ĐẠI HỌC QUỐC GIA HÀ NỘI
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ

NGUYỄN ANH NGỌC

DỰ ĐOÁN TƯƠNG TÁC THUỐC TỪ VĂN BẢN Y SINH SỬ DỤNG MẠNG NƠ RON TÍCH CHẬP

LUẬN VĂN THẠC SĨ CÔNG NGHỆ THÔNG TIN
NGƯỜI HƯỚNG DẪN KHOA HỌC: TS. ĐẶNG THANH HẢI

Hà Nội 2021

ĐẠI HỌC QUỐC GIA HÀ NỘI
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ

NGUYỄN ANH NGỌC

DỰ ĐOÁN TƯƠNG TÁC THUỐC TỪ VĂN BẢN Y SINH SỬ DỤNG MẠNG NƠ RON TÍCH CHẬP

NGÀNH: CÔNG NGHỆ THÔNG TIN
CHUYÊN NGÀNH: HỆ THỐNG THÔNG TIN
MÃ SỐ: 8480104.01

LUẬN VĂN THẠC SĨ CÔNG NGHỆ THÔNG TIN
NGƯỜI HƯỚNG DẪN KHOA HỌC: TS. ĐẶNG THANH HẢI

Hà Nội 2021

LỜI CẢM ƠN

Trước tiên, tôi xin bày tỏ sự biết ơn chân thành và sâu sắc nhất tới TS. Đặng Thanh Hải – Giáo viên hướng dẫn trực tiếp – đã hết lòng hỗ trợ và giúp đỡ tôi trong quá trình nghiên cứu và hoàn thiện luận văn thạc sĩ của mình. Đồng thời tôi cũng gửi lời cám ơn chân thành đến các thành viên nhóm nghiên cứu của TS. Đặng Thanh Hải đã hỗ trợ tôi rất nhiều trong quá trình thực hiện luận văn này.

Tôi cũng xin gửi lời cảm ơn chân thành tới các thầy, các cô là giảng viên của trường Đại học Công nghệ đã tận tình dạy dỗ và hướng dẫn cho tôi trong suốt quá trình học tập thạc sĩ tại trường.

Mặc dù đã hết sức cố gắng hoàn thành luận văn nhưng chắc chắn sẽ không tránh khỏi những sai sót. Kính mong nhận được sự cảm thông, chỉ bảo tận tình của các quý thầy cô và các bạn.

Tôi xin chân thành cảm ơn!

LỜI CAM ĐOAN

Tôi xin cam đoan đây là công trình nghiên cứu khoa học của riêng tôi và được sự hướng dẫn khoa học của TS. Đặng Thanh Hải. 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 luận văn 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 luận văn của mình.

Học viên Cao học

Nguyễn Anh Ngọc

MỤC LỤC

DANH MỤC HÌNH VẼ

DANH MỤC BẢNG BIỂU

DANH MỤC TỪ VIẾT TẮT

CID: Chemical-induced Disease – Quan hệ Tác dụng phụ của thuốc gây ra bệnh từ văn bản y sinh.

LSTM: Long Short-Term Memory – Bộ nhớ Ngắn hạn Dài.

RNN: Recurrent Neural Network – Mạng nơ-ron hồi quy.

POS: Part of speech – Từ loại.

CNN: Convolutional Neural Network – Mạng nơ-ron tích chập.

ME: Maximum Entropy Model – Mô hình Entropy Cực đại.

PP: Post processing – Tiền xử lý.

Chương 1: GIỚI THIỆU CHUNG

1.1 Giới thiệu

Tương tác thuốc (bệnh-thuốc) là một loại quan hệ giữa các thực thể y sinh, ví dụ: quan hệ <tác dụng phụ>, quan hệ <điều trị> có thể xảy ra giữa bệnh và thuốc. Các nhà khoa học cần tự động trích xuất thông tin liên quan, ví dụ, mối quan hệ ngữ nghĩa giữa các thực thể y sinh, từ các cơ sở dữ liệu này. Ví dụ, các nhà khoa học cần biết loại thuốc nào chữa khỏi một loại bệnh nhất định hoặc loại bệnh nào là tác dụng phụ (Chemical-Induced Diseases CID) của một loại thuốc nhất định. Những mối quan hệ này có thể giúp các chuyên gia cập nhật kiến thức và nâng cao chuyên môn trong lĩnh vực của họ. Các mối quan hệ này có thể được phát hiện từ nhiều văn bản khác nhau trong tài liệu y sinh [1].

Hiểu được mối quan hệ giữa hóa chất và bệnh tật là rất quan trọng trong các nhiệm vụ y sinh khác nhau, chẳng hạn như khám phá thuốc mới và phát triển liệu pháp mới.

Trong các phương pháp học máy, mạng nơ-ron tích chập (CNN) là một phương pháp học máy mạnh mẽ được đề xuất gần đây đã thể hiện tiềm năng lớn cho nhiều nhiệm vụ xử lý ngôn ngữ tự nhiên như phân tích quan điểm/cảm xúc, cũng như trong trích xuất tương tác thuốc [2]. Một nghiên cứu cải tiến về mạng tích chập dựa trên sự phụ thuộc đã cho kết quả khá tốt trong việc mô hình hóa câu trong văn bản [3].

Việc trích xuất mối quan hệ CID được mô tả trong các tài liệu y sinh được xác định ở cấp độ tài liệu, tức là các mối quan hệ có thể được mô tả ở những câu khác nhau trong tài liệu đó. Hơn nữa, nhiệm vụ trích xuất quan hệ CID yêu cầu mối quan hệ giữa các bệnh và hóa chất cụ thể nhất [4].

Do các thực thể hóa chất và bệnh tật có thể có nhiều đề cập ở các câu khác nhau trong một tài liệu, vì vậy chúng ta coi trường hợp này là "cấp độ nội câu" khi đề cập đến hóa chất và bệnh tật ở trong cùng một câu hoặc là "cấp độ liên câu" nếu ngược lại. Vì vậy, việc trích xuất quan hệ CID có thể được đơn giản hóa từ cấp độ tài liệu đến cấp độ đề cập, xem xét các câu sau:

S1	Possible intramuscular **midazolam**[D008874]-associated **cardiorespiratory arrest**[D006323] and **death**[D003643].
S2	**Midazolam hydrochloride**[D008874] is commonly used for dental or endoscopic procedures.
S3	Although generally consisted safe when given intramuscularly, intravenous administration is known to cause respiratory and **cardiovascular depression**[D012140].
S4	This report describes the first published case of **cardiorespiratory arrest**[D006323] and **death**[D003643] associated with intramuscular administration of **midazolam**[D008874].
S5	Information regarding **midazolam**[D008874] use is reviewed to provide recommendation for safe administration.
R1	D008874-D012140
R2	D008874-D006323

Bảng 1.1 Một ví dụ của trích xuất quan hệ bệnh do hóa chất gây ra (CID)

	Chemical-Disease concept pair	Relation type
R1	D008874-D012140	CID
R2	D008874-D006323	CID
R3	D008874-D003643	None

Các câu trên Bảng 1.1 được trích từ cùng một tài liệu (PMID: 2375138). Trong số đó, các từ ***in đậm*** đề cập đến hóa chất và bệnh tật, trong đó ***midazolam*** và ***Midazolam hydrochloride*** đề cập đến cùng một khái niệm hóa học có số định danh là D008874 (C1), ***cardiorespiratory arrest*** đại diện cho khái niệm bệnh lý có số định danh là D006323 (D1), ***cardiovascular depression*** đề cập đến một khái niệm bệnh có số định danh là D012140 (D2) và ***death*** đề cập đến một khái niệm bệnh khác với số định danh là D003643 (D3). Hóa chất C1 có hai lần cùng xuất hiện với bệnh D1 ở mức độ nội câu ở cả hai câu (S1) và (S4), trong khi đó cùng xuất hiện một lần với bệnh D2 ở mức độ liên câu. Tuy nhiên, không phải tất cả sự cùng xuất hiện của hóa chất và bệnh tật đều được coi là một mối quan hệ CID hợp lệ. Ví dụ, không có mối quan hệ giữa C1 và D3 vì khái niệm bệnh lý ***death*** là quá chung chung để phản ánh mối quan hệ CID.

1.1.1 Bài toán dự đoán tương tác thuốc từ văn bản y sinh

Thông thường bài toán dự đoán tác thuốc từ văn bản y sinh được chia làm hai bài toán cụ thể như sau:

- Bài toán 1: Nhận dạng thực thể bệnh lý và thực thể thuốc (Named Entity Recognition – NER)
- Bài toán 2: Trích xuất mối quan hệ bệnh lý do thuốc gây ra (Chemical-Induced Disease - CID)

1.1.2 Bài toán nhận dạng thực thể bệnh lý và thực thể thuốc

Nhận dạng thực thể bệnh lý và thực thể thuốc là một bài toán tiền xử lý thiết yếu trong việc xử lý các văn bản y sinh, và là một bài toán xử lý ngôn ngữ tự nhiên. Việc xác định được chính xác các thực thể trong tài liệu sẽ giúp việc xác định các tính chất hóa học, các đặc tính và các mối quan hệ được nêu ra trong văn bản.

Bảng 1.2 Bảng mô tả đầu vào và đầu ra đối với việc nhận dạng thực thể bệnh lý và thực thể thuốc

Đầu vào	Đầu ra	
Possible intramuscular **midazolam** associated **cardiorespiratory arrest** and **death** …	midazolam cardiorespiratory arrest death	Chemical Disease Disease

Theo ví dụ ở Bảng 1.2 trên, từ dữ liệu đầu vào chúng ta xử lý tách ra được các thực thể liên quan tới thuốc và bệnh như sau: midazolam (thuốc), cardiorespiratory arrest (bệnh lý), death (bệnh lý).

Trong nội dung của luận văn này tác giả không đi chi tiết vào bài toán nhận diện thực thể mà chỉ đi chi tiết vào bài toán trích xuất mối quan hệ CID giữa bệnh lý – thuốc. Các thực thể thuốc và thực thể bệnh lý sẽ được sử dụng sẵn từ dữ liệu đầu vào là bộ dữ liệu BioCreative V CDR sẽ được đề cập chi tiết ở phần 4.3.

1.1.3 Bài toán trích xuất mối quan hệ giữa bệnh – thuốc

Bài toán trích xuất mối quan hệ giữa bệnh lý do thuốc gây ra chính là việc xác định một loại bệnh lý xảy ra khi người dùng sử dụng một loại thuốc nào đó được viết trong văn bản y sinh.

Mô tả cụ thể về bài toán chúng ta có thể xem trong ví dụ sau:

Bảng 1.3 Bảng mô tả đầu vào và đầu ra của việc trích xuất mối quan hệ giữa thuốc và bệnh

Đầu vào	Đầu ra
…Possible intramuscular **midazolam**[D008874]-associated **cardiorespiratory arrest**[D006323] and **death**[D003643]…	cardiorespiratory arrest (chứng ngừng tim) và midazolam (thuốc): Có quan hệ

Như ở trong ví dụ ở Bảng 1.3 trên, từ đầu vào của bài toán là một đoạn văn bản y sinh có chứa các loại bệnh lý và thuốc (đã được nhận diện ở bài toán nhận diện thực thể), sau khi trích xuất mối quan hệ giữa thuốc và bệnh lý chúng ta có được một cặp thuốc và bệnh lý có quan hệ với nhau (midazolam + cardiorespiratory arrest).

Trích xuất quan hệ CID thường được xây dựng như một bài toán phân loại nhị phân. Cho trước một văn bản y sinh bao gồm nhiều câu, một danh sách các thực thể được đề cập trong văn bản và một danh sách các cặp định danh của hóa chất và bệnh tật, chúng ta phải trả lời xem liệu giữa một cặp định danh hóa chất và bệnh bất kỳ có sự tương tác **hóa chất gây ra bệnh** hay không.

Bài toán trích xuất quan hệ CID đặt ra nhiều thách thức cần phải giải quyết. Trước hết, các tương tác giữa thuốc và bệnh được gán nhãn ở mức định danh thay vì ở mức đề cập. Điều này có nghĩa là, một thực thể có thể xuất hiện nhiều lần (đề cập) và ở các câu khác nhau trong văn bản. Như trong ví dụ ở trên, thực thể thuốc midazolam được đề cập đến bốn lần, trong khi đó thực thể bệnh chất cardiorespiratory arrest xuất hiện hai lần và nằm ở các câu khác nhau trong văn bản. Như vậy sự tương tác giữa thực thể trong nhiệm vụ trích xuất quan hệ CID được thể hiện vượt ra ranh giới của câu, các tương tác này được luận văn gọi là các quan hệ liên câu, trong khi đó tương tác giữa các thực thể nằm trong cùng một câu được luận văn gọi là các quan hệ nội câu. Việc nhận biết các quan hệ liên câu thường phức tạp hơn nhiều so với các quan hệ nội câu.

1.2 Mục tiêu của luận văn

Luận văn đề xuất giải quyết bài toán dự đoán tương tác bệnh - thuốc từ các văn bản y sinh bằng một mô hình nơ-ron tích chập cho phép tận dụng được thông tin phụ

thuộc toàn cục trong cả đoạn văn bản. Mô hình nơ-ron tích chập cho phép trích xuất các đặc trưng trong câu kết hợp với khả năng của mạng LSTM để phát hiện các phụ thuộc xa ở mức liên câu trong văn bản.

1.3 Cấu trúc của luận văn

Dựa trên mục tiêu cụ thể đã trình bày trong phần trước, luận văn được tổ chức thành bốn chương với các nội dung cụ thể như sau:

Chương 1: Giới thiệu chung

- Giới thiệu chung về bối cảnh của hướng nghiên cứu trên thế giới, tại sao hướng nghiên cứu này lại quan trọng và cần thiết. Cơ sở khoa học để thực hiện đề tài dựa trên mạng nơ-ron tích chập.

Chương 2: Các phương pháp liên quan

- Giới thiệu cơ bản về mạng nơ-ron tích chập, nền tảng khoa học cho việc thực hiện mục tiêu của đề tài.

Chương 3: Mô hình đề xuất

- Mô tả nguồn cơ sở dữ liệu cho việc thực hiện đề tài. Mô hình, phương pháp đề xuất để thực hiện mục tiêu của đề tài.

Chương 4: Kết quả thực nghiệm và kết luận

- Đưa ra những kết quả thu được từ việc thực hiện đề tài. Từ kết quả đó, đưa ra những thảo luận, đánh giá.
- Tóm lược lại mục tiêu của đề tài, các cơ sở khoa học, phương pháp thực hiện, những kết quả đã đạt được trong đề tài.
- Mở ra hướng nghiên cứu tiếp tục trong tương lai

Chương 2: CÁC PHƯƠNG PHÁP LIÊN QUAN

Nội dung chương này trình bày cơ bản về học sâu và mạng nơ-ron nhân tạo, các mạng nơ-ron được áp dụng phổ biến, có hiệu quả tốt trong bài toán của lĩnh vự xử lý ngôn ngữ tự nhiên như mạng nơ-ron hồi quy RRN và LSTM, mạng nơ-ron tích chập CNN. Đồng thời chương này trình bày tổng quan về một số phương pháp biểu diễn từ theo ngữ cảnh cũng như các phương pháp liên quan trong trích xuất quan hệ CID, làm nền tảng khoa học cho việc thực hiện mục tiêu của luận văn.

2.1 Học sâu và mạng nơ-ron

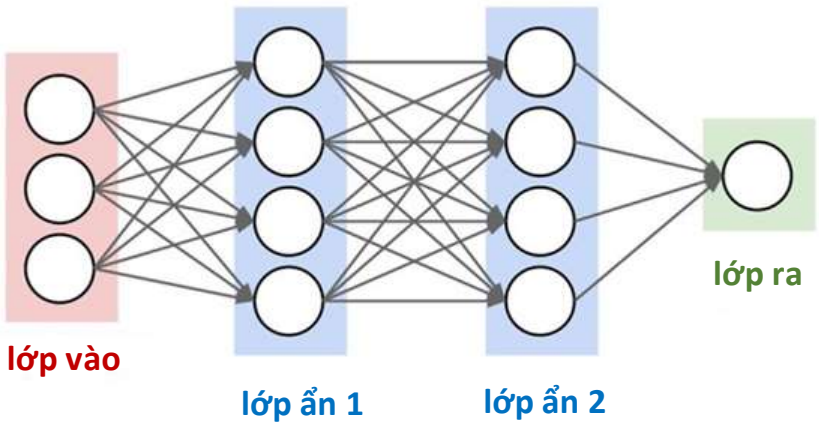
2.1.1 Trí tuệ nhân tạo

Trí tuệ nhân tạo (Artificial Intelligence - AI) đề cập đến trí thông minh do máy móc đạt được, trái ngược với trí thông minh tự nhiên của con người. Trí tuệ nhân tạo được con người thiết kế ra để giải quyết một số công việc cụ thể.

Học máy (Machine Learning) là một tập con các phương thức bên trong AI, đặc biệt đề cập đến các thuật toán và mô hình số được thiết lập để phân tích dữ liệu và lấy hoặc học khả năng ra quyết định để đạt được một số nhiệm vụ nhất định. Mục tiêu của nó là phát hiện ra các mô hình ẩn trong dữ liệu dưới các ràng buộc dữ liệu, ví dụ như kích thước dữ liệu và chất lượng, cho phép giải quyết được các vấn đề đang được quan tâm.

2.1.2 Mạng nơ-ron nhân tạo

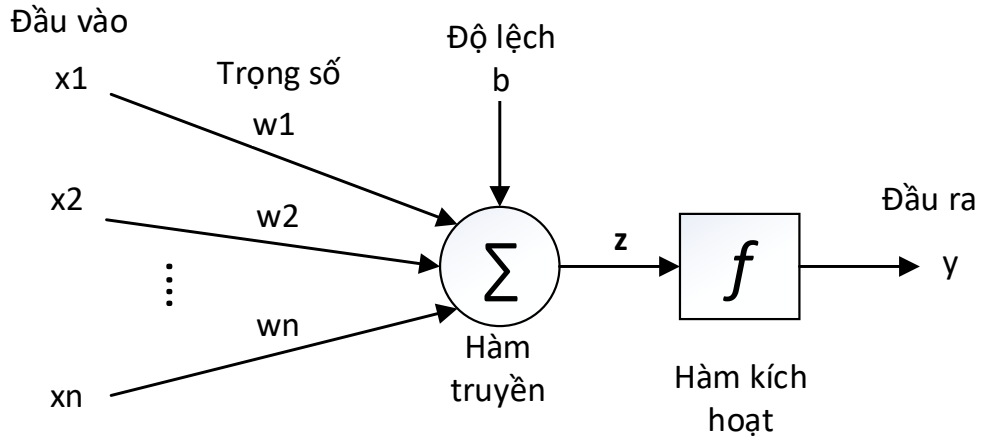
Trong Học Sâu, mạng nơ-ron nhân tạo hay còn gọi là các Multilayer Perceptron là các kiểu kiến trúc đơn giản nhất. Mỗi mạng nơ-ron nhân tạo bao gồm 3 thành phần chính: lớp vào, các lớp ẩn và lớp ra. Hình 2.1 mô tả kiến trúc một mô hình nơ-ron nhân tạo.



Hình 2.1 Minh họa mạng nơ-ron nhân tạo

Trong đó, các lớp ẩn của mô hình nơ-ron nhân tạo được tạo nên từ một hoặc nhiều đơn vị được gọi là các tế bào (perceptron). Các tế bào tại một lớp thực thiện tổ hợp đầu

ra của lớp trước đó thành đầu vào của lớp đứng sau hoặc là đầu ra của mạng. Cụ thể hơn, mỗi tế bào sẽ có các liên kết tới lớp đứng trước được gọi là trọng số (weights). Quá trình tính toán của mỗi tế báo sẽ diễn ra như sau: Mỗi đầu ra ở lớp phía trước sẽ được nhân với giá trị trọng số tương ứng với chúng và cộng tổng lại. Giá trị này sau đó được cộng thêm với một đại lượng có tên là độ lệch (bias) và đưa qua một hàm kích hoạt (activation function). Giá trị đầu ra sau đó được sử dụng như là đầu vào của lớp đứng phía sau, hoặc nếu lớp hiện tại là lớp cuối cùng trong mạng thì nó sẽ được sử dụng như là đầu ra của mạng. Hình 2.2 mô tả quá trình tính toán của một tế bào.



Hình 2.2 Minh họa quá trình tính toán của một tế bào.

Các hàm kích hoạt trong mạng nơ-ron nhân tạo thông thường là các hàm phi tuyến tính. Điều này là cần thiết để mạng có thể hoặc được những mối quan hệ phức tạp giữa đầu vào và đầu ra. Một số hàm kích hoạt thường được sử dụng có thể kể đến như sigmoid, tanh hay rectified linear unit (ReLU). Việc chọn các hàm kích hoạt sao cho phù hợp phụ thuộc vào quá trình thực nghiệm trên từng nhiệm vụ cụ thể.

Trong thực hành, để cho thuận tiện cũng như tăng tốc quá trình tính toán, toàn bộ trọng số và độ lệch của tương ứng với mỗi lớp sẽ được tổ chức dưới dạng ma trận **W** và vector **b** với mỗi hàng là các giá trị trọng số và độ lệch tương ứng của mỗi tế bào, điều này giúp đơn giản hóa quá thao tác tính toán của mạng thành các phép nhân ma trận hoặc cộng các vector. Cụ thể quá trình tính toán của một lớp có thể được viết lại như sau.

$$z = \boldsymbol{W}x + \boldsymbol{b}, \qquad (2.1)$$
$$y = f(z)$$

Trong đó x là vector đầu vào, z là biến trung gian lưu giá trị tổng của đầu vào với trọng số tương ứng, a là vector đầu ra và f là hàm kích hoạt phi tuyến.

Công thức tính toán trên chỉ áp dụng cho một lớp trong mạng nơ-ron nhân tạo. Trong thực tế, các mạng nơ-ron nhân tạo có thể có nhiều hơn hai lớp. Luận văn tiến hành tổng quát hóa công thức tính toán cho mạng nơ-ron nhân tạo nhiều lớp như sau.

$$z^l = \boldsymbol{W}^l a^{[l-1]} + \boldsymbol{b}^l \qquad (2.2)$$

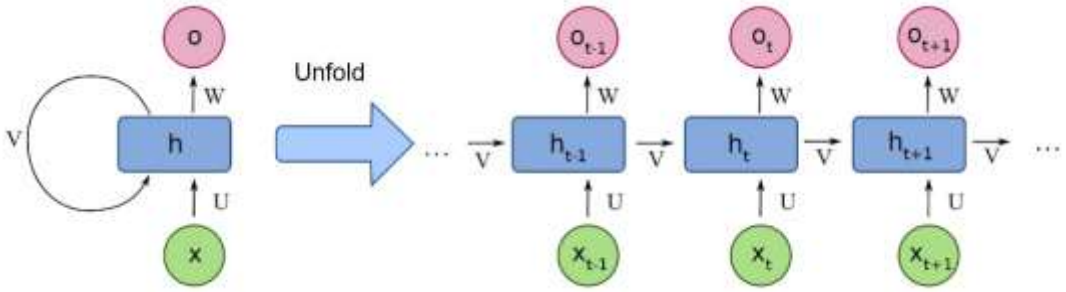
$$y^l = f(z^{[l]})$$

Với $y^{[l-1]}$ là đầu ra của lớp thứ $l-1$, $z^{[l]}$ là vector trung gian và $y^{[l]}$ là vector đầu ra của lớp thứ l. $W^{[l]}$ và $b^{[l]}$ là trọng số và độ lệch tương ứng của lớp thứ l. Để cho thuận tiện về mặt kí hiệu, thông thường chúng ta coi vector đầu vào x là vector $y^{[0]}$. Với mạng nơ-ron có 1 lớp vào, L lớp ẩn và 1 lớp ra thì vector đầu ra $y^{[L+1]}$ sẽ tương ứng với đầu ra của mạng.

2.2 Mạng nơ-ron hồi quy RNN và LSTM

Khi phải giải quyết các bài toán mà dữ liệu đầu vào có dạng là chuỗi như các văn bản, chuỗi thời gian, hay tín hiệu âm thanh, .. các mạng nơ-ron nhân tạo không còn phù hợp nữa. Thứ nhất, các mạng nơ-ron nhân tạo chỉ có thể nhận vào đầu vào với kích thước cố định, điều này không phù hợp nếu chúng ta cần xử lý những chuỗi dữ liệu có độ dài khác nhau. Thứ hai, mạng nơ-ron nhân tạo không cho phép phần tử ở các vị trí khác nhau trên chuỗi đầu vào chia sẻ những đặc trưng mà chúng học được, tuy nhiên điều này lại rất cần thiết trong một số nhiệm vụ của xử lý ngôn ngữ tự nhiên như mô hình hóa ngôn ngữ hay nhận diện tên thực thể. Để giải quyết những vấn đề nêu trên, mạng nơ-ron hồi quy RNN đã được đề xuất.

Mạng RNN thực hiện tính toán trên chuỗi đầu vào một cách tuần tự từ trái qua phải hoặc từ phải qua trái, chính vì vậy mà RNN có khả năng ghi nhớ những nó đã xử lý và sử dụng thông tin đó cho việc dự đoán tương lai. Cụ thể, các mạng nơ-ron hồi quy nhận hai đầu vào, một đầu vào biểu diễn cho thông tin mốc thời gian hiện tại và phần còn lại là các thông tin mạng đã nhận được trong quá khứ. Hình 2.3 mô ta kiến trúc một mạng RNN.



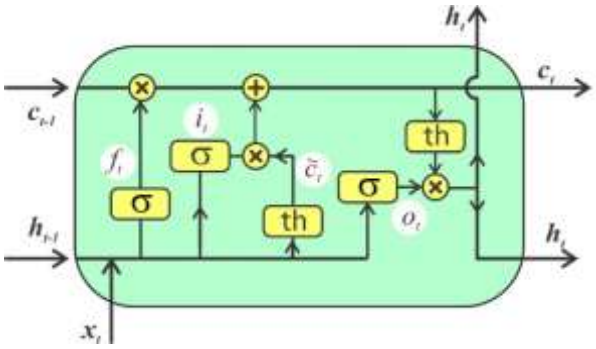
Hình 2.3 Mô tả một mạng nơ-ron hồi quy RNN.

Giả sử chúng ta có một chuỗi đầu vào **x** = {x_1, x_2, ...x_N }, tại mốc thời gian t, RNN tính toán đầu ra của thời điểm hiện tại sử dụng đầu vào x_t và trạng thái ẩn của thời điểm trước đó $ht-1$. Quá trình tính toán được mô tả bằng công thức sau.

$$h_t = \tanh\ (Ux_t + Vh_{t-1} + b) \qquad (2.3)$$
$$o_t = \boldsymbol{W}_o h_t + \boldsymbol{b}_o$$

Đầu ra của mỗi mốc thời gian *ot* có thể được sử dụng khác nhau tùy thuộc vào mỗi dạng bài toán. Đối với bài toán dự đoán từ tiếp theo (next word prediction), đầu ra của mốc thời gian cuối cùng *oN* có thể được xem như biểu diễn của toàn bộ chuỗi và được sử dụng để dự đoán xem từ xuất hiện tiếp theo sẽ là từ nào. Còn trong bài toán gãn nhãn từ loại (POS tagging) thì đầu ra của từng mốc thời gian tương ứng sẽ được sử dụng để dự đoán xem từ hiện tại thuộc loại từ gì, ví dụ: danh từ, động từ, tính từ, ...

Tuy nhiên, các mô hình RNN lại gặp phải vấn đề tiêu biến và bùng nổ đạo hàm (vanishing and exploding gradident) khi sử dụng với các chuỗi có độ dài lớn. Điều này khiến mô hình nơ-ron hồi quy thông thường rất khó có thể học cũng như bào toàn được các phụ thuộc xa. Và để hạn chế điều này, một biến thể khác của mạng nơ-ron hồi quy RNN đó chính là mạng LSTM đã được đề xuất. Kiến trúc mạng LSTM bao gồm nhiều cổng (cổng quên, cổng vào và cổng ra), giúp kiểm soát và lưu trữ các thông tin cần thiết trong quá trình học. Hình 2.4 minh họa mô hình LSTM.



Ghi chú
$\boldsymbol{x}_t$ *đầu vào*
f_t *cổng quên*
i_t *cổng vào*
$\tilde{c}_t$ *ô cập nhật*
c_t *ô trạng thái*
$\boldsymbol{h}_t$ *đầu ra*

Hình 2.4 Minh họa kiến trúc của mạng LSTM

Ở thời điểm *t*, mạng LSTM sử dụng đầu vào hiện tại x_t, trạng thái ẩn ở trước đó h_{t-1} và trạng thái bộ nhớ ở mốc thời gian trước c_{t-1} để tính toán. Cổng quên được sử dụng để lọc các thông tin không còn cần thiết nữa đối với mạng LSTM. Cổng này sử dụng đầu vào x_t và trạng thái ẩn trước đó h_{t-1} cùng các ma trận trọng số tương ứng $\mathbf{V_f}$, $\mathbf{U_f}$, $\mathbf{b}_f$ với hàm kích hoạt là sigmoid. Hàm sigmoid đưa khoảng giá trị của *ft* về khoảng [0, 1], biểu diễn lượng thông tin được giữ lại hay bỏ đi. Quá trình tính toán diễn ra như sau.

$$f_t = \sigma\left(\boldsymbol{U}_{\mathrm{f}} x_t + \boldsymbol{V}_{\mathrm{f}} h_{t-1} + \boldsymbol{b}_f\right) \qquad (2.4)$$

Cổng vào được sử dụng để điều khiển thông tin nhận được từ đầu vào x_t và trạng thái ẩn trước đó h_{t-1}. Tương tự như cổng quên, cổng vào cũng sử dụng x_t và h_{t-1}, các ma trận trọng số tương ứng là $\mathbf{V_i}$, $\mathbf{U_i}$, $\mathbf{b_i}$ cùng hàm kích hoạt là sigmoid. Công thức tính toán như sau.

$$i_t = \sigma\left(\boldsymbol{U}_{\mathrm{i}} x_t + \boldsymbol{V}_{\mathrm{i}} h_{t-1} + \boldsymbol{b}_i\right) \quad (2.5)$$

Giá trị cổng quên f_t và cổng vào i_t được sử dụng để tính toán giá trị trạng thái bộ nhớ ở thời điểm hiện tại c_t. Trong đó cổng quên f_t kiểm soát lượng thông tin được giữ lại hay bỏ đi từ trạng thái bộ nhớ ở thời điểm trước c_{t-1} còn cổng vào i_t điều khiển lượng thông tin nhận được các đầu vào ở thời điểm hiện tại x_t và h_{t-1}. $\mathbf{V_c}$, $\mathbf{U_c}$, $\mathbf{b_c}$ là trọng số tương ứng dùng để tính toán ct, $\circ$ biểu thị cho phép nhân Hadamard. Quá trình tính toán diễn ra như sau.

$$c_t = f_t \circ c_{t-1} + i_t \circ tanh\left(\boldsymbol{U}_{\mathrm{c}} x_t + \boldsymbol{V}_{\mathrm{c}} h_{t-1} + \boldsymbol{b}_c\right) \quad (2.6)$$

Cổng ra làm nhiệm vụ trích lọc ra các thông tin cần thiết từ trạng thái bộ nhớ c_t. Cổng ra sử dụng đầu vào x_t, trạng thái bộ nhớ trước đó h_{t-1} và các ma trận trọng số tương ứng là $\mathbf{V_o}$, $\mathbf{U_o}$, $\mathbf{b_o}$ cùng hàm kích hoạt là sigmoid. Cuối cùng, trạng thái ẩn ở thời điểm hiện tại h_t sẽ được tính toán từ giá trị cổng ra o_t và trạng thái bộ nhớ c_t.

$$o_t = \sigma\left(\boldsymbol{U}_{\mathrm{o}} x_t + \boldsymbol{V}_{\mathrm{o}} h_{t-1} + \boldsymbol{b}_o\right) \quad (2.7)$$

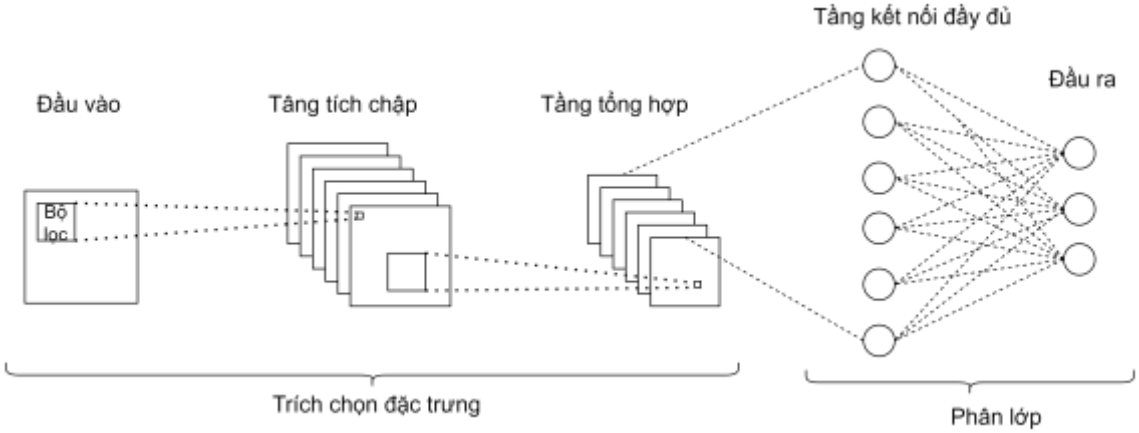
$$h_t = o_t \circ \tanh(c_t)$$

Giá trị trạng thái ẩn h_t và trạng thái bộ nhớ c_t được tiếp tục sử dụng để tính toán các biểu diễn ở các mốc thời gian sau đó. Các cổng kiểm soát thông tin trong mô hình LSTM giúp hạn chế rất tốt nhược điểm của mạng RNN thông thường. Tuy nhiên, nếu chuỗi đầu vào có độ dài rất lớn (200-300 từ) thì ngay cả mô hình LSTM cũng sẽ khó có thể học cũng như bảo toàn các phụ thuộc xa do vấn đề tiêu biến và bủng nổ đạo hàm.

2.3 Mạng nơ-ron tích chập CNN

Mạng nơ-ron tích chập CNN là một trong những mô hình học sâu phổ biến nhất và có ảnh hưởng nhiều nhất trong lĩnh vực thị giác máy (computer vision). Mạng tích chập CNN ban đầu thường được dùng để giải quyết vấn đề xử lý hình ảnh (như nhận dạng ảnh, phân tích video …). Sau đó, mạng tích chập bắt đầu được ứng dụng và có hiệu quả tốt trong bài toán của lĩnh vự xử lý ngôn ngữ tự nhiên và hầu hết đều giải quyết tốt các bài toán này. Yoon Kim và cộng sự đã đưa ra kiến trúc mạng tích chập ứng dụng trong phần việc phân lớp câu [5].

Hình 2.5 minh họa mạng tích chập truyền thống thường có kiến trúc bao gồm hai phần chính: Phần 1 dùng để trích chọn đặc trưng gồm đầu vào, tầng tích chập (convolution layer), tầng tổng hợp (Pooling layer). Phần 2 dùng để phân lớp dữ liệu gồm có tầng kết nối đầy đủ (fully connected) và đầu ra.



Hình 2.5 Kiến trúc chung của một mạng tích chập CNN truyền thống

Tầng tích chập: Phép tích chập là phép toán tuyến tính thực hiện trên 2 đồ hàm số để đo lường sự chồng chéo của chúng. Với f và g là hàm số phức trong không gian R^d, phép nhân tích chập của f và g được biểu diễn:

$$(f * g)(x) = \int_{R^d} f(x-y)\, g(y) dy \tag{2.8}$$

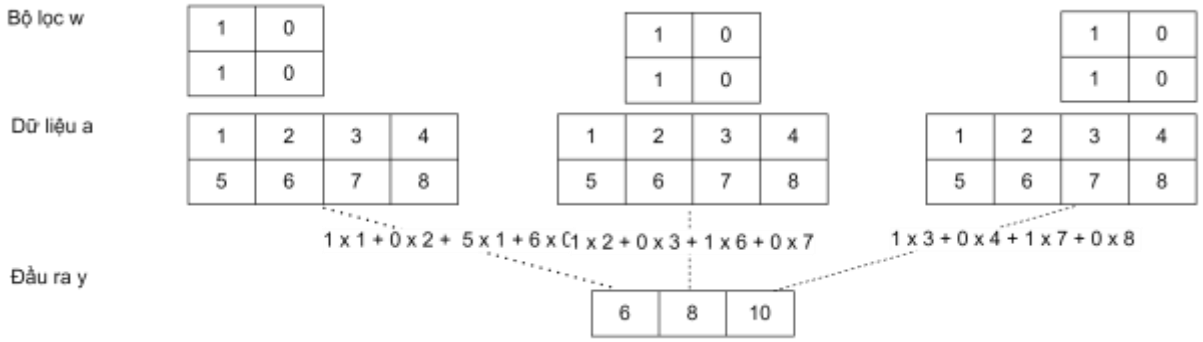
Phép nhân tích chập được định nghĩa là phép toán trên không gian khả tích của các hàm tuyến tính nên có đầy đủ các tính chất giao hoán, kết hợp và phân phối. Trong mạng tích chập, phép nhân chập được biểu diễn khác một chút. Cho tín hiệu đầu vào và bộ lọc lần lượt là các vector $a \in R^N$ và $w \in R^f$, khi đó đầu ra là vector y được tính bằng:

$$y_n = \sum_{i=0}^{f-1} a_{n+i}\, w_i \tag{2.9}$$

Với n thoả mãn $0 \leq n < N - f + 1$. Vậy $y \in R^{N-f+1}$.

Công thức được diễn giải bằng lời qua các bước sau:

1) Đặt vị trí bộ lọc w vào vị trí ứng với f phần tử đầu tiên của đầu vào a
2) Nhân từng phần tử tương ứng của f và a rồi cộng các phần tử tương ứng lại để được phần tử tương ứng của y
3) Trượt bộ lọc f một bước (hoặc nhiều bước), nếu phần tử cuối cùng của bộ lọc không vượt ra ngoài phần tử cuối cùng của đầu vào a thì lặp lại bước 2.



Hình 2.6 Minh họa phép tích chập

Hình 2.6 trình bày một ví dụ về phép tích chập. Nhận thấy rằng đầu ra có kích thước bé hơn kích thước đầu vào nên nếu tiếp tục sử dụng đầu ra để làm đầu vào cho tầng tích chập tiếp theo thì dữ liệu sẽ bị giảm kích thước. Trong trường hợp không muốn đầu ra không bị giảm kích thước hoặc muốn thay đổi kích thước đầu ra lớn hơn, ta sẽ giả sử rằng đầu vào có kích thước lớn hơn kích thước đầu vào thực tế. Kỹ thuật này được gọi là thêm lề (padding). Kỹ thuật thêm lề cũng được sử dụng trong trường hợp dữ liệu đầu vào có kích thước không đều nhau.

Bộ lọc w

1	0
1	0

Dữ liệu a

0	0	0	0	0
0	1	2	3	4
0	5	6	7	8

Đầu ra y

0	1	2	3
0	6	8	10

Hình 2.7 Minh họa kỹ thuật thêm lề trong phép tích chập

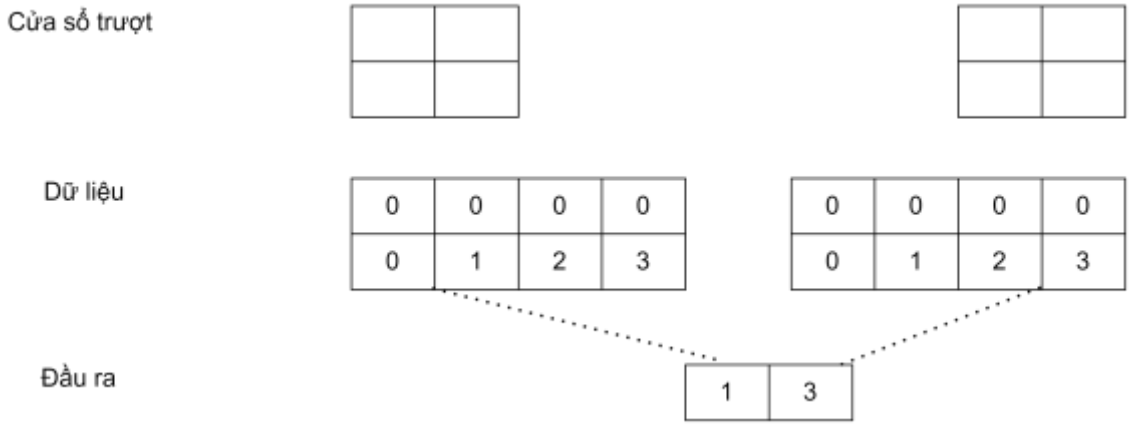
Trong ví dụ trên Hình 2.7, ta sử dụng kỹ thuật thêm lề bằng cách thêm một hàng và một cột có giá trị đều bằng 0. Sau đó nhân tích chập thì thu được đầu ra có kích thước 2x3 bằng với kích thước đầu vào thực tế.

Bộ lọc *w* được dịch sang phải một ô so với dữ liệu đầu vào. Số ô được dịch này gọi là bước trượt (stride). Phụ thuộc vào mục đích sử dụng mà bước trượt có các giá trị dương khác nhau.

***Tầng tổng hợp (pooling layer)*:** Tầng tổng hợp được xem là một phép giảm kích thước dữ liệu mà vẫn giữ được đặc trưng quan trọng của dữ liệu. Việc giảm kích thước dữ liệu giúp giảm số lượng tham số, tránh quá khớp (overfit), và giảm thời gian tính toán.

Tầng tổng hợp cũng sử dụng cửa sổ trượt giống như bộ lọc của tầng tích chập. Điểm khác biệt là tầng tổng hợp sử dụng phép tổng hợp (chọn dữ liệu đặc trưng) thay vì phép nhân chập ở tầng tích chập.

Có hai phương thức tổng hợp thường dùng nhất là gộp cực đại (Max pooling) và gộp trung bình (Average Pooling).

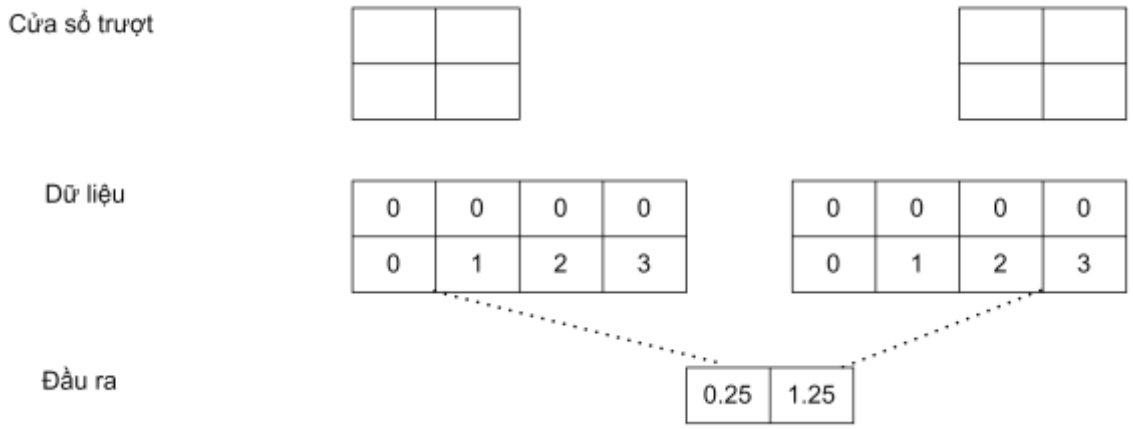


Hình 2.8 Minh họa về phép gộp cực đại (max pooling)

Ở Hình 2.8, phép gộp cực đại (Max pooling) được diễn ra gồm 3 bước:

1) Đặt cửa sổ trượt vào vị trí tương ứng với phần tử đầu tiên của dữ liệu.
2) Thực hiện tổng hợp đối với vùng dữ liệu tương ứng với cửa sổ trượt. Trong trường hợp này là chọn dữ liệu lớn nhất.
3) Trượt cửa sổ một bước sang vùng dữ liệu tiếp theo cần tổng hợp, nếu cửa sổ vẫn nằm trong vùng dữ liệu đầu vào thì lặp lại bước 2.

Và sau khi tổng hợp, dữ liệu đầu ra sẽ là 1 và 3, đại diện cho hai vùng dữ liệu được tổng hợp theo phương thức gộp cực đại (Max pooling).



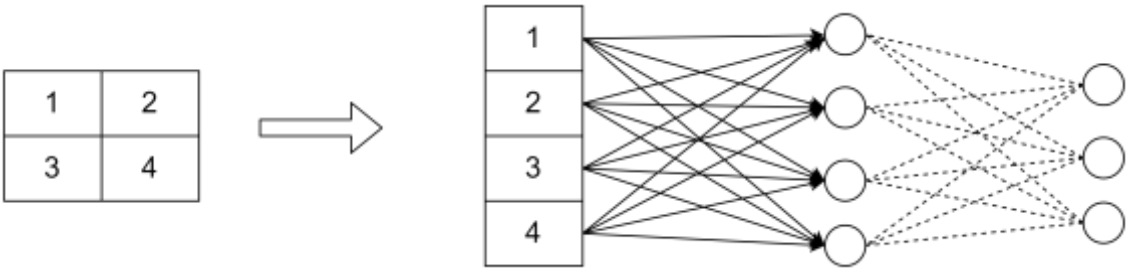
Hình 2.9 Minh họa về phép gộp trung bình (Average Pooling)

Ở phép gộp trung bình (Average Pooling), các bước trượt giống phép gộp cực đại (Max Pooling), điểm khác biệt là cách tổng hợp. Thay vì chọn giá trị lớn nhất trong vùng dữ liệu như phép gộp cực đại (Max Pooling), phép gộp trung bình (Average Pooling) thực hiện phép lấy trung bình đối với vùng tổng hợp:

Ở ví dụ trong Hình 2.9, có hai lần tổng hợp. Ở lần đầu tiên, giá trị tổng hợp sẽ bằng trung bình cộng của bốn giá trị dữ liệu tương ứng với cửa sổ trượt ở vị trí thứ nhất: $giá\ tổng\ hợp = \frac{0+0+0+1}{4} = 0.25$

Lần tổng hợp thứ hai: $giá\ trị\ tổng\ hợp = \frac{0+0+2+3}{4} = 1.25.$

Tầng kết nối đầy đủ (fully connected layer):



Hình 2.10 Minh họa về tầng kết nối đầy đủ trong mạng nơ-ron tích chập CNN

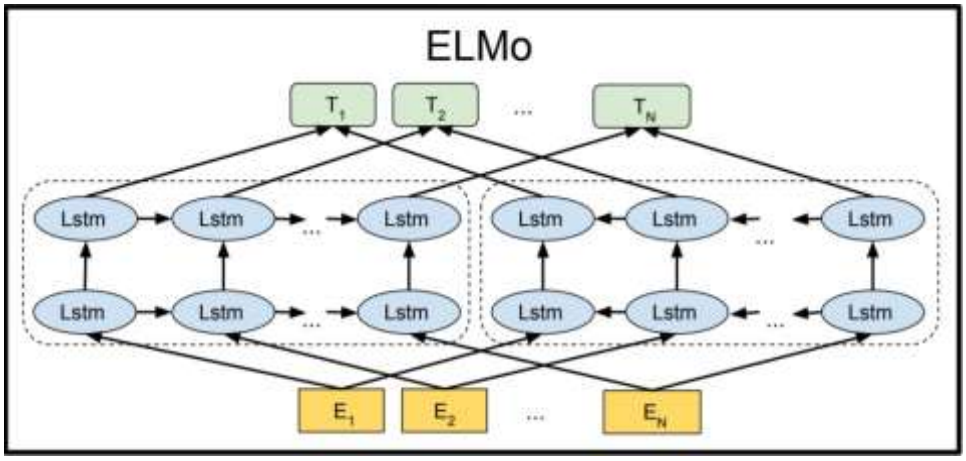
Ở Hình 2.10, sau khi đã giảm kích thước đến một mức độ hợp lý, ma trận cần được trải phẳng (flatten) thành một vector và sử dụng các kết nối hoàn toàn giữa các tầng. Quá trình này sẽ diễn ra cuối mạng CNN và sử dụng hàm kích hoạt là ReLU. Tầng kết nối đầy đủ cuối cùng (fully connected layer) sẽ có số lượng đơn vị bằng với số lớp và áp dụng hàm kích hoạt là softmax nhằm mục đích tính phân phối xác xuất.

2.4 Biểu diễn từ theo ngữ cảnh

Các phương pháp truyền thống để sinh biểu diễn cho từ [6] [7] chỉ có thể tạo ra các vector cố định và độc lập với ngữ cảnh, điều này giới hạn hiệu suất của chúng trong

các nhiệm vụ của xử lý ngôn ngữ tự nhiên. Nhiều nghiên cứu gần đây đã đề xuất các mô hình ngôn ngữ hiện đại cho phép học các biểu diễn của từ theo ngữ cảnh bằng cách sử dụng nhiều loại nhiệm vụ mô hình hóa ngôn ngữ khác nhau.

Cụ thể hơn, chọn ELMo [8] làm ví dụ, mô hình này học các biểu diễn có ngữ cảnh của từ bằng cách tiền huấn luyện một mô hình nơ-ron hồi quy, có thể là LSTM trên một bộ dữ liệu có kích thước lớn không được gán nhãn với nhiệm vụ chính là **dự đoán từ tiếp theo** (ALM). Sau khi đã huấn luyện xong, trạng thái ẩn của mô hình LSTM sẽ được sử dụng như như biểu diễn của mỗi từ tương ứng với chúng. Và bởi vì ELMo tính toán biểu diễn của từ bằng mạng LSTM, cho nên cùng một từ nhưng nằm trong các câu khác nhau sẽ sinh ra các biểu diễn khác nhau hay còn gọi là các biểu diễn của từ theo ngữ cảnh. Một ưu điểm nữa của mô hình này đó là liệu huấn luyện không cần thiết phải là dữ liệu có nhãn, cho nên ELMo hoàn toán có thể được huấn luyện trên các tập văn bản có kích thước lớn tùy ý. Tiền huấn luyện ELMo trên một tập dữ liệu lớn cùng nhiệm vụ mô hình hóa ngôn ngữ có thể tạo ra các biểu diễn phong phú và tổng quát, thích hợp cho các ứng dụng hạ nguồn. Hình 2.11 mô tả kiến trúc của mô hình nhúng từ theo ngữ cảnh ELMo.



Hình 2.11 Minh họa kiến trúc của mô hình Embedding from Language Model (ELMo).

Tuy nhiên, các biểu diễn có ngữ cảnh sinh ra từ những mô hình ngôn ngữ nêu trên có thể không thích ứng tốt với các bài toán y sinh học bởi vì sự không khớp giữa dữ liệu tiền huấn luyện và dữ liệu của nhiệm vụ đích. Nhiều công trình gần đây đã mở rộng sự thành công của các mô hình ngôn ngữ mạnh mẽ sang lĩnh vực y sinh học. [9] đề xuất BioElmo, một phiên bản y sinh học của Elmo [8] được tiền huấn luyện trên 10 triệu tóm tắt (2.46 tỷ từ) trích ra từ kho dữ liệu PubMed. Hiệu suất của mô hình sử dụng BioELMo vượt trội hoàn toàn so với ELMo khi đo lường trên hai bộ dữ liệu y sinh tiêu chuẩn.

2.5 Các phương pháp liên quan cho trích xuất quan hệ bệnh-thuốc

2.5.1 Các phương pháp dựa trên học máy

Các phương pháp ban đầu cho trích xuất quan hệ bệnh do hóa chất gây ra (CID) thường tập trung vào việc xây dựng một cách thủ công các tập đặc trưng cú pháp và ngữ nghĩa như là đầu vào của một số thuật toán học máy. [10] thiết kế một tập đặc trưng phong phú bao gồm đặc trưng ngữ cảnh, thông tin về thực thể, các thông tin từ miền tri thức phụ trợ và sử dụng chúng như là đầu vào cho thuật toán Support Vector Machine (SVM). [11] đã áp dụng mô hình Maximum Entropy (ME) với nhiều loại đặc trưng ngôn ngữ để trích xuất CID. Họ cũng phát triển nhiều đặc trưng phụ thuộc khác như là đường đi phụ thuộc ngắn nhất giữa các thực thể (SDP), tuy nhiên, lại chỉ áp dụng cho quan hệ nằm trong cùng một câu. [12] mở rộng đường đi phụ thuộc ngắn nhất (SDP) trong cùng một câu thành phiên bản có thể áp dụng chéo giữa các câu để tăng khả năng dự đoán của mô hình cho các quan hệ liên câu. Các phương pháp nêu trên đã đạt được nhiều kết quả đáng khích lệ cho nhiệm vụ trích xuất quan hệ CID. Tuy nhiên, việc thiết kế thủ công các tập đặc trưng ngôn ngữ học như vậy lại rất tốn thời gian và công sức.

2.5.2 Các phương pháp dựa trên học sâu

Đến nay, các mô hình học sâu đã và đang chiếm ưu thế trong nhiều lĩnh vực của xử lý ngôn ngữ tự nhiên. Nhiều công trình đã được đề xuất để nghiên cứu độ hiệu quả của mô hình CNN và mô hình LSTM cho bài toán trích xuất quan hệ CID. [10] đạt được kết quả tốt bằng việc sử dụng kết hợp hai mô hình LSTM và CNN cùng với các phương pháp nhúng từ truyền thống và một vài đặc trưng ngôn ngữ khác. [2] giới thiệu một mô hình CNN dùng để trích xuất đặc trưng từ đường đi phụ thuộc ngắn nhất giữa hai thực thể cho dự đoán quan hệ trong cùng một câu. [13] cải tiến mô hình CNN cho trích xuất quan CID bằng phương pháp nhúng từ mức ký tự. [14] đề xuất cơ chế tập trung (multi-head self-attention), cơ chế này cho phép mô hình có thể học các thông tin quan trọng từ các chiều không gian con biểu diễn khác nhau.

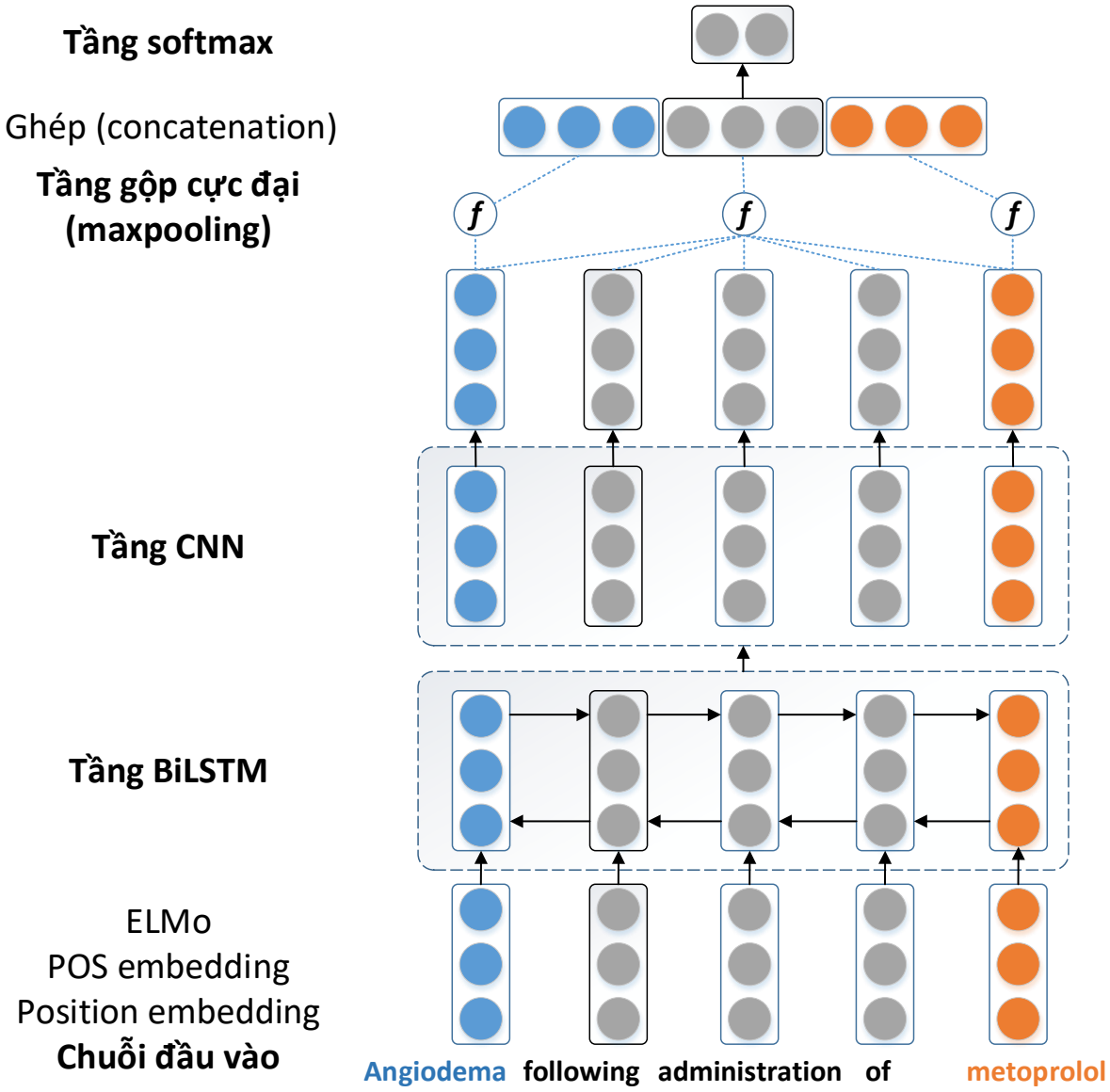
Gần đây, một số nghiên cứu tập trung vào xây dựng một đồ thị phụ thuộc thống nhất cho mỗi tài liệu và áp dụng một cơ chế lặp để tăng cường các biểu diễn của mỗi đỉnh trong đồ thị. [15] đề xuất sử dụng một mô hình mạng đồ thị tích chập (Graph Convolutional Network - GCN) trên một cấu trúc đồ thị mức tài liệu mà các cạnh của đồ thị là liên kết phụ thuộc và liên kết định danh. Công trình của họ đã thu được kết quả đầy hứa hẹn trong việc trích xuất các tương tác liên câu. [4] kết hợp mô hình GCN với cơ chế tập trung Multi-head Self-attention trên đồ thị phụ thuộc mức tài liệu và thu được kết quả tiến tiến nhất cho bài toán trích xuất quan hệ CID.

Từ những phương pháp liên quan nêu trên – đặc biệt là ý tưởng sử dụng mô hình kết hợp giữa mạng nơ-ron tích chập CNN và mạng hồi quy LSTM, luận văn sẽ trình bày mô hình đề xuất cho việc thực hiện mục tiêu của luận văn trong chương 3.

Chương 3: MÔ HÌNH ĐỀ XUẤT

Trong chương này, luận văn sẽ trình bày chi tiết phương pháp đề xuất cho việc giải quyết bài toán trích xuất tương tác bệnh do hóa chất gây ra (CID). Trong phần đầu tiên, khóa luận sẽ đề xuất kiến trúc tổng thể mô hình mạng nơ-ron tích chập CNN kết hợp với mạng nơ-ron hồi quy LSTM để giải quyết bài toán. Ở phần thứ hai, luận văn sẽ mô tả chi tiết từng thành phần chi tiết của kiến trúc đề xuất.

3.1 Mô hình đề xuất



Hình 3.1 Mô hình đề xuất mạng nơ-ron tích chập CNN kết hợp với LSTM

Hình 3.1 trình bày kiến trúc tổng quan của mô hình mà luận văn đề xuất trong giải quyết bài toán trích xuất tương tác bệnh do hóa chất gây ra (CID).

Đầu tiên, ở chuỗi đầu vào, luận văn đề xuất phương pháp tạo ra biểu diễn đầu vào cho mỗi từ trong văn bản bằng cách kết hợp Vector từ theo ngữ cảnh (ELMo), Vector từ loại và Vector Position.

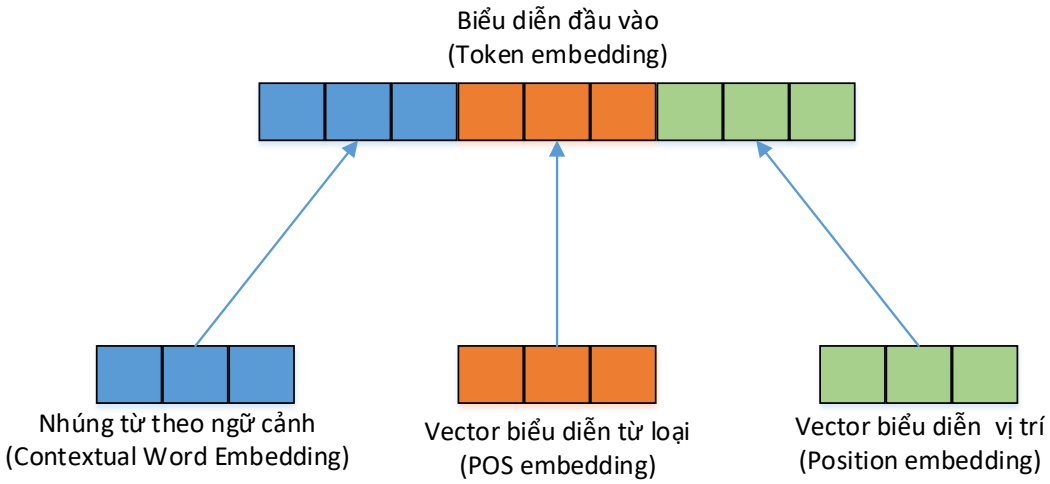
Tiếp đó, biểu diễn đầu vào được đưa qua mạng hồi quy LSTM hai chiều (Bidirectional LSTM) để đồng thời mã hóa được cả thông tin ngữ cảnh từ trái qua phải và từ phải qua trái.

Kết quả của tầng LSTM sẽ được đưa qua mạng tích chập CNN để trích xuất được đặc trưng mong muốn trong quan hệ CID.

Cuối cùng, luận văn đưa ra dự đoán ở mức định danh bằng cách sử dụng hàm gộp cực đại (max pooling) và hàm softmax trên tập dự đoán của tất cả các cặp đề cập.

3.2 Biểu diễn đầu vào

Phần này sẽ mô tả phương pháp tạo ra các vector đầu vào cho mô hình. Hình 3.2 minh họa cách luận văn tạo ra biểu diễn đầu vào cho mỗi từ trong văn bản.



Hình 3.2 Biểu diễn các vector đầu vào

Cụ thể, gọi $x_i \in \mathbb{R}^d$ là biểu diễn của token thứ *i* trong chuỗi đầu vào $w_1, w_2, \dots, w_n$. Mỗi biểu diễn xi sẽ được tạo ra như là một tổ hợp của vector nhúng từ theo ngữ cảnh (contextual word embedding) $e_{w_i} \in \mathbb{R}^{d1}$, và vector biểu diễn cho từ loại (POS embedding) $p_{w_i} \in \mathbb{R}^{d2}$ và vector biểu diễn vị trí (position embedding) $d_{w_i} \in \mathbb{R}^{d3}$, $(d = d_1 + d_2 + d_3)$, $\circ$ là ký hiệu cho phép nối vector.

$$x_i = e_{w_i} \circ p_{w_i} \circ d_{w_i} \quad (3.1)$$

3.2.1 Word embedding – ELMo

Các phương pháp truyền thống để sinh biểu diễn cho từ [6] bỏ qua ngữ nghĩa của các từ trong những ngữ cảnh khác nhau, điều này giới hạn khả năng của những mô hình nhúng từ tĩnh đó trong hiệu suất của nhiều nhiệm vụ xử lý ngôn ngữ tự nhiên. Vì vậy, để tạo ra những biểu diễn từ ngữ có thể thay đổi tùy thuộc vào ngữ cảnh xung quanh nó, luận văn sử dụng phiên bản y sinh của ELMo [9] đã được tiền huấn luyện trên 10 triệu tóm tắt trích ra từ kho dữ liệu Pubmed. Mỗi vector từ theo ngữ cảnh e_{w_i} là liên tục, nằm trong không gian d_1 chiều và được giữ cố định trong quá trình huấn luyện.

3.2.2 POS embedding

Bên cạnh vector biểu diễn từ theo ngữ cảnh, luận văn cũng sử dụng thêm cả thông tin về từ loại (part of speech) ở trong biểu diễn đầu vào. Với mỗi từ trong văn bản đầu vào, luận văn dùng một vector để biểu diễn loại từ tương ứng với từ đó. Vector từ loại p_{w_i} được khởi tạo ngẫu nhiên, nằm trong không gian d_2 chiều và được cập nhật trong quá trình huấn luyện mô hình.

3.2.3 Position embedding

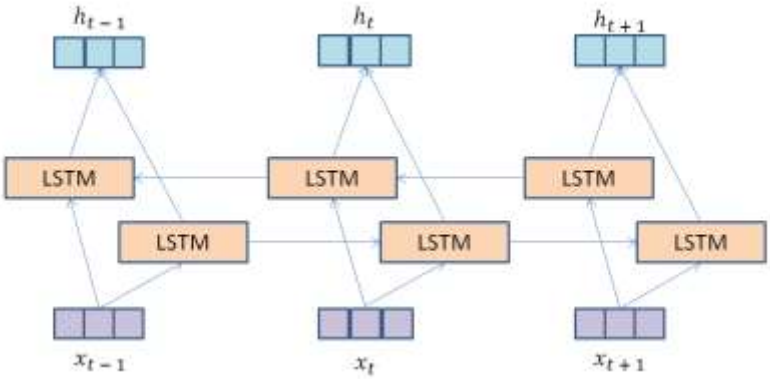
Position embedding là đặc trưng khoảng cách về vị trí của một từ so với các thực thể. Thông tin về vector khoảng cách d_{w_i} nằm trong không gian d_3 sẽ được xử lý trong bước tiền xử lý dữ liệu.

Do việc trích xuất mối quan hệ giữa bệnh lý và thuốc sẽ bao gồm 2 thực thể, cho nên luận văn sẽ trích xuất 2 quan hệ liên quan tới vị trí của từ đến thực thể bệnh lý và đến thực thể thuốc.

3.3 Mô hình mạng nơ-ron tích chập CNN kết hợp với LSTM

3.3.1 Tầng mạng nơ-ron hồi quy LSTM

Trong phần trước luận văn đã trình bày phương pháp tạo ra vector biểu diễn cho mỗi từ trong đoạn văn bản là một tổ hợp của vector nhúng từ theo ngữ cảnh, vector từ loại và vector vị trí. Tuy nhiên, các biểu diễn từ loại và vị trí là những vector được khởi tạo ngẫu nhiên từ đầu và cần phải được cập nhật trong quá trình huấn luyện. Vì vậy, luận văn đã sử dụng mạng LSTM để mã hóa lại thông tin ngữ cảnh cũng như cung cấp cho mô hình khả năng uyển chuyển để thay đổi các biểu diễn đó sao cho phù hợp với bài toán. Hình 3.3 minh họa mô hình LSTM mà luận văn sử dụng.



Hình 3.3 Minh họa mô hình LSTM sử dụng để thu thập thông tin ngữ cảnh.

Như đã trình bày ở mục 2.2, mạng LSTM bao gồm các cổng điều khiển để khắc phục vấn đề tiêu biến đạo hàm. Ở mỗi bước *t*, mạng LSTM tính toán trạng thái ẩn h_t và trạng thái tế bào *ct* bằng cách sử dụng vector đầu vào *xt*, trạng thái ẩn trước đó h_{t-1} và trạng thái tế bào trước đó c_{t-1}. Quá trình tính toán cụ thể như sau:

$$\begin{gathered} i_t=\sigma(W_i x_t+U_i h_{(t-1)}+b_i) \\ f_t=\sigma(W_f x_t+U_f h_{(t-1)}+b_f) \\ o_t=\sigma(W_o x_t+U_o h_{(t-1)}+b_o) \\ g_t=\tanh(W_g x_t+U_g h_{(t-1)}+b_g) \\ c_t=f_t \odot c_{(t-1)}+ i_t \odot g_t \\ h_t= o_t \odot \tanh(c_t) \end{gathered} \tag{3.2}$$

Thêm nữa, luận văn sử dụng mô hình LSMT hai chiều (Bidirectional LSTM) bao gồm hai mạng LSTM riêng biệt gọi là Forward LSTM và Backward LSTM để đồng thời mã hóa được cả thông tin ngữ cảnh từ trái qua phải và từ phải qua trái. Cuối cùng, với mỗi vector biểu diễn *xt*, mạng Bidirectional LSTM tạo ra một trạng thái ẩn cuối cùng h_t là kết quả của phép nối hai vector trạng thái ẩn xuôi h_f và ngược h_b. Quá trình tình toán diễn ra như sau.

$$\begin{gathered} h_t^f = LSTM^f\ (x_t, h_{t-1}^f) \\ h_t^b = LSTM^b\ (x_t, h_{t-1}^b) \\ h_t = h_t^f \circ h_t^b \end{gathered} \tag{3.3}$$

3.3.2 Tầng mạng nơ-ron tích chập CNN

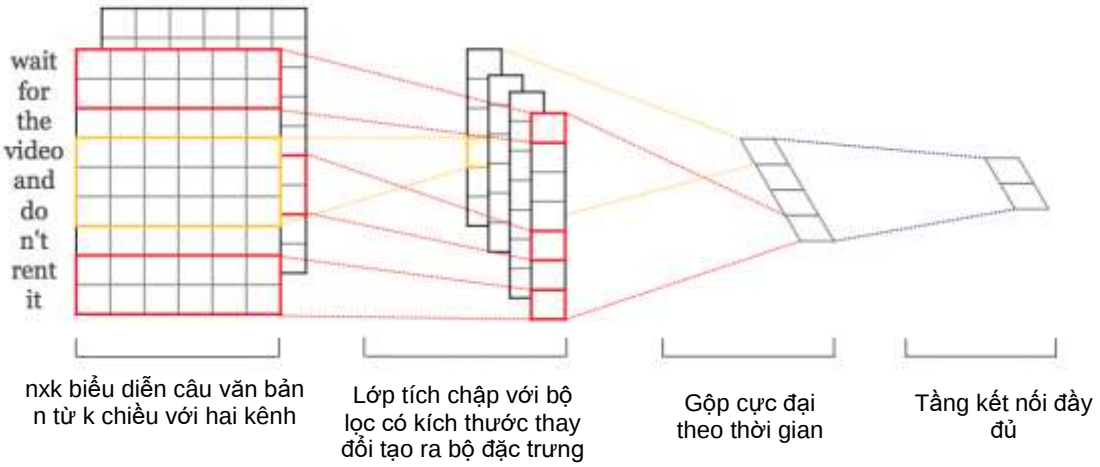
Trong những năm gần đây, mạng nơ-ron hồi quy LSTM và mạng nơ-ron tích chập CNN là hai mạng học sâu phổ biến nhất, đều đã được áp dụng thành công trong trích xuất quan hệ CID từ văn bản y sinh. Giữa mạng hồi quy LSTM và mạng tích chập CNN có những đặc tính khác nhau rõ rệt. Mô hình mạng nơ-ron hồi quy có kiến trúc mạng nơ-ron tuần tư và mạnh hơn trong việc nắm bắt các đặc trưng của các chuỗi từ dài – là các đặc trưng phụ thuộc liên kết xa ở liên câu. Trong khi đó mạng nơ-ron tích chập CNN có kiến trúc mạng thần kinh phân cấp và học tốt các đặc trưng từ vựng và cú pháp cục bộ. Bởi vậy mô hình mạng nơ ron tích chập phù hợp để nắm bắt các đặc trưng của câu ngắn, trong khi mô hình mạng nơ ron hồi quy thích hợp hơn để xử lý các câu dài và

phức tạp hoặc cũng như đặc trưng giữa các câu với nhau trong văn bản. Luận văn sử dụng mô hình kết hợp các ưu điểm của mạng nơ-ron tích chập CNN và mạng nơ-ron hồi quy LSTM để trích xuất các quan hệ CID từ văn bản y sinh.

Từ kết quả xử lý của mạng LSTM hai chiều với khả năng mã hóa lại thông tin ngữ cảnh cũng như cung cấp cho mô hình khả năng nắm bắt những phụ thuộc xa, mô hình đề xuất đưa qua mạng nơ-ron tích chập để trích xuất được các đặc trưng mong muốn trong quan hệ CID.

Lấy ý tưởng từ [5], luận văn sử dụng mạng nơ ron tích chập CNN với các thành phần: Mạng nơ ron tích chập bao gồm một tập hợp các lớp tích chập được chồng lên nhau và sử dụng các hàm kích hoạt không tuyến tính như ReLU hay tanh.

Phép tích chập sử dụng một hạt nhân và biến đổi với dữ liệu của các lớp trước để tạo ra một dữ liệu mới, gọi là các dữ liệu đặc trưng và cung cấp chúng cho các lớp tiếp theo. Các phép gộp, như gộp tối đa (max-pooling) hoặc là gộp trung bình (average-pooling) có thể được thêm vào sau khi tích chập để giảm kích thước của các đặc trưng. Điều này cho phép mô hình giảm chi phí tính toán và phân tích dữ liệu ở nhiều mức độ khác nhau. Ngoài các lớp này, mạng nơ ron tích chập cũng có thể kết hợp với các mạng nơ ron khác và hoạt động bình thường. Hình 3.4 minh họa kiến trúc mô hình mạng CNN trong phân lớp câu.



Hình 3.4 Kiến trúc mô hình mạng CNN với hai kênh cho đầu vào cho câu văn bản [5]

Với $x_i \in R^k$ là biểu diễn từ thứ i trong câu tương ứng với vector biểu diễn từ k-chiều. Khi đó câu có độ dài n từ (bổ sung bước đệm nếu cần) sẽ được biểu diễn dưới dạng

$$x_{1:n} = x_1 \oplus x_2 \oplus ... \oplus x_n \quad (3.4)$$

Trong đó $\oplus$ là phép nối chuỗi, $x_{i:i+j}$ tương ứng với chuỗi các từ $x_i, x_{i+1}, \dots, x_{i+j}$. Áp dụng phép tích chập liên quan đến bộ lọc $w \in \mathbb{R}^{hk}$ với cửa sổ h từ để tạo ra một đặc trưng (feature) mới. Áp dụng lọc với cửa sổ h trượt từ đầu đến hết câu để tạo ra bộ đặc trưng (feature map).

Sau đó áp dụng phép gộp cực đại theo thời gian (max-over-time pooling) đối với bộ đặc trưng này này để giữ lại những đặc tính quan trọng nhất – tương ứng với giá trị lớn nhất – cho từng bộ đặc trưng. Phép gộp này cũng xử lý tốt đối với độ dài thay đổi theo câu.

Mỗi đặc tính được trích xuất từ một bộ lọc, mô hình sử dụng nhiều bộ lọc (với kích thước cửa sổ thay đổi) để thu được nhiều đặc tính. Những đặc tính này tạo thành lớp áp chót trước khi chuyển sang lớp softmax kết nối đầy đủ có đầu ra là phân phối xác suất trên các nhãn.

3.4 Dự đoán mức định danh

Do các quan hệ bệnh do hóa chất gây ra (CID) được gán nhãn ở mức định danh thay vì ở mức đề cập, cho nên luận văn đề xuất một phương pháp tính toán cho phép tổng hợp lại thông tin từ tất cả các cặp đề cập có thể có của một cặp thực thể hóa chất và bệnh tật tương ứng, ở trong toàn bộ văn bản. Toàn bộ thông tin thu được ở mức đề cập sẽ được sử dụng để đưa ra dự đoán cuối cùng ở mức định danh.

Xét $c = \{c1, c2, ..., cm\}$ và $d = \{d1, d2, ..., dn\}$ lần lượt là tập vector biểu diễn của các đề cập thực thể hóa chất và căn bệnh, trong đó m và n là số lần được đề cập đến trong tài liệu của mỗi loại thực thể. Luận văn sử dụng các phép biến đổi tuyến tính riêng biệt cùng với hàm kích hoạt tanh để giảm chiều cũng như chiếu mỗi vector đề cập của bệnh và hóa chất xuống các không gian biểu diễn khác nhau. Biểu thức tính toán cụ thể như sau.

$$\begin{aligned} c_i^{final} &= \tanh(W_c c_i + b_c), \forall i = 1 \dots m \\ d_j^{final} &= \tanh\left(W_d c_j + b_d\right), \forall j = 1 \dots n \end{aligned} \tag{3.5}$$

Trong đó W_c, b_c và W_d, b_d là các trọng số và độ lệch của mô hình tương ứng cho các thực thể hóa chất và bệnh tật. c_i^{final} và d_j^{final} lần lượt là các vector biểu diễn cuối cùng cho đề cập thứ i của thực thể hóa chất và đề cập thứ j của thực thể bệnh.

Để tính điểm dự đoán cho mỗi cặp đề cập hóa chất - bệnh, luận văn sử dụng các vector biểu diễn cuối cùng của chúng và thông tin về khoảng cách tương đối giữa cặp đề cập đó trên văn bản (khoảng cách này được đo bằng số từ nằm giữa hai đề cập). Cụ thể, luận văn tính toán một vector hai chiều, biểu diễn cho việc có hay không mối quan hệ CID giữa hai đề cập thực thể. Công thức tính toán như sau.

$$a_{ij} = W_{score}\left(c_i^{final} \circ d_j^{final} \circ R_{|p_{c_i} - p_{d_j}|}\right) + b_{score} \tag{3.6}$$

Trong đó W_{score}, b_{score} là các tham số của mô hình, $R_{|p_{c_i} - p_{d_j}|}$ là vector biểu diễn của khoảng cách tương đối giữa hai đề cập thực thể và có thể được cập nhật khi huấn luyện.

Ngoài vector biểu diễn của mỗi đề cập thực thể thì thông tin về khoảng cách giữa chúng cũng là cần thiết để làm tăng tính chính xác cho dự đoán. Một cách trực giác, dự đoán của các cặp đề cập nằm cách xa nhau thông thường sẽ rất khó để có thể xác định được đúng quan hệ so với dự đoán của các cặp đề cập có khoảng cách gần nhau. Một giải pháp là chúng ta có thể chọn một ngưỡng cố định từ trước, mà tại đó, dự đoán của các cặp đề cập có khoảng cách lớn sẽ không được xem xét. Tuy nhiên việc lựa chọn giá trị ngưỡng phù hợp thường rất khó và đòi hỏi thời gian tìm kiếm thông qua tập phát triển. Vì vậy, luận văn thêm thông tin khoảng cách dưới dạng vector để mô hình có thể tự động học cách đưa ra trọng số cho mỗi cặp đề cập dựa trên khoảng cách giữa chúng. Các vector biểu diễn cho khoảng cách tương đối được khởi tạo ngẫu nhiên và cho phép cập nhật trong quá trình huấn luyện mô hình.

Cuối cùng, luận văn tính toán điểm số cho dự đoán ở mức định danh bằng cách sử dụng hàm max pooling trên tập dự đoán của tất cả các cặp đề cập.

$$\text{final\_score}(c, d) = \max(a_{ij}), \forall i = 1...m, \boldsymbol{j} = 1...n \tag{3.7}$$

3.5 Huấn luyện mô hình

Đối với bài toán trích xuất quan hệ, luận văn đưa điểm dự đoán ở mức định danh qua một hàm softmax để tính một phân phối xác suất trên tập các nhãn quan hệ.

$$\mathbf{P}(\mathbf{r}_{c,d}) = \text{Softmax}(\text{final\_score}(c, d)) \tag{3.8}$$

Sau đó thực hiện tối thiểu hóa hàm negative log-likelihood của nhãn thực sự của quan hệ khi biết các tham số mô hình θ_{re} cho bài toán trích xuất quan hệ; $r^*_{c,d}$ là nhãn thực sự của quan hệ giữa thực thể hóa chất *c* và thực thể căn bệnh *d*.

$$l_{re} = -\log p(r_{c,d} = r^*_{c,d} | \theta_{re}) \tag{3.9}$$

Chương 4: KẾT QUẢ THỰC NGHIỆM VÀ KẾT LUẬN

Từ mô hình, phương pháp thực hiện đã được đề xuất, chương này luận văn trình bày về cách thực hiện dự đoán tương tác bệnh – thuốc, từ việc lựa chọn bộ dữ liệu, luồng quy trình xử lý, lựa chọn độ đo đánh giá đến khi đưa những kết quả thu được ttheo mục tiêu của luận văn. Từ kết quả đó, luận văn đưa ra những thảo luận, đánh giá, so sánh với với những phương pháp tương đương gần đây cũng như mở ra hướng nghiên cứu trong tương lai.

4.1 Độ đo đánh giá

Để đo lường hiệu suất của mô hình, luận văn thực hiện tính điểm F1 trên lớp CID. Cụ thể, luận văn tính toán hai độ đo trung gian khác là Precision và Recall. Trong đó, Precision được định nghĩa là tỉ lệ giữa số lượng quan hệ mô hình dự đoán được chính xác chia cho số lượng dự đoán mô hình đưa ra. Công thức tính điểm Precision như sau:

$$\text{Precision} = \frac{|\text{Predicted} \cap \text{Golden}|}{|Predicted|} \tag{4.1}$$

Ở phần còn lại, Recall được định nghĩa là tỉ lệ giữa số lượng quan hệ mô hình dự đoán được chính xác chia số lượng quan hệ thật sự trong dữ liệu.

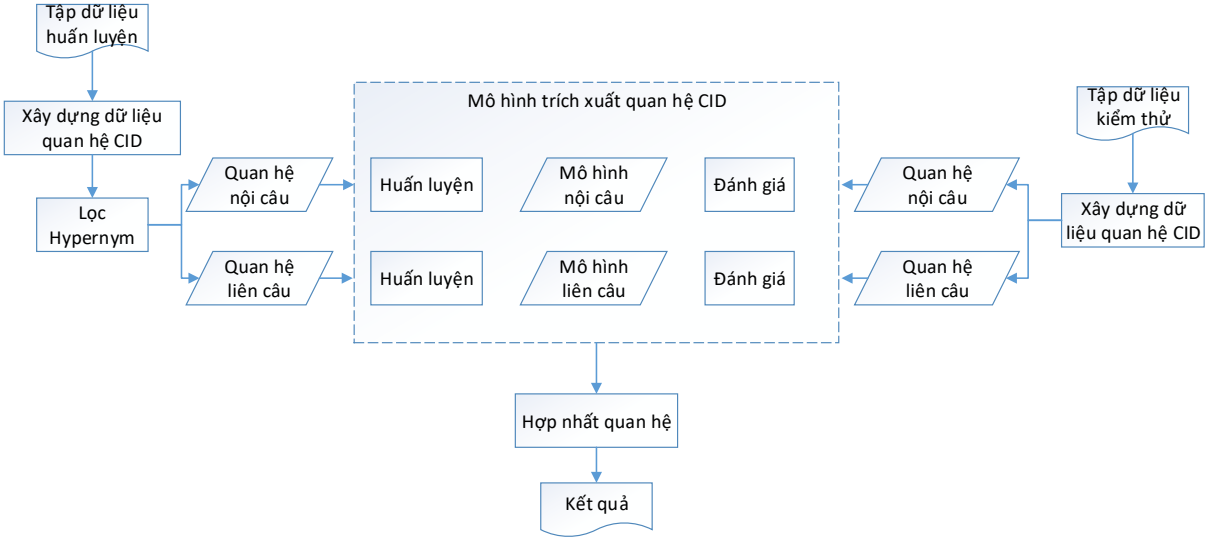
$$\text{Recall} = \frac{|\text{Predicted} \cap \text{Golden}|}{|Golden|} \tag{4.2}$$

Precision trả lời cho câu hỏi: trong số các điểm dữ liệu được mô hình phân loại vào lớp Positive, có bao nhiêu điểm dữ liệu thực sự thuộc về lớp Positive. Mặt khác, Recall giúp chúng ta biết được có bao nhiêu điểm dữ liệu thực sự ở lớp Positive được mô hình phân lớp đúng trong mọi điểm dữ liệu thực sự ở lớp Positive.

Một mô hình tốt khi cả Precision và Recall đều cao, thể hiện cho mô hình ít phân loại nhầm giữa các lớp cũng như tỉ lệ bỏ sót các đối tượng thuộc lớp cần quan tâm là thấp. Tuy nhiên, hai giá trị Precision và Recall thường không cân bằng với nhau (giá trị này tăng thì giá trị kia thường có xu hướng giảm). Độ đo F1 là một đại lượng cân bằng được tính bằng điểm Precision và Recall thông qua công thức sau.

$$\text{F1} = \frac{2 \times Precision \times Recall}{Precision + Recall} \tag{4.3}$$

4.2 Cách thức thực hiện



Hình 4.1 Cách thức thực hiện dự đoán tương tác thuốc

Phần này trình bày cách tiếp cận học có giám sát đối với việc trích xuất quan hệ CID. Hình 4.1 trình bày kiến trúc tổng thể xử lý dữ liệu của mô hình mà luận văn đề xuất. Toàn bộ quy trình của phương pháp tiếp cận có thể được chia thành các bước tuần tự như sau:

Tạo dữ liệu quan hệ CID (Relation instance construction):

Đầu tiên, từ bộ dữ liệu BioCreative V CDR (sẽ được đề cập ở phần sau) làm đầu vào, các cặp đề cập thuốc và bệnh dưới dạng <đề cập thuốc, đề cập đến bệnh> được trích xuất dưới dạng cặp đề cập có thể có bằng cách sử dụng một số quy tắc lọc theo phương pháp dựa trên cả tập dữ liệu huấn luyện và dữ liệu kiểm tra.

Tất cả các cặp đề cập thuốc – bệnh được tạo ra từ các đề cập đến thuốc và bệnh trong cùng một văn bản theo cách ghép đôi, tức là nếu một tài liệu chứa m đề cập thuốc khác nhau và n đề cập bệnh khác nhau, thì sẽ có m x n cặp đề cập đến bệnh và thuốc khác nhau. Các cặp đề cập này được gộp thành hai nhóm/tập dữ liệu tương ứng ở cấp độ nội câu và cấp độ liên câu. Nhóm nội câu có nghĩa là một cặp đề cập đến từ cùng một câu, trong khi nhóm liên câu có nghĩa các cặp đề cập nằm ở các câu khác nhau. Sau khi áp dụng các quy tắc lọc dựa trên kinh nghiệm (heuristic) khác nhau để xây dựng tập dữ liệu các cặp đề cập được coi là các thể hiện quan hệ CID.

Xây dựng dữ liệu quan hệ CID ở cấp độ nội câu

Trước khi đưa vào mô hình trích xuất quan hệ CID, cần xây dựng bộ dữ liệu các cặp đề cập thuốc – bệnh ở cấp độ nội câu cho cả quá trình huấn luyện và đánh giá. Vì mục đích này, luận văn đã áp dụng một số quy tắc dựa trên kinh nghiệm (heuristic) đơn giản nhưng hiệu quả như sau:

1) Khoảng cách thực thể báo giữa hai đề cập trong một cặp phải nhỏ hơn k (ở đây k được đặt k là 10 theo kinh nghiệm).
2) Nếu có nhiều đề cập trong một câu đề cập đến cùng một thực thể, thì cặp đề cập thuộc - bệnh gần nhất nên được giữ lại.
3) Bất kỳ đề cập nào xảy ra trong dấu ngoặc đơn nên được bỏ qua.

Xây dựng dữ liệu quan hệ CID ở cấp độ liên câu

Việc xây dựng cặp đề cập thuốc – bệnh ở cấp độ liên câu để huấn luyện và đánh giá tuân thủ các quy tắc sau:

1) Chỉ những thực thể không nằm trong tập dữ liệu quan hệ CID ở cấp độ nội câu mới được xem xét ở cấp độ liên câu.
2) Khoảng cách câu giữa hai lần đề cập trong một cặp đề cập thuốc – bệnh phải nhỏ hơn n (ở đây đặt n là 3 theo kinh nghiệm).
3) Nếu có nhiều lượt đề cập đề cập đến cùng một thực thể, hãy giữ nguyên cặp đề cập thuốc - bệnh có khoảng cách gần nhất.

Lọc hypernym (bao quát) cho tập dữ liệu huấn luyện

Trong một số trường hợp, có mối quan hệ bao quát/chi tiết (hypernym/hyponym) giữa các khái niệm về bệnh hoặc thuốc, trong đó một khái niệm này phụ thuộc vào một khái niệm khác bao quát hơn. Tuy nhiên, các quan hệ bệnh do hóa chất gây ra (CID) chỉ được gán cho các cặp thực thể hóa chất - bệnh tật cụ thể nhất. Lấy ví dụ, tương tác *tobacco causes cancer* ("thuốc lá gây ung thư") có thể sẽ bị gán váo lớp Negative nếu trong văn bản tồn tại một thực thể bệnh cụ thể hơn ví dụ như *lung cancer* ("ung thư phổi"). Điều này có thể khiến các tương tác đúng bị gán nhãn là sai, gây ảnh hưởng tới hiệu suất của mô hình.

Trích xuất quan hệ CID

Trích xuất quan hệ CID có thể được xây dựng như một bài toán phân loại nhị phân. Từ tập dữ liệu các cặp đề cập thuốc – bệnh đã được xây dựng ở cấp độ nội câu, liên câu để thực hiện huấn luyện mô hình từ tất cả các cặp đề cập **có thể có** của một cặp thực thể hóa chất và bệnh tật tương ứng, ở trong toàn bộ văn bản với tập huấn luyện và điều chỉnh các siêu tham số bằng tập phát triển. Cuối cùng, sử dụng cả tập huấn luyện lẫn tập phát triển để huấn luyện mô hình cùng các siêu tham số đã tìm được trước đó. Mô hình sau khi được huấn luyện sẽ được đánh giá trên tập Test.

Hợp nhất quan hệ (Relation merging)

Do các quan hệ CID được gán nhãn ở mức định danh thay vì ở mức đề cập (mention), nên cần thực hiện tổng hợp dự đoán. Một giả định là một cặp thực thể bệnh – thuốc có thể được đề cập nhiều lần ở cấp độ nội câu hoặc cấp độ liên câu và nếu ít nhất một cặp đề cập này có mối quan hệ CID, luân văn sẽ tổng hợp và coi hai thực thể bệnh – thuốc có mối quan hệ CID thực sự.

4.3 Bộ dữ liệu văn bản y sinh BioCreative V CDR

4.3.1 Dữ liệu quan hệ thuốc và bệnh - BioCreative V CDR

Luận văn sử dụng bộ dữ liệu BioCreative V CDR [16] để huấn luyện, phát triển và đánh giá mô hình. Bộ dữ liệu chuẩn bao gồm 1500 bản tóm tắt trích ra từ kho PubMed, với 500 bản tóm tắt cho tập huấn luyện, tập phát triển và tập kiểm tra tương ứng. Bảng 4.1 mô tả vài thống kê của bộ dữ liệu CDR.

Bảng 4.1 Một vài thống kê về bộ dữ liệu CDR

Subset	Abstracts	Chemical-induced
Training	500	1038
Development	500	1012
Test	500	1066

Dữ liệu BioCreative V CDR được cung cấp dưới hai định dạng khác nhau là PubTator (định dạng text) và BioC (định dạng XML) nên chúng ta chỉ cần sử dụng một trong hai loại này để tiến hành xử lý dữ liệu. Ở đây tác giả chọn sử dụng định dạng PubTator để tiến hành xử lý.

Dữ liệu mỗi bài viết trong BioCreative V CDR bao gồm có tiêu đề và tóm tắt của văn bản như minh họa trong Hình 4.2:

```
2375138|t|Possible intramuscular midazolam-associated cardiorespiratory arrest and death.
2375138|a|Midazolam hydrochloride is commonly used for dental or endoscopic procedures. A
2375138 23  32  midazolam   Chemical    D008874
2375138 44  68  cardiorespiratory arrest    Disease D006323
2375138 73  78  death   Disease D003643
2375138 80  103 Midazolam hydrochloride Chemical    D008874
2375138 265 306 respiratory and cardiovascular depression   Disease D012140|D002318 respi
2375138 358 382 cardiorespiratory arrest    Disease D006323
2375138 387 392 death   Disease D003643
2375138 441 450 midazolam   Chemical    D008874
2375138 474 483 midazolam   Chemical    D008874
2375138 CID D008874 D012140
2375138 CID D008874 D006323

2265898|t|Serial epilepsy caused by levodopa/carbidopa administration in two patients on
2265898|a|Two patients with similar clinical features are presented: both patients had ch
2265898 7   15  epilepsy    Disease D004827
2265898 26  44  levodopa/carbidopa  Chemical    C009265
2265898 170 191 chronic renal failure   Disease D007676
2265898 294 312 carbidopa/levodopa  Chemical    C009265
2265898 352 364 hallucinosis    Disease D001523
2265898 379 387 seizures    Disease D012640
2265898 550 560 vitamin B6  Chemical    D025101
2265898 CID C009265 D004827
```

Hình 4.2 Dữ liệu định dạng PubTator của BioCreative V CDR

Mỗi thực thể thuốc và bệnh sẽ bao gồm các thông tin: vị trí xuất hiện trong bài viết, tên thuốc/bệnh, loại (thuốc/bệnh) và mã định danh của thực thể.

Các bản ghi thực thể thuốc và bệnh đã được nhận diện và tách ra thành các dòng riêng có dánh dấu vị trí của chúng xuất hiện trong văn bản y sinh.

Mối quan hệ của thuốc và bệnh xuất hiện trong văn bản y sinh được thể hiện bởi các dòng có chữ "CID", tiếp theo là mã định danh của thuốc và mã định danh của bệnh. Chúng ta hiểu được khi một cặp thuốc và bệnh xuất hiện tại đây là cặp thuốc và bệnh lý này "Có quan hệ" với nhau.

Để tạo ra dữ liệu huấn luyện, đầu tiên tiến hành loại bỏ hết các thực thể hóa chất và bệnh tật có id là -1. Tiếp theo, tất cả các thực thể hóa chất và bênh tật xuất hiện trong văn bản sẽ được ghép cặp lại, quá trình này được thực hiện ở mức định danh chứ không phải mức đề cập. Sