](#_heading=h.2zbgiuw)

[1.4.1 Rule-based 7](#_heading=h.1egqt2p)

[1.4.2 Retrieval-Based 7](#_heading=h.3ygebqi)

[1.4.3 Generative model 8](#_heading=h.2dlolyb)

[CHƯƠNG 2:CƠ SỞ LÝ THUYẾT 9](#_heading=h.sqyw64)

[2.1 Bags of Word 9](#_heading=h.3cqmetx)

[2.2 Model ANN 9](#_heading=h.1rvwp1q)

[2.2.1. Giới thiệu ANN. 9](#_heading=h.4bvk7pj)

[2.2.2. Kiến trúc mạng nơ ron nhân tạo. 10](#_heading=h.2r0uhxc)

[2.2.3 Quá trình xử lý thông tin của một ANN. 11](#_heading=h.3q5sasy)

[2.2.4 Ưu điểm và nhược điểm của mạng nơ ron nhân tạo. 13](#_heading=h.25b2l0r)

[2.3 Model LSTM 14](#_heading=h.kgcv8k)

[2.3.1 Giới thiệu 14](#_heading=h.34g0dwd)

[2.3.2 Kiến trúc mô hình LSTM 14](#_heading=h.1jlao46)

[CHƯƠNG 3:MÔ HÌNH CHATBOT BẰNG PHƯƠNG PHÁP TRUY VẤN THÔNG TIN 18](#_heading=h.2iq8gzs)

[3.1 Xây dựng ngữ liệu 18](#_heading=h.xvir7l)

[3.1.1.Cấu trúc ngữ liệu 18](#_heading=h.3hv69ve)

[3.1.2.Nguồn dữ liệu 19](#_heading=h.4h042r0)

[3.1.3. Thống kê dữ liệu 19](#_heading=h.2w5ecyt)

[3.2 Mô hình Retrieval-based Chatbot 21](#_heading=h.3vac5uf)

[3.2.1 Tiền xử lý dữ liệu 21](#_heading=h.pkwqa1)

[3.2.1.1 Tách từ tiếng việt 22](#_heading=h.39kk8xu)

[3.2.1.2 Lowercase và remove unnecessary characters 24](#_heading=h.48pi1tg)

[3.2.1.3 Loại bỏ stopword 24](#_heading=h.2nusc19)

[3.2.2 Trích xuất đặc trưng 25](#_heading=h.1302m92)

[3.2.3 Module phân lớp 27](#_heading=h.3mzq4wv)

[3.2.4 Tạo phản hồi 30](#_heading=h.haapch)

[CHƯƠNG 4: THỰC NGHIỆM 31](#_heading=h.319y80a)

[4.1 Môi trường thực nghiệm 31](#_heading=h.1gf8i83)

[4.2 Tập dữ liệu 31](#_heading=h.40ew0vw)

[4.3 Tham số chạy mô hình 32](#_heading=h.2fk6b3p)

[4.4 Tiêu chí đánh giá mô hình 33](#_heading=h.1tuee74)

[4.4.1 Phân lớp 33](#_heading=h.4du1wux)

[4.4.2 Độ đo BLEU 33](#_heading=h.2szc72q)

[4.5 Kết quả thực nghiệm. 35](#_heading=h.3s49zyc)

[CHƯƠNG 5 KẾT LUẬN VÀ ĐÁNH GIÁ 39](#_heading=h.meukdy)

[5.1 Kết luận 39](#_heading=h.36ei31r)

[5.2 Đánh giá 39](#_heading=h.1ljsd9k)

**DANH MỤC KÍ HIỆU VÀ CHỮ VIẾT TẮT**

**CÁC CHỮ VIẾT TẮT**

NLP : Natural Language Processing

BOW : Bag Of Words

SGD : Stochastic gradient descent

BLEU : Bilingual Evaluation Understudy

IR : Information Retrieval

ANN : Artificial Neural Network

LSTM : Long Short Term Memory

**DANH MỤC CÁC BẢNG BIỂU, HÌNH VẼ, ĐỒ THỊ**

**DANH MỤC BẢNG**

[Bảng 1 Thống kê dữ liệu 20](#_heading=h.1baon6m)

[Bảng 2 Tham số chạy mô hình ANN 31](#_heading=h.upglbi)

[Bảng 3 Tham số chạy mô hình LSTM 32](#_heading=h.3ep43zb)

[Bảng 4 : Độ đo BLEU của mô hình 34](#_heading=h.184mhaj)

**DANH MỤC HÌNH**

[Hình 1 Kiến trúc mạng nơ ron nhân tạo 10](#_heading=h.1664s55)

[Hình 2 : Kiến trúc mô hình LSTM 15](#_heading=h.43ky6rz)

[Hình 3 Cấu trúc ngữ liệu 19](#_heading=h.1x0gk37)

[Hình 4 Mô hình Retrieval-based Chatbot 21](#_heading=h.2afmg28)

[Hình 5 Ví dụ bài toán tách từ 23](#_heading=h.1opuj5n)

[Hình 6 Sequential Model of Keras 27](#_heading=h.2250f4o)

**CHƯƠNG 1: GIỚI THIỆU**

**1.1 Giới thiệu về đề tài**

Chatbot là một chương trình máy tính được xây dựng để có thể trò chuyện với con người. Một chat-bot đơn giản có thể dùng để thay con người trả lời các câu hỏi lặp đi lặp lại của người dùng như: “Sự kiện A diễn ra khi nào?”, “iPhone X giá bao nhiêu?”,…

Chatbot thường được mô tả là một trong những biểu hiện tương tác tiên tiến và hứa hẹn nhất giữa con người và máy móc. Tuy nhiên, từ quan điểm công nghệ, một chatbot chỉ đại diện cho sự tiến hóa tự nhiên của hệ thống Trả lời câu hỏi sử dụng Xử lý ngôn ngữ tự nhiên (NLP). Xây dựng câu trả lời cho câu hỏi bằng ngôn ngữ tự nhiên là một trong những ví dụ điển hình nhất về Xử lý ngôn ngữ tự nhiên được áp dụng trong các ứng dụng cuối cùng của các doanh nghiệp.

Để thành công, một hệ thống chatbot phải có thể thực hiện hiệu quả cả hai nhiệm vụ này:

* Phân tích yêu cầu người dùng
* Phản hồi

**1.2 Phân tích yêu cầu bài toán**

Mục tiêu của đề tài là xây dựng chatbot có khả năng đối thoại tiếng việt bằng phương pháp truy vấn thông tin (Information retrieval). Cụ thể trong đồ án này sẽ xây dựng một chatbot tư vấn pháp luật cơ bản

Đôi khi, người dùng cần tham khảo những thông tin pháp luật cơ bản trước khi cần sự tư vấn 1 – 1 từ luật sư chuyên nghiệp. Đối với những trường hợp như vậy, chatbot rất hữu ích vì có thể đáp ứng các truy vấn cơ bản của người dùng với kiến ​​thức chuyên sâu, chính xác và dễ sử dụng. Chỉ cần tạo kịch bản hỏi – đáp với từ khoá tương ứng vào các chủ đề luật pháp thường gặp, sau đó Chatbot sẽ có thể tự động trả lời ngay trong thời gian thực khi người dùng nhắn tin vào hệ thống.

**1.3 Hướng giải quyết**

Trong đồ án này sẽ tập trung xây dựng Chatbot dựa trên retrieval-based với bộ dữ liệu luật pháp lý cơ bản đã được định nghĩa

Retrieval-based model có một kho định nghĩa trước các câu trả lời chúng có thể sử dụng, không giống như generative models cái mà có thể generate câu trả lời chúng không bao giờ nhìn thấy trước đây. Từ một câu đầu vào tới retrieval-based model là một context sẽ chọn ra một câu trả lời response có khả năng nhất. Ouputs của model là câu trả lời tốt nhất. Để tìm ra câu trả lời tốt nhất cần phải tính toán điểm cho tất cả các câu trả lời và chọn ra một với số điểm cao nhất.

Nhưng tại sao lại tạo một retrieval-based model nếu có thể tạo một generative model? Generative models dường như linh hoạt hơn vì chúng không cần kho câu trả lời định nghĩa trước, vấn đề là generative models không hoạt động tốt trong thực tế. Bởi vì có quá nhiều tự do cho việc chúng có thể trả lời như thế nào, generative models có thể tạo ra các lỗi cú pháp, không có nghĩa, không nhất quán. Chúng cũng cần một số lượng dữ liệu training lớn và khó để tối ưu. Đa số hệ thống ngày nay là retrieval-based, hoặc kết hợp giữa retrieval-based và generative. Nếu tạo một chatbot ngày nay, lựa chọn tốt nhất là tạo một retrieval-based model.

**1.4 Các hướng tiếp cận**

Chatbot là một hệ thống dựa vào các kỹ thuật về xử lý ngôn ngữ tự nhiên, vậy nên giống với hầu hết các bài toán về xử lý ngôn ngữ khác, các hệ thống Chatbot có thể được chia làm 3 loại:

Rule-based: Hệ thống Chatbot dựa vào các quy luật, thói quen trong ngôn ngữ của người dùng.

Retrieval-based: Hệ thống Chatbot xây dựng trên một kho dữ liệu hội thoại cho trước – Kho văn bản này có thể được thu thập bằng lượng lớn dữ liệu từ các cuộc nói chuyện của người dùng, sử dụng các phương pháp trích xuất thông tin (Information Retrieval) hoặc các phương pháp máy học để tạo ra câu trả lời dựa vào ngữ cảnh trò chuyện với người dùng.

Generation-based: Một chương trình chatbot miền mở tạo ra các kết hợp ngôn ngữ ban đầu thay vì chọn từ các câu trả lời được xác định trước. Các mô hình seq2seq được sử dụng để dịch máy có thể được sử dụng để xây dựng các chatbot chung.

***1.4.1 Rule-based***

Hiểu một các đơn giản, Rule-based chatbot sẽ trả lời người dùng dựa hoàn toàn vào các thói quen sử dụng ngôn ngữ của chúng ta mà không cần xử lý việc ghi nhớ thông tin trước đó. Tuy rằng trong ngôn ngữ nói hàng ngày của chúng ta, mỗi người đều sẽ có những cách diễn đạt, cách sử dụng từ ngữ riêng để tạo ra các câu hội thoại của bản thân, nhưng thói quen sử dụng ngôn ngữ con người có xu hướng lặp lại khá nhiều.

Hệ thống Chatbot thành công và nổi tiếng nhất thuộc loại này là *ELIZA* Chatbot*.* ELIZA được tạo ra từ năm 1966, và được coi như một bước tiến quan trọng trong lịch sử của Chatbot nói chung và trí tuệ nhân tạo nói riêng

***1.4.2 Retrieval-Based***

Một hệ thống dựa trên truy xuất đặc biệt tập trung vào một miền cụ thể. Ở đây chúng tôi sử dụng một miền đóng của truy vấn hỏi đáp pháp luật được sử dụng để thực hiện. Các bot dựa trên truy xuất hoạt động bằng cách sử dụng nguyên tắc của các luồng hoặc đồ thị có hướng. Về cơ bản, loại bot như vậy được đào tạo để xếp hạng phản hồi tốt nhất từ ​​một tập hợp hữu hạn các phản hồi được xác định trước. Các câu trả lời ở đây được nhà phát triển nhập theo cách thủ công hoặc dựa trên nền tảng kiến ​​thức về thông tin đã có từ trước.

Chatbot dựa trên truy xuất sử dụng các kỹ thuật như đối sánh từ khóa, máy học hoặc học sâu để xác định phản hồi phù hợp nhất. Bất kể kỹ thuật nào, các chatbot này chỉ cung cấp các phản hồi được xác định trước và không tạo ra kết quả mới.

Một ví dụ về chatbot dựa trên truy xuất là Mitsuku. Chatbot chứa hơn 300.000 mẫu phản hồi được xác định trước và cơ sở kiến thức của hơn 3000 đối tượng. Chatbot này có thể tạo các bài hát và bài thơ dựa trên nền tảng kiến thức của chatbot.

***1.4.3 Generative model***

Các hệ thống dựa trên truy xuất được giới hạn trong các phản hồi được xác định trước. Chatbots sử dụng các phương pháp tổng hợp có thể tạo ra các cuộc đối thoại mới dựa trên một lượng lớn dữ liệu đào tạo đàm thoại.

Các chatbots tạo ra sử dụng kết hợp giữa học có giám sát, học không giám sát, học tăng cường và học đối đầu để đào tạo nhiều bước.

Học tập có giám sát cấu trúc một cuộc hội thoại như một vấn đề theo trình tự. Học theo trình tự liên kết ánh xạ đầu vào của người dùng với phản hồi do máy tính tạo ra. Tuy nhiên, kiểu học này có xu hướng ưu tiên những câu trả lời có xác suất cao như “Tôi biết”. Hệ thống học tập có giám sát cũng gặp khó khăn khi đưa danh từ riêng vào lời nói của chúng vì chúng ít xuất hiện hơn trong cuộc đối thoại so với các từ khác. Do đó, các chatbot học tập có giám sát nghe lặp đi lặp lại và không thể thúc đẩy cuộc trò chuyện ổn định của con người.

Để giải quyết vấn đề đó, các nhà phát triển tận dụng phương pháp học tăng cường để dạy chatbots cách tối ưu hóa các cuộc đối thoại để có được một số phần thưởng tích lũy.

**CHƯƠNG 2:CƠ SỞ LÝ THUYẾT**

**2.1 Bags of Word**

Bag of words là một kỹ thuật Xử lý ngôn ngữ tự nhiên của mô hình văn bản. Về mặt kỹ thuật, chúng ta có thể nói rằng nó là một phương pháp trích xuất đối tượng địa lý bằng dữ liệu văn bản. Cách tiếp cận này là một cách đơn giản và linh hoạt để trích xuất các tính năng từ tài liệu.

Một túi từ là một biểu diễn của văn bản mô tả sự xuất hiện của các từ trong một tài liệu. Chúng tôi chỉ theo dõi số lượng từ và bỏ qua các chi tiết ngữ pháp và thứ tự từ. Nó được gọi là “bag” các từ vì bất kỳ thông tin nào về thứ tự hoặc cấu trúc của các từ trong tài liệu đều bị loại bỏ. Mô hình chỉ quan tâm đến việc các từ đã biết có xuất hiện trong tài liệu hay không, chứ không phải ở vị trí nào trong tài liệu.

Một trong những vấn đề lớn nhất với văn bản là nó lộn xộn và không có cấu trúc, đồng thời các thuật toán học máy thích các đầu vào có độ dài cố định có cấu trúc, được xác định rõ ràng và bằng cách sử dụng kỹ thuật Bag-of-Words, chúng ta có thể chuyển đổi các văn bản có độ dài thay đổi thành độ dài cố định vectơ.

Ngoài ra, ở cấp độ chi tiết, các mô hình học máy hoạt động với dữ liệu số hơn là dữ liệu văn bản. Vì vậy, cụ thể hơn, bằng cách sử dụng kỹ thuật bag-of-words (BoW), chúng tôi chuyển một văn bản thành vectơ số tương đương của nó.

**2.2 Model ANN**

***2.2.1. Giới thiệu ANN.***

Mạng Neuron nhân tạo (Artificial Neural Network- ANN) là 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 Neuron được gắn kết để xử lý thông tin. ANN giống như bộ não con người, được học bởi kinh nghiệm (thông qua huấn luyện), có khả năng lưu giữ những kinh nghiệm hiểu biết (tri thức) và sử dụng những tri thức đó trong việc dự đoán các dữ liệu chưa biết (unseen data).

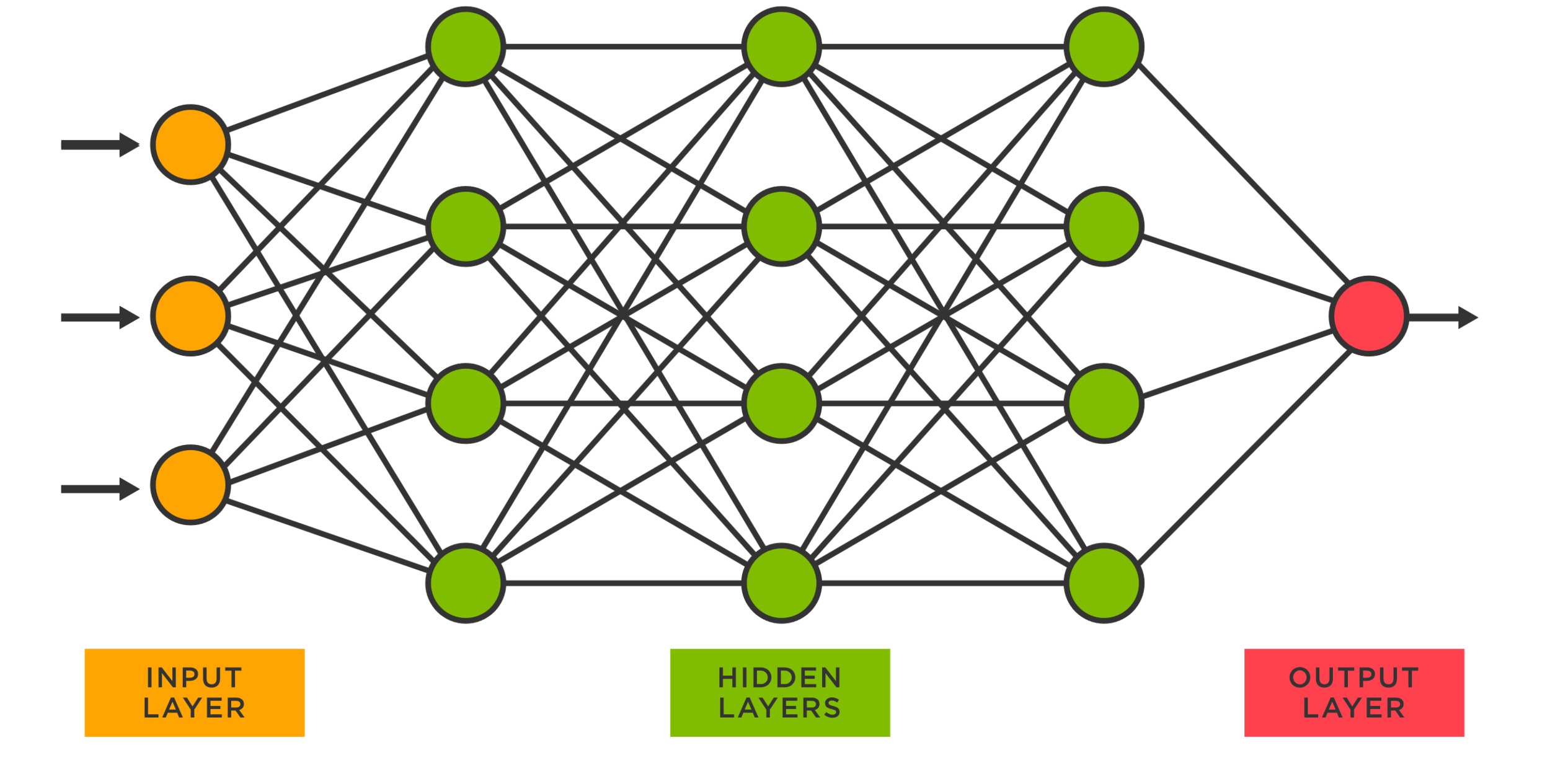
Các ứng dụng của mạng Neuron được sử dụng trong rất nhiều lĩnh vực như điện, điện tử, kinh tế, quân sự,… để giải quyết các bài toán có độ phức tạp và đòi hỏi có độ chính xác cao như điều khiển tự động, khai phá dữ liệu, nhận dạng,…

Ví dụ: Bài toán phân loại(classification), bài toán nhận dạng mẫu(pattern recognition), bài toán dự đoán(Prediction), bài toán điều khiển và tối ưu(optimization), bài toán lọc nhiễu(Noise filtering).

***2.2.2. Kiến trúc mạng nơ ron nhân tạo.***

Một mạng nơ ron có thể gồm 1 hoặc nhiều nơ ron, mỗi nơ ron là một đơn vị xử lý thông tin , hoạt động động song song và được liên kết với nhau.

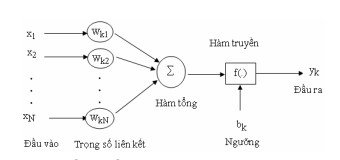
Cấu trúc đặc trưng của một mạng nơ ron gồm 3 tầng:Tầng vào(input layer), Tầng ẩn(hidden layer) ,Tầng ra(output layer).



Hình 1 Kiến trúc mạng nơ ron nhân tạo

* Tầng vào(input layer):Nhận các giá trị đầu vào cho mạng.
* Tầng ẩn(hidden layer): Là tập hợp các tế bào thần kinh giữa các lớp đầu vào và đầu ra, nó thể hiện cho quá trình suy luận logic của mạng.Có thể có một hoặc nhiều lớp.
* Tầng ra(output layer):Thể hiện cho những đầu ra của mạng.

***2.2.3 Quá trình xử lý thông tin của một ANN.***

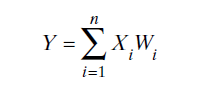


Inputs: Mỗi Input tương ứng với 1 thuộc tính (attribute) của dữ liệu (patterns). Ví dụ như trong ứng dụng của ngân hàng xem xét có chấp nhận cho khách hàng vay tiền hay không thì mỗi Input là một thuộc tính của khách hàng như thu nhập, nghề nghiệp, tuổi, số con,…

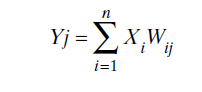
Output: Kết quả của một ANN là một giải pháp cho một vấn đề, ví dụ như với bài toán xem xét chấp nhận cho khách hàng vay tiền hay không thì output là yes (cho vay) hoặc no (không cho vay).

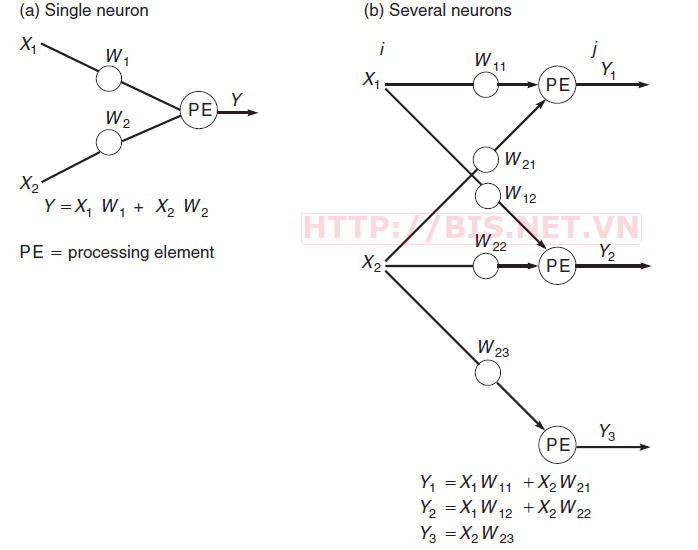
Connection Weights (Trọng số liên kết) : Đây là thành phần rất quan trọng của một ANN, nó thể hiện mức độ quan trọng (độ mạnh) của dữ liệu đầu vào đối với quá trình xử lý thông tin (quá trình chuyển đổi dữ liệu từ Layer này sang layer khác). Quá trình học (Learning Processing) của ANN thực ra là quá trình điều chỉnh các trọng số (Weight) của các input data để có được kết quả mong muốn.

Summation Function (Hàm tổng): Tính tổng trọng số của tất cả các input được đưa vào mỗi Neuron (phần tử xử lý PE). Hàm tổng của một Neuron đối với n input được tính theo công thức sau:



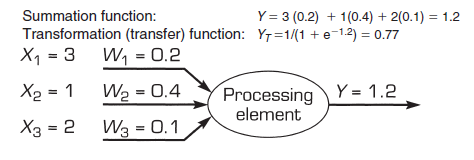
 Hàm tổng đối với nhiều Neurons trong cùng một Layer (Xem hình b):





 Transformation (Transfer) Function (Hàm chuyển đổi)

Hàm tổng (Summation Function) của một Neuron cho biết khả năng kích hoạt (Activation) của neuron đó còn gọi là kích hoạt bên trong (internal activation). Các Neuron này có thể sinh ra một output hoặc không trong ANN (nói cách khác rằng có thể output của 1 Neuron có thể được chuyển đến layer tiếp trong mạng Neuron theo hoặc không). Mối quan hệ giữa Internal Activation và kết quả (output) được thể hiện bằng hàm chuyển đổi (Transfer Function).



Việc lựa chọn Transfer Function có tác động lớn đến kết quả của ANN. Hàm chuyển đổi phi tuyến được sử dụng phổ biến trong ANN là sigmoid (logical activation) function.

http://bis.net.vn/photos/storage/20110612161354813.png

Trong đó :

YT: Hàm kích hoạt

Y: Hàm tổng

Kết quả của Sigmoid Function thuộc khoảng [0,1] nên còn gọi là hàm chuẩn hóa (Normalized Function).

Kết quả xử lý tại các Neuron (Output) đôi khi rất lớn, vì vậy transfer function được sử dụng để xử lý output này trước khi chuyển đến layer tiếp theo. Đôi khi thay vì sử dụng Transfer Function người ta sử dụng giá trị ngưỡng (Threshold value) để kiểm soát các output của các neuron tại một layer nào đó trước khi chuyển các output này đến các Layer tiếp theo. Nếu output của một neuron nào đó nhỏ hơn Threshold thì nó sẻ không được chuyển đến Layer tiếp theo.

***2.2.4 Ưu điểm và nhược điểm của mạng nơ ron nhân tạo.***

Ưu điểm:

* Hoạt động song song và theo thời gian thực.
* Tính dung sai hay khả năng chịu lỗi.
* Tính tự tổ chức.

Có thể tự thích nghi.

Nhược điểm:

* Không có quy tắc để xác định cấu trúc mạng.
* Không có phương pháp để đánh giá hoạt động bên trong ANN.
* Khó giải thích.
* Khó trong việc dự đoán hiệu năng của hệ thống trong tương lai.

**2.3 Model LSTM**

***2.3.1 Giới thiệu***

Mạng bộ nhớ ngắn hạn dài hạn là viết tắt của LSTM. Đây là một loại RNN được trang bị để học các điều kiện đường dài, đặc biệt là trong nhóm các vấn đề dự báo.

Bộ nhớ ngắn hạn dài có các kết nối đầu vào, tức là, nó phù hợp để xử lý toàn bộ nhóm dữ liệu, ngoài việc tập trung dữ liệu đơn lẻ như hình ảnh. Điều này khám phá ứng dụng trong dịch máy, nhận dạng giọng nói, v.v. Trí nhớ ngắn hạn dài là một loại RNN đặc biệt, cho thấy khả năng thực thi đặc biệt đối với một loạt các vấn đề.

Cấu trúc giống như chuỗi của bộ nhớ ngắn hạn dài cho phép nó chứa dữ liệu trong thời gian dài hơn, giải quyết các công việc sai sót trong thử nghiệm mà trận chiến mạng nơ-ron tái diễn thông thường quá hoặc về cơ bản không thể giải quyết.

Một số ứng dụng của LSTM như : Dự đoán cấu trúc bậc hai của protein, tổng hợp giọng nói, mô hình âm nhạc đa hình,chuyển đổi video thành văn bản, mô hình hỏi và trả lời, tạo hình ảnh bằng cách sử dụng các mô hình chú ý, chú thích hình ảnh,nhận dạng chữ viết, dịch máy, mô hình ngôn ngữ

***2.3.2 Kiến trúc mô hình LSTM***

Ba phần quan trọng của mô hình LSTM bao gồm:

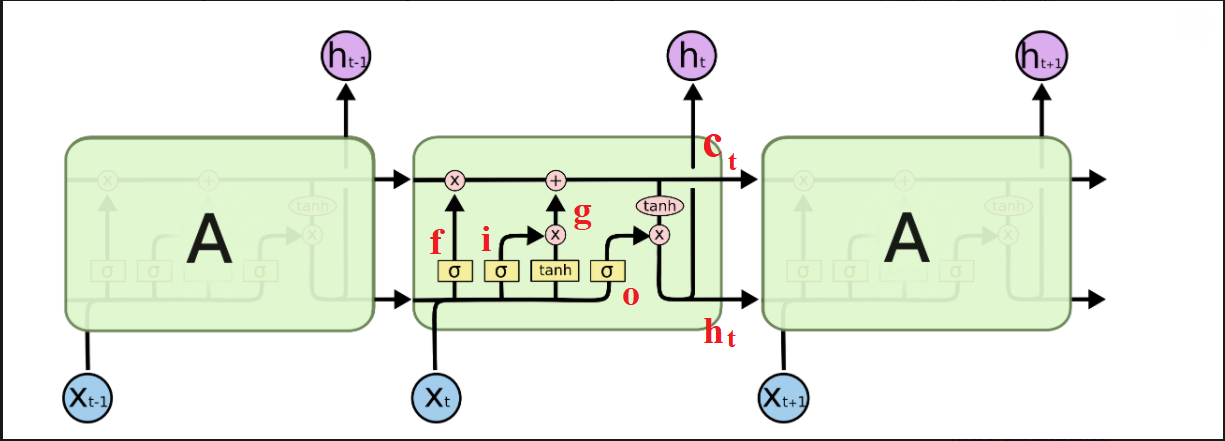
* Cổng Gates : Loại bỏ thông tin mà tại thời điểm này, không cần thiết cho việc hoàn thành công việc lặt vặt. Sự phát triển này là cơ bản để hợp lý hóa việc trình bày mạng.
* Cổng đầu ra: Chọn và cung cấp thông tin quan trọng.
* Cổng nhập liệu: Chịu trách nhiệm bổ sung thông tin cho các ô.

Logic đằng sau LSTM:

Một ô nhớ được gọi là 'trạng thái tế bào' duy trì trạng thái của nó trong thời gian dài là phần trọng tâm của mô hình LSTM. Trạng thái ô là đường phẳng đi qua điểm cao nhất bên dưới biểu đồ. Nó có xu hướng được hình dung như một đường vận chuyển mà qua đó dữ liệu chỉ đơn giản là dòng, không bị thay đổi.

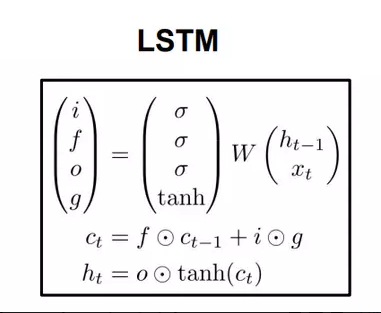
Thông tin có thể được xóa hoặc thêm ra khỏi trạng thái ô LSTM và được kiểm soát bởi các cổng. Ngoài ra, các cổng này cho phép dòng thông tin đi qua tất cả các ô. Nó bao gồm một hoạt động gia tăng theo chiều kim đồng hồ và một lớp lưới không quan tâm sigmoid giúp ích cho thành phần.

Tầng cong cung cấp giá trị số ở đâu đó trong dòng 1 và 0, trong đó và 1 biểu thị mọi thứ nên được cho qua và 0 là không có gì không nên bỏ qua.



Hình 2 : Kiến trúc mô hình LSTM

Một số tính toán :



Đầu tiên, chúng ta có *i, f, g* có công thức gần giống hệt nhau và chỉ khác mỗi ma trận tham số. Chính ma trận này sẽ quyết định chức năng khác nhau của từng cổng. σ là ký hiệu của hàm sigmoid. Quan sát hình 2 để thấy rõ hơn vị trí các cổng

* Input gate *i* - cổng vào

Cổng vào giúp quyết định bao nhiêu lượng thông tin đầu vào sẽ ảnh hưởng đến trạng thái mới. Quyết định bằng cách nào, thông qua đặc điểm của hàm sigmoid (đầu ra nằm trong khoảng [0,1]), như vậy khi một vector thông tin đi qua đây, nếu nhân với 0, vector sẽ bị triệt tiêu hoàn toàn. Nếu nhân với 1, hầu hết thông tin sẽ được giữ lại.

* Tương tự như vậy, *f* là forget gate - cổng quên

Cổng quyết định sẽ bỏ đi bao nhiêu lượng thông tin đến từ trạng thái trước đó

* Cuối cùng, cổng o là output gate - cổng ra

Cổng điều chỉnh lượng thông tin có thể ra ngoài yt và lượng thông tin truyền tới trạng thái tiếp theo.

* Tiếp theo, g thực chất cũng chỉ là một trạng thái ẩn được tính dựa trên đầu vào hiện tại xt và trạng thái trước ht−1​. Tính hoàn toàn tương tự như input gate, chỉ thay vì dùng sigmoid, ta dùng tanh. Kết hợp hai điều này lại để cập nhật trạng thái mới.
* Cuối cùng, ta có ct​ là bộ nhớ trong của LSTM. Nhìn vào công thức, có thể thấy nó là tổng hợp của bộ nhớ trước ct−1​ đã được lọc qua cổng quên *f*, cộng với trạng thái ẩn g đã được lọc bởi cổng vào *i*. Cell state sẽ mang thông tin nào quan trọng truyền đi xa hơn và sẽ được dùng khi cần. Đây chính là long term memory.
* Sau khi có ct ​, ta sẽ đưa nó qua cổng ra để lọc thông tin một lần nữa, thu được trạng thái mới ht

Nếu nhìn kỹ một chút, ta có thể thấy RNN truyền thống là dạng đặc biệt của LSTM. Nếu thay giá trị đầu ra của input gate là 1 và đầu ra forget gate là 0 (không nhớ trạng thái trước), ta được RNN thuần.

Nhìn một lượt qua kiến trúc LSTM, ta có thể tóm tắt:

* Thứ nhất, LSTM có long-term memory. Tuy nhiên, ht, gt ​ khá giống với RNN truyền thống, tức có short-term memory. Nhìn chung, LSTM giải quyết phần nào vanishing gradient so với RNN, nhưng chỉ một phần.
* Với lượng tính toán như trên, RNN đã chậm, LSTM nay còn chậm hơn.

Tuy vậy, với những cải tiến so với RNN thuần, LSTM đã và đang được sử dụng phổ biến. Trên thực tế, cách cài đặt LSTM cũng rất đa dạng và linh hoạt theo bài toán, tuy nhiên vẫn dựa trên LSTM chuẩn như trên.

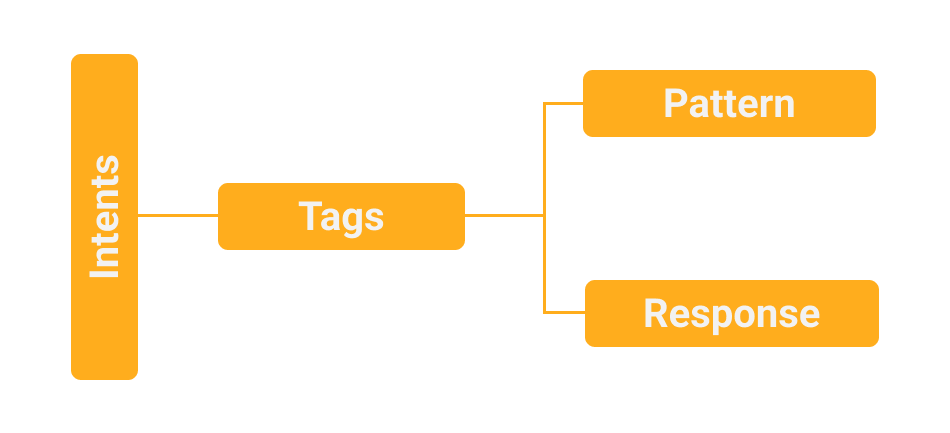
**CHƯƠNG 3:MÔ HÌNH CHATBOT BẰNG PHƯƠNG PHÁP TRUY VẤN THÔNG TIN**

**3.1 Xây dựng ngữ liệu**

***3.1.1.Cấu trúc ngữ liệu***

Mục đích cơ bản của việc xây dựng mô hình học sâu bằng cách sử dụng các câu hỏi đầu vào để thúc đẩy cuộc hội thoại theo cách có ý nghĩa hơn được xem xét trong khi xây dựng mô hình hệ thống. Ở đây, hệ thống được đề xuất hoạt động trên tập dữ liệu truy vấn hỏi đáp phạt luật Việt Nam được phát triển thủ công và nó tạo ra các phản hồi có liên quan dựa trên kho lưu trữ được xác định trước.

Chuẩn bị tập dữ liệu là nhiệm vụ đầu tiên được thực hiện trong quá trình triển khai hệ thống. Chúng tôi thậm chí có thể lấy bộ dữ liệu trực tuyến, nhưng vấn đề là, hầu hết các bộ dữ liệu không ở định dạng mà chúng tôi đang tìm kiếm. Vì vậy, trong trường hợp như vậy chúng tôi đã tạo tập dữ liệu cho riêng mình. Nó sẽ thực sự hữu ích trong việc tạo ra các phản hồi có liên quan. tập dữ liệu phải được cập nhật lặp đi lặp lại và nên đào tạo lại mô hình để cải thiện độ chính xác, nếu không, dữ liệu sẽ lỗi thời. Chúng tôi lưu trữ bộ câu hỏi và câu trả lời vào 1 tệp .Json . Mỗi đối tượng trong file Json được tạo thành từ ba thẻ chính là : tag : hiển thị nội dung chính đằng sau thông điệp của người dùng, pattern : là một câu hỏi liên quan đến các bộ luật Việt Nam ban hành liên quan đến tag, response : các câu trả lời mà chatbot sẽ xuất ra khi nó xác định mục đích đằng sau thông điệp của người dùng. Các câu hỏi và câu trả lời ở định dạng văn bản và được viết bằng tiếng Việt.



Hình 3 Cấu trúc ngữ liệu

***3.1.2.Nguồn dữ liệu***

Dữ liệu được tìm hiểu và xây dụng thủ công với hơn 2000 cặp câu hỏi và trả lời dựa trang web hỏi đáp trực tuyến về các bộ luật của Việt Nam được ban hành năm 2021 như :

* Website Luật Dương Gia
* Website Luật Việt Nam
* Website Thư viện Pháp Luật

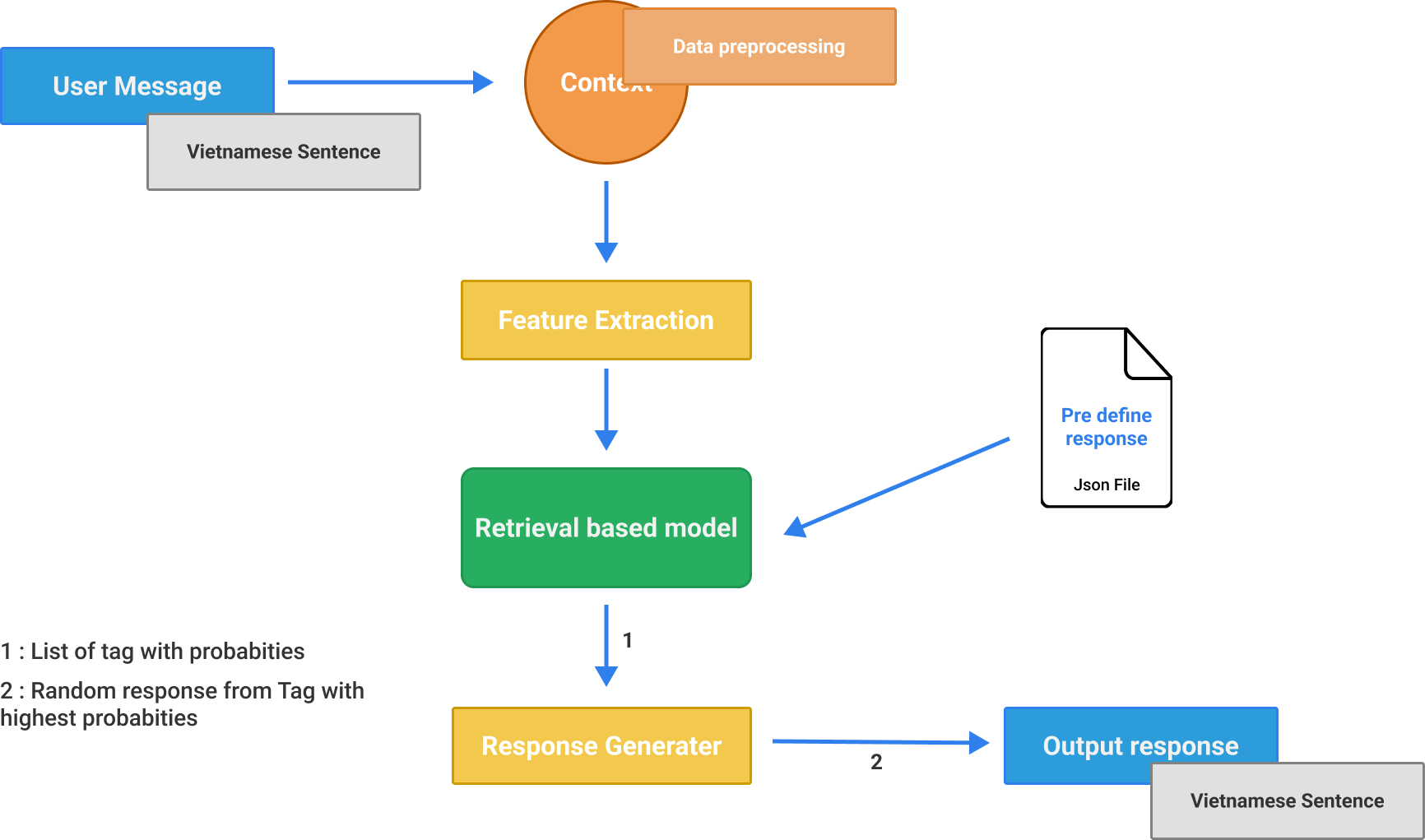
Qua đó có các câu hỏi về các bộ luật Việt Nam đã được ban hành như : Luật an toàn giao thông, luật hình sự, luật kết hôn và gia đình, luật dân sự, …

***3.1.3. Thống kê dữ liệu***

| Tên luật | Số câu |
| --- | --- |
| Luật dân sự | 186 |
| Luật hình sự | 153 |
| Luật giao thông đường bộ | 151 |
| Luật bảo vệ môi trường | 60 |
| Luật hôn nhân và gia đình | 32 |
| Luật nghĩa vụ quân sự | 30 |
| Luật phòng chống ma túy | 30 |
| Luật Bảo vệ trẻ em | 26 |
| Luật chứng khoáng | 25 |
| Giao tiếp thông thường | 25 |
| Luật an ninh mạng | 25 |
| Luật phòng chống tham nhũng | 20 |
| Luật quản lý thuế | 20 |

Bảng 1 Thống kê dữ liệu

**3.2 Mô hình Retrieval-based Chatbot**



Hình 4 Mô hình Retrieval-based Chatbot

***3.2.1 Tiền xử lý dữ liệu***

Sau khi tạo tập dữ liệu, các tác vụ tiền xử lý khác nhau được thực hiện để làm sạch dữ liệu. Nhiệm vụ tiền xử lý được gọi là một nhiệm vụ quan trọng trong khi chúng tôi đang xử lý bất kỳ nhiệm vụ NLP nào. Ở đây, dữ liệu mà chúng tôi đang xem xét được xử lý trước để loại bỏ bất kỳ nội dung nào không có bất kỳ thông tin hữu ích nào trong đó. Tác vụ tiền xử lý được thực hiện ở trong tập dữ liệu hoàn chỉnh ban đầu và cũng được thực hiện tương tự khi người dùng đưa ra một đầu vào cụ thể. Về cơ bản, chúng tôi đã sử dụng các hàm thư viện Json để đọc tập dữ liệu. Sau đó, các tác vụ phụ trong giai đoạn tiền xử lý văn bản đã được thực hiện bởi bộ công cụ Natural Language Tool Kit (NLTK), Underthesea và một số hàm xử lý chuỗi trong Python.

3.2.1.1 Tách từ tiếng việt

Thuật ngữ "tách từ" trong Tiếng Anh là "word segmentation", dịch ra Tiếng Việt là "tách từ" hoặc "phân đoạn từ". Dùng từ nào cũng được, nhưng chúng tôi dùng từ "tách từ" trong bài viết này.

Tách từ, về mặt biểu hiện, là gom nhóm các từ đơn liền kề thành một cụm từ có ý nghĩa. Ví dụ: "Cách tách từ cho Tiếng Việt." sau khi tách từ thì thành "Cách tách từ cho Tiếng\_Việt ." Về hình thức, các từ đơn được gom nhóm với nhau bằng cách nối với nhau bằng ký tự gạch dưới "\_", trong trường hợp này là từ Tiếng\_Việt. Sau khi thực hiện tách từ thì mỗi từ (token) trong câu được cách nhau bởi một khoảng trắng, trong trường hợp này như "Tiếng\_Việt ." thì từ "Tiếng\_Việt" cách đấu "." bởi 1 khoảng trắng. Đây là quy ước chung cho tất cả các ngôn ngữ của bài toán tách từ trong xử lý ngôn ngữ tự nhiên. Việc quy ước như vậy là để tạo thành chuẩn chung và để dễ xử lý hơn trong lập trình.

Về mặt ngữ nghĩa, việc tách từ văn bản đầu vào trước khi đưa vào huấn luyện mô hình máy học là để giải quyết các bài toán liên quan đến ngữ nghĩa của văn bản, tức là kết quả đầu ra mang tính suy luận dựa trên việc hiểu ý nghĩa của văn bản đầu vào. Ví dụ như các dạng bài toán: phát hiện đạo văn, tóm tắt văn bản, hỏi đáp tự động, hỗ trợ khách hàng tự động, phân tích cảm xúc văn bản, dịch máy, trợ lý ảo...

Mục tiêu của việc tách từ văn bản đầu vào là để khử tính nhập nhằng về ngữ nghĩa của văn bản. Tùy vào từng loại ngôn ngữ có những đặc điểm khác nhau mà việc tách từ văn bản cũng có độ khó khăn khác nhau. Đựa theo đặc điểm của ngôn ngữ tự nhiên mà ngôn ngữ được phân thành các loại:

Ngôn ngữ hòa kết (flexional), ví dụ: Đức, Latin, Hi Lạp, Anh, Nga...

Ngôn ngữ chắp dính (agglutinate), ví dụ: Thổ Nhĩ Kỳ, Mông Cổ, Nhật Bản, Triều Tiên,...

Ngôn ngữ đơn lập (isolate), là ngôn ngữ phi hình thái, không biến hình, đơn âm tiết, ví dụ: Việt Nam, Hán,...

Với ngôn ngữ hòa kết như Tiếng Anh, thì việc tách từ khá đơn giản vì ranh giới từ được nhận diện bằng khoảng trắng và dấu câu.

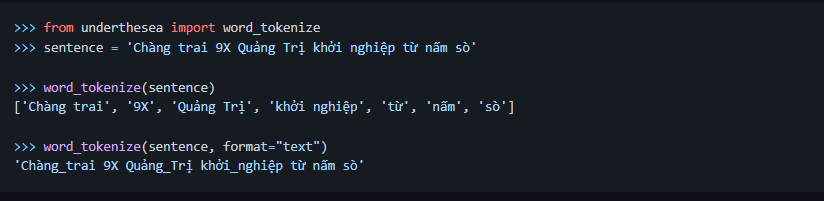
Với ngôn ngữ Tiếng Việt, thuộc loại hình đơn lập, mang đặc điểm là từ Tiếng Việt không biến đổi hình thái, ranh giới từ không được xác định mặc nhiên bằng khoản trắng. Tiếng Việt có đặc điểm là ý nghĩa ngữ pháp nằm ở ngoài từ, phương thức ngữ pháp chủ yếu là trật tự từ và từ hư. Cho nên có trường hợp một câu có thể có nhiều ngữ nghĩa khác nhau tuỳ vào cách ta tách từ như thế nào, gây nhập nhằng về ngữ nghĩa của câu. Ví dụ:

Với câu "Xoài phun thuốc sâu không ăn." có thể được tách từ như sau, với ý nghĩa hoàn toàn khác nhau:

Xoài / phun thuốc / sâu / không / ăn.

Xoài / phun / thuốc sâu / không / ăn.

Điều đó cho thấy, công việc tách từ trong Tiếng Việt không phải là chuyện dễ dàng, vì nó tạo ra các câu có ngữ nghĩa hoàn toàn khác nhau, gây ảnh hưởng đến chất lượng huấn luyện mô hình học. Chính vì vậy, công việc tách từ là rất quan trọng đối với xử lý ngôn ngữ Tiếng Việt, nhất là khi giải quyết các bài toán liên quan đến ngữ nghĩa của văn bản. Thật may là hiện nay có khá nhiều thư viện mã nguồn mở của bài toán này. Do đó, chúng ta chỉ việc cài đặt và sử dụng. Ở đây chúng tôi chọn thư viện [underthesea](https://github.com/undertheseanlp/underthesea) cho bài toán tách từ .



Hình 5 Ví dụ bài toán tách từ

3.2.1.2 Lowercase và remove unnecessary characters

Việc đưa dữ liệu về chữ viết thường là rất cần thiết. Bởi vì đặc trưng này không có tác dụng ở bài toán phân loại văn bản. Đưa về chữ viết thường giúp giảm số lượng đặc trưng (vì máy tính hiểu hoa thường là 2 từ khác nhau) và tăng độ chính xác hơn cho mô hình.

Như mình đã nói, tiền xử lý bao gồm việc loại bỏ các dữ liệu không có tác dụng cho việc phân loại văn bản. Việc này giúp:

* Giảm số chiều đặc trưng, tăng tốc độ học và xử lý
* Tránh làm ảnh hưởng xấu tới kết quả của mô hình
* Các dấu ngắt câu, số đếm và các ký tự đặc biệt khác không giúp bạn phân loại một văn bản thuộc chuyên mục nào. Do đó, chúng ta nên loại bỏ nó đi.
* Riêng với số đếm, ngày tháng, email (Các token đặc biệt). Nếu có thể bạn nên đưa nó về các token chung như: <number>, <date>, <email>, … Việc này có thể không giúp ích cho mô hình học tốt hơn nhưng sẽ giúp ích cho bạn trong việc giữ được mạch của dữ liệu.

3.2.1.3 Loại bỏ stopword

Cũng giống như việc loại bỏ các ký tự đặc biệt ở trên, nhưng bây giờ là các từ tiếng Việt hẳn hoi. Vậy tại sao cần loại bỏ?

Stopword là các từ xuất hiện nhiều ở tất cả các chuyên mục cần phân loại. Do đó, chúng là các đặc trưng không có tác dùng cho việc phân loại văn bản.

Các stopword thường là các từ nối (của, là, có, được, những,…) và các từ đặc trưng của dữ liệu (ví dụ như các từ “máy bay, tiếp viên” là các stopword nếu làm bài phân loại đánh giá khách hàng của doanh nghiệp vận tải hàng không. Ở đây chúng tôi xây dựng bộ stopword với khoảng 100 từ .

***3.2.2 Trích xuất đặc trưng***

Trích xuất đặc trưng là nhiệm vụ tiếp theo mà chúng ta cần quan tâm, Sau khi làm sạch dữ liệu, nó sẽ được chuyển đổi theo cách mà máy có thể hiểu được. Vì mục đích đó, chúng tôi sử dụng các kỹ thuật tạo tính năng này. Chúng tôi có thể thực hiện tạo tính năng theo nhiều cách khác nhau. Ở đây, khái niệm Bag Of Words (BOW) được xem xét để tạo ra các tính năng từ văn bản đầu vào. Đầu ra từ trình tạo tính năng này sẽ được cung cấp làm đầu vào cho mô hình học sâu để dự đoán phản hồi.

Một Bag Of Words là một biểu diễn của văn bản mô tả sự xuất hiện của các từ trong một tài liệu. Chúng tôi chỉ theo dõi số lượng từ và bỏ qua các chi tiết ngữ pháp và thứ tự từ. Nó được gọi là “túi” các từ vì bất kỳ thông tin nào về thứ tự hoặc cấu trúc của các từ trong tài liệu đều bị loại bỏ. Mô hình chỉ quan tâm đến việc các từ đã biết có xuất hiện trong tài liệu hay không, chứ không phải ở vị trí nào trong tài liệu.

Dưới đây là ví dụ về Bag Of Words

Ví dụ :

Câu 1: ” Chào mừng bạn đến với học tập, Bây giờ hãy bắt đầu học tập”

Câu 2: “ Học tập là một bài thực hành tốt "

Sau khi đã được tiền xử lý dữ liệu đưa về viết thường và xóa các ký tự đặc biệt và các từ không chứa nhiều thông tin cho việc dự đoán thì các câu sẽ trở thành :

Câu 1: ” chào mừng bạn đến với học tập bây giờ hãy bắt đầu học tập”

Câu 2: “ học tập là bài thực hành tốt "

Xem qua tất cả các từ trong văn bản trên và lập danh sách tất cả các từ trong từ vựng mô hình của chúng ta :

* chào mừng
* bạn
* đến
* học tập
* bây giờ
* bắt đầu
* bài thực hành
* tốt

Bây giờ vì từ vựng chỉ có 8 từ, chúng ta có thể sử dụng biểu diễn tài liệu có độ dài cố định là 8, với một vị trí trong vectơ để cho điểm mỗi từ.

Phương pháp tính điểm mà chúng tôi sử dụng ở đây là đếm sự hiện diện của mỗi từ và đánh dấu 0 cho sự vắng mặt. Phương pháp cho điểm này được sử dụng phổ biến .

Đối với câu 1, số lượng từ như sau:

| Từ vựng | Tần suất |
| --- | --- |
| chào mừng | 1 |
| bạn | 1 |
| đến | 1 |
| học tâp | 2 |
| bây giờ | 1 |
| bắt đầu | 1 |
| bài thực hành | 0 |
| tốt | 0 |

Dưới dạng vector sẽ là ➝ [ 1,1,1,2,1,1,0,0 ]

Đối với câu 2, số lượng từ như sau:

| Từ vựng | Tần suất |
| --- | --- |
| chào mừng | 0 |
| bạn | 0 |
| đến | 0 |
| học tâp | 1 |
| bây giờ | 0 |
| bắt đầu | 0 |
| bài thực hành | 1 |
| tốt | 1 |

Dưới dạng vector sẽ là ➝ [ 0,0,0,1,0,0,1,1 ]

Cách tiếp cận được sử dụng trong ví dụ hai là cách thường được sử dụng trong kỹ thuật Bag-of-Words, lý do là các bộ dữ liệu được sử dụng trong Machine learning rất lớn và có thể chứa từ vựng vài nghìn hoặc thậm chí hàng triệu từ. Do đó, xử lý trước văn bản trước khi sử dụng bag-of-words là một cách tốt hơn để thực hiện.

Có nhiều bước tiền xử lý khác nhau có thể tăng hiệu suất của Bag-of-Words. Một số trong số chúng được giải thích rất chi tiết trong phần tiền xử lý dữ liệu.

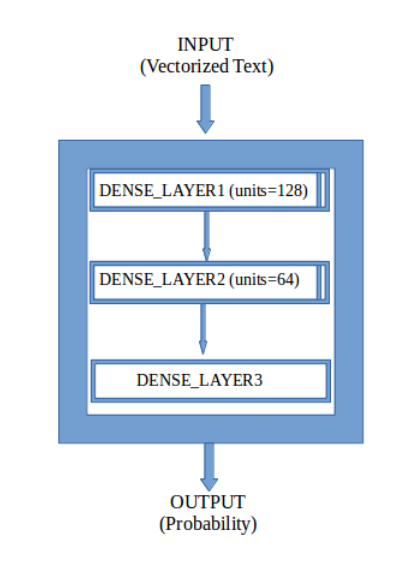
Trong các ví dụ trên, chúng tôi sử dụng tất cả các từ từ từ vựng để tạo thành một vectơ, đây không phải là cách thực tế và cũng không phải là cách tốt nhất để triển khai mô hình BoW. Trong thực tế, chỉ một số từ trong từ vựng, tốt hơn là hầu hết các từ thông dụng được sử dụng để tạo thành vectơ.

***3.2.3 Module phân lớp***

Hệ thống chatbot truy xuất dựa trên DL được triển khai tại đây.Chúng tôi đã đề cập rằng mục đích chính của chúng tôi là xây dựng một hệ thống học sâu lấy đầu vào là một tập hợp các truy vấn được người dùng yêu cầu với nhiều tính cách và tạo ra các câu trả lời chính xác nhất và thúc đẩy cuộc trò chuyện trở nên nhiều hơn một cách có ý nghĩa. Ở đây chúng tôi dùng 2 model đó là ANN và LSTM :

* Ở mô hình ANN :

Mô hình học sâu từ keras được gọi là Sequential được sử dụng ở đây để xây dựng mô hình , có thể hiểu rằng có 3 lớp cho mô hình DL. Ngoài ra, 2 lớp bỏ học (Dropout) cũng có ở giữa mỗi lớp dày đặc (Dense Layer ). Trong khái niệm 3 lớp này, lớp đầu tiên có 128 tế bào thần kinh, lớp thứ hai có 64 tế bào thần kinh và lớp thứ ba có số ý định (intents) là số lượng tế bào thần kinh. Về cơ bản, mục đích của mạng này là dự đoán mục đích nào sẽ được lựa chọn nếu chúng ta được cung cấp một số dữ liệu đầu vào. Đầu vào cho mô hình DL Sequential sẽ là văn bản được vector hóa.



Hình 6 Sequential Model of Keras

* Ở mô hình LSTM :

Bộ nhớ ngắn hạn dài, được gọi là LSTM, là một loại mạng nơ-ron tuần hoàn có thể học và lưu trữ thông tin trước đó. Chúng có nhiều lớp ẩn và kết nối phản hồi, khiến chúng đặc biệt tốt trong việc xử lý chuỗi dữ liệu. Khi dữ liệu đi qua mỗi lớp, thông tin không liên quan sẽ bị loại bỏ, chỉ để lại thông tin có liên quan. LSTM không xử lý từng điểm dữ liệu một cách độc lập, mà cố gắng phân tích mối quan hệ giữa từng điểm dữ liệu sau mỗi lần lặp. Điều này cho phép mô hình ghi nhớ và giữ lại thông tin có thể được sử dụng để xử lý các điểm dữ liệu mới.

Chúng tôi sẽ thêm một vài lớp LSTM và đặt return\_sequences thành 1. Điều này sẽ làm cho đầu ra của lớp trở thành đầu ra trạng thái ẩn cuối cùng.

Các đầu ra LSTM sẽ được đưa vào một lớp Dense với chức năng kích hoạt SoftMax vì nó cho phép chúng ta thực hiện phân loại nhiều lớp .

Trong mô-đun trước, chúng ta đã thảo luận về việc tạo tính năng. Ở đây, mục đích nhúng được thực hiện bởi khái niệm Bag Of Words (BOW). Sau khi chuyển đổi văn bản sang túi từ, chúng ta nhận được chỉ mục tương ứng cho từng từ được sử dụng trong dữ liệu huấn luyện. Điều này sẽ được chuyển đến mô hình DL. Dữ liệu sẽ đi qua từng lớp dày đặc. Sau đó, mô hình được đào tạo với Stochastic gradient descent (SGD). Số lớp và số nơ-ron được xem xét sau khi thực hiện điều chỉnh tham số. Hệ thống này thậm chí còn được đào tạo có và không có lớp bỏ học. Sau khi đào tạo mô hình, toàn bộ điều được chuyển thành một mảng numpy và được lưu lại. Chúng ta có thể sử dụng mô hình này để dự đoán ý định chính xác nhất.

Với mục đích dự đoán, ngưỡng sai số (error threshold) 0,25 được sử dụng. Điều này được sử dụng chỉ để tránh danh sách các ý định quá nhiều. Vì vậy, khi đầu vào được chuyển qua mô hình DL, nó thường tạo danh sách các ý định từ kho lưu trữ và đó là các xác suất tương ứng cùng với khả năng chúng khớp với ý định chính xác. Sau đó, chúng tôi lọc lại danh sách bằng cách sử dụng khái niệm về giá trị lưu giữ thông tin lỗi. Chỉ những ý định có xác suất lớn hơn ngưỡng mới được xem xét. Để xây dựng mô hình, cả hai chức năng kích hoạt ReLu và softmax đều được sử dụng.

Hàm ReLu được sử dụng cả trong lớp đầu vào và lớp ẩn và hàm Softmax được sử dụng trong lớp đầu ra. Chúng tôi đã sử dụng ReLu trong đầu vào và lớp ẩn vì nó thường có thể dẫn đến các giải pháp ít dày đặc hơn. Nhiều hàm kích hoạt khác, chẳng hạn như sigmoid, sẽ luôn xuất ra giá trị khác 0. Điều này có nghĩa là mọi tế bào thần kinh sẽ được xem xét ở mỗi lần vượt qua, ngay cả khi chúng ta không cần nó. Hàm ReLU không phải lúc nào cũng xuất ra giá trị khác 0, do đó, nó dẫn đến việc sử dụng ít nơ-ron hơn và ít phụ thuộc hơn giữa các đối tượng. Hàm softmax được sử dụng trong lớp đầu ra vì mục đích của hàm này là đưa ra xác suất của một số lớp nhất định xảy ra. Đây cũng là mục đích của chúng tôi là lấy xác suất của từng ý định trong tập dữ liệu bằng cách so sánh nó với truy vấn đã nhập.

***3.2.4 Tạo phản hồi***

Tạo phản hồi là nhiệm vụ cuối cùng của mô hình hệ thống. Mô hình ANN và LSTM cung cấp cho chúng ta danh sách các ý định sau khi lọc nó với ngưỡng lỗi cùng với các xác suất tương ứng của chúng. Nhiệm vụ của mô-đun tạo phản hồi là chọn một mô-đun từ danh sách đó rất phù hợp cho truy vấn đã nhập. Mô-đun này lấy đầu ra từ mô-đun DL làm đầu vào. Sau đó, danh sách các ý định này được kiểm tra với kho lưu trữ (ở đây là tệp Json của các ý định) và một trong những xác suất cao nhất phù hợp hoàn toàn với đầu vào được tạo ra làm phản hồi cuối cùng.

**CHƯƠNG 4: THỰC NGHIỆM**

**4.1 Môi trường thực nghiệm**

Python 3.8.8

Các bộ thư viện được sử dụng.

* Natural Language Tool Kit (NLTK) : Thư viện hỗ trợ tiền xử lý dữ liệu như : Xóa ký tự lẻ, đưa về chữ viết thường …
* Underthesea : Thư viện sử dụng cho việc tách từ
* Matplotlib: thư viện sử dụng để vẽ đồ thị trong python
* Sklearn (PolymialFeature, feature\_selection, model\_selection, metrics, linear\_model, ansemble, neighbors )
* Pickle: lưu trữ modul (model,...) quan trọng.
* Tkinter : kinter là một gói trong Python có chứa module Tk hỗ trợ cho việc lập trình GUI

**4.2 Tập dữ liệu**

Dữ liệu được tìm hiểu và xây dụng thủ công với hơn 2000 cặp câu hỏi và trả lời dựa trang web hỏi đáp trực tuyến về các bộ luật của Việt Nam được ban hành năm 2021.

Dữ liệu có dạng như sau:

{

**"tag":** "den\_vangnhapnhay",

**"patterns":** ["Khi gặp tín hiệu đèn giao thông màu vàng nhấp nháy, người tham gia giao thông có được phép đi tiếp không?",

"gặp đèn vàng bật nhấp nháy thì ta phải làm gì khi chưa đến vạch dừng",

" Khi chưa đến vạch dừng mà gặp đèn vàng bật nhấp nháy thì ta phải làm gì "],

**"responses":** ["Được đi nhưng phải giảm tốc độ, chú ý quan sát, nhường đường cho người đi bộ qua đường."]

}

Ý nghĩa các thẻ ở đây là :

* tag : hiển thị nội dung chính đằng sau thông điệp của người dùng,
* pattern : là một câu hỏi liên quan đến các bộ luật Việt Nam ban hành liên quan đến tag,
* response : các câu trả lời mà chatbot sẽ xuất ra khi nó xác định mục đích đằng sau thông điệp của người dùng.
* Các câu hỏi và câu trả lời ở định dạng văn bản và được viết bằng tiếng Việt.

**4.3 Tham số chạy mô hình**

Tham số chạy mô hình ANN :

| Parameter | Value |
| --- | --- |
| Layers | Dense |
| Activation | ReLu and Softmax |
| Optimizer | SGD |
| Threshold | 0.25 |
| Epochs | 300 |
| Batch Size | 8 |

Bảng 2 Tham số chạy mô hình ANN

Tham số chạy mô hình LSTM :

| Parameter | Value |
| --- | --- |
| Layers | LSTM |
| Activation | Softmax |
| Optimizer | Adam |
| Epochs | 300 |
| Batch Size | 8 |

Bảng 3 Tham số chạy mô hình LSTM

**4.4 Tiêu chí đánh giá mô hình**

***4.4.1 Phân lớp***

Kết quả :

* Ở mô hình ANN đánh giá phân lớp đạt khoảng 0.6967 điều đó chứng tỏ kết quả mong muốn của người dùng so với kết quả được dự đoán có thể đúng lên đến 69.67%.
* Ở mô hình LSTM đánh giá phân lớp khá thấp chỉ đạt điều đó chứng tỏ kết quả mong muốn của người dùng so với kết quả được dự đoán có thể đúng khoảng 6%.

| Độ đo | Mô hình ANN | Mô hình LSTM |
| --- | --- | --- |
| Phân lớp | 0.6967 | 0,06 |

Bảng 4 : Đánh giá phân lớp của mô hình

***4.4.2 Độ đo BLEU***

BLEU(Bilingual Evaluation Understudy) là một thuật toán so sánh chuỗi thường được sử dụng trong bài toàn dịch máy (Machine Translation) và nó được xem như một độ đo khi so sánh một bản dịch (candidate translation) với một hay nhiều bản dịch tham khảo (reference translation). Không những thế, BLEU còn được sử dụng trong các bài toán sinh văn bản (text generation) như:

* Language generation
* Image caption generation
* Text summarization
* Speech recognition

Giá trị của BLEU Score giao động từ 0 đến 1 .1 được xem là giá trị hoàn hảo và 0 là giá trị không hoàn hảo khi sử dụng BLEU Score

Một số ưu điểm của BLEU Score:

* + Nhanh và ít tốn kém.
  + Dễ hiểu.
  + Không phụ thuộc vào ngôn ngữ.
  + Có độ tương đồng cao với cách đánh giá của con người.
  + Được áp dụng rộng rãi.

Cách hoạt động:

Thuật toán này hoạt động bằng cách đếm số matching n-grams của candidate và reference ,kết quả sẽ là số match chia cho số từ của candidate. Các match này không phụ thuộc vào vị trí. Càng match nhiều tức là càng tốt.

Tuy nhiên, nếu một từ match với reference nhưng được lặp lại nhiều lần trong candidate sẽ dẫn đến việc đánh giá bị sai (được gọi là overgenerate "resonable" words).

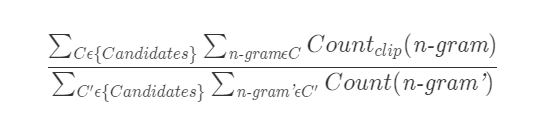
Ví dụ:

reference = [['this', 'is', 'a', 'test']]

candidate = ['this', 'this', 'this', 'this']

Do đó, khi đếm matching n-grams cần chú ý cả số lần xuất hiện của từ trong reference, một từ trong reference khi được match rồi thì không nên match nữa.

BLEU còn được dùng để đánh giá một corpus (tập hợp của các sentence, hay một đoạn văn) khá là tốt. Đầu tiên là tính số match với từng câu. Cộng các số này rồi chia cho tổng số n-gram từ các câu là ra modified precision score cho test corpus.



Dữ liệu đánh giá : Dữ liệu đánh giá được lấy khoảng 30% trong Bộ dữ liệu đã được tạo ở trên và dữ liệu này không thuộc dữ liệu dùng để train model.

Kết quả :

* Ở mô hình ANN BLEU score đạt khoảng 66.64 điều đó chứng tỏ kết quả mong muốn của người dùng so với kết quả được dự đoán có thể đúng lên đến 66.64%.
* Ở mô hình LSTM cho BLEU score khá thấp chỉ đạt điều đó chứng tỏ kết quả mong muốn của người dùng so với kết quả được dự đoán có thể đúng khoảng 1.75%.

| Độ đo | Mô hình ANN | Mô hình LSTM |
| --- | --- | --- |
| BLEU | 0.6664 | 0,0175 |

Bảng 4 : Độ đo BLEU của mô hình

**4.5 Kết quả thực nghiệm.**

Chúng tôi đã thử nghiệm cả hai mô hình chatbot bằng cách sử dụng cùng một bộ câu hỏi và so sánh mức độ chính xác mà chúng có thể dự đoán câu trả lời chính xác. Cả hai đều mang lại cho chúng tôi hiệu suất tốt, tuy nhiên, ANN có thể đưa ra dự đoán chính xác hơn nhiều về các câu hỏi mà mô hình nhận được so với mô hình LSTM. Mô hình ANN trả lời đúng hơn 65% câu hỏi trong khi LSTM chỉ có thể trả lời đúng khoảng 1%-10% câu hỏi

Một số kết quả thực nghiệm ở mô hình ANN :

Kết quả 1 :

{

"**question**": "Lệnh gọi khám sức khỏe phải được giao cho công dân trước thời điểm khám sức khỏe bao nhiêu ngày ?",

"**answer**": [

"15 ngày"

],

"**predict**": "15 ngày"

}

Kết quả 2 :

{

"**question**": "Chất hướng thần là gì ?",

"**answer**": [

"chất hướng thần là chất kích thích hoặc ức chế thần kinh hoặc gây ảo giác, nếu sử dụng nhiều lần có thể dẫn tới tình trạng nghiện đối với người sử dụng"

],

"**predict**": "chất hướng thần là chất kích thích hoặc ức chế thần kinh hoặc gây ảo giác, nếu sử dụng nhiều lần có thể dẫn tới tình trạng nghiện đối với người sử dụng"

}

Kết quả 3 :

{

"**question**": "Nghia vụ của người kê khai tài sản , thu nhập",

"**answer**": [

"Người có nghĩa vụ kê khai phải kê khai tài sản, thu nhập và biến động về tài sản, thu nhập của mình, của vợ hoặc chồng, con chưa thành niên theo quy định của Luật này"

],

"**predict**": "Người có nghĩa vụ kê khai phải kê khai tài sản, thu nhập và biến động về tài sản, thu nhập của mình, của vợ hoặc chồng, con chưa thành niên theo quy định của Luật này."

}

Kết quả 4 :

{

"**question**": "người nghiện ma túy là gì ?",

"**answer**": [

"người sử dụng chất ma túy, thuốc gây nghiện, thuốc hướng thần và bị phụ thuộc vào các chất này"

],

"**predict**": "người sử dụng chất ma túy, thuốc gây nghiện, thuốc hướng thần và bị phụ thuộc vào các chất này."

}

Kết quả 5 :

{

"**question**": " Khái niệm quyền của trẻ em khuyết tật?",

"**answer**": [

" Trẻ em khuyết tật được hưởng đầy đủ các quyền của trẻ em và quyền của người khuyết tật theo quy định của pháp luật.Được hỗ trợ, chăm sóc, giáo dục đặc biệt để phục hồi chức năng, phát triển khả năng tự lực và hòa nhập xã hội"

],

"**predict**": " Trẻ em khuyết tật được hưởng đầy đủ các quyền của trẻ em và quyền của người khuyết tật theo quy định của pháp luật.Được hỗ trợ, chăm sóc, giáo dục đặc biệt để phục hồi chức năng, phát triển khả năng tự lực và hòa nhập xã hội."

}

Một số kết quả thực nghiệm ở mô hình LSTM :

Kết quả 1 :

{

"**question**": " Tàng trữ thông tin bịa đặt có bị ở tù?",

"**answer**": [

" Bị phạt từ 05 năm đến 15 năm"

],

"**predict**": Người nào thực hiện một trong các hành vi sau đây xâm phạm chỗ ở của người khác, thì bị phạt cải tạo không giam giữ đến 02 năm hoặc phạt tù từ 03 tháng đến 02 năm:Khám xét trái pháp luật chỗ ở của người khác;Dùng vũ lực, đe doạ dùng vũ lực, gây sức ép về tinh thần hoặc thủ đoạn trái pháp luật khác buộc người khác phải rời khỏi chỗ ở hợp pháp của họ;Dùng mọi thủ đoạn trái pháp luật nhằm chiếm, giữ chỗ ở hoặc cản trở trái phép, không cho người đang ở hoặc quản lý hợp pháp chỗ ở được vào chỗ ở của họ;Tự ý xâm nhập chỗ ở của người khác mà không được sự đồng ý của chủ nhà hoặc người quản lý hợp pháp"

}

Kết quả 2 :

{

"**question**": " Số tiền bị phạt khi Chạy xe môtô , xe gắn máy quá tốc độ quy định từ 05 km / h đến dưới 10 km / h?",

"**answer**": [

" Phạt từ 100.000 đồng đến 200.000 đồng"

],

"**predict**": Cầm giữ tài sản là việc bên có quyền (sau đây gọi là bên cầm giữ) đang nắm giữ hợp pháp tài sản là đối tượng của hợp đồng song vụ được chiếm giữ tài sản trong trường hợp bên có nghĩa vụ không thực hiện hoặc thực hiện không đúng nghĩa "

}

**CHƯƠNG 5 KẾT LUẬN VÀ ĐÁNH GIÁ**

**5.1 Kết luận**

Chatbot là một trong những nghiên cứu đang phát triển là trong lĩnh vực trí tuệ nhân tạo. Công cụ đàm thoại có thể được sử dụng trong bất kỳ lĩnh vực nào để cung cấp hỗ trợ cho người dùng. Hệ thống được đề xuất là một trò chuyện dựa trên truy xuất hoạt động dựa trên nueral network. Khái niệm học sâu được sử dụng cho xây dựng mô hình. Vì chatbot là một hệ thống dựa trên truy xuất, chatbot đặc biệt hoạt động trong một miền khép kín. Ở đây, chúng tôi đã xây dựng một bộ dữ liệu của truy vấn pháp luật và hệ thống hoạt động tốt trên tên miền đó, độ chính xác đào tạo của hệ thống được đề xuất là 90%. Ở đây khi người dùng hỏi bất kỳ truy vấn nào trong miền, hệ thống sẽ tạo ra phản hồi tương ứng và thúc đẩy cuộc trò chuyện một cách có ý nghĩa. Cập nhật thủ công bộ dữ liệu là cần thiết theo định kỳ để cải thiện hiệu quả của hệ thống và giúp hệ thống trả lời hầu hết các truy vấn của người dùng.

**5.2 Đánh giá**

Vì bộ ngữ liệu chưa được đầy đủ và còn hạn chế, mô hình chúng tôi chưa được training một cách hiệu quả, dẫn đến một vài trường hợp sinh câu trả lời tối nghĩa, không hợp lý, tự nhiên.

Chúng tôi không nhận định đây là một mô hình quá tốt, chưa áp dụng được vào thực tế, chỉ sử dụng để hỗ trợ môn học.

**TÀI LIỆU THAM KHẢO**

**Tiếng Anh**

[1] Retrieval based Chatbots — Using NLTK & Keras - [Sol M. Lozano](https://medium.com/@solozano0725?source=post_page-----e4f86b262b17--------------------------------)

[2] AUBot : A Chatbot for Student FAQs - Fatima Ghadieh

[3] AUBot : A Chatbot for Student FAQs - AUBot : A Chatbot for Student FAQs

[4] Bag of Words (BoW) : https://www.mygreatlearning.com/blog/bag-of-words/

[5] Chatbot for university related FAQs

[6] Neural Networks by Christos Stergiou and Dimitrios Siganos

[7] Neural Networks and Learning Machines Third Edition by Simon Haykin

**Tiếng việt**

[1] Tiền xử lý dữ liệu tiếng việt : <https://blog.luyencode.net/phan-loai-van-ban-tieng-viet/>

[2] Tổng quan về Artificial Neural Network : <https://viblo.asia/p/tong-quan-ve-artificial-neural-network-1VgZvwYrlAw>

[3] Bài giảng Máy học và mạng Neural - Bài 4: Mạng nơron nhân tạo - Vũ Đức Lung : https://doc.edu.vn/tai-lieu/bai-giang-may-hoc-va-mang-neural-bai-4-mang-noron-nhan-tao-vu-duc-lung

[4] Recurrent Neural Network:Từ RNN đến LSTM: https://viblo.asia/p/recurrent-neural-network-tu-rnn-den-lstm-gGJ597z1ZX2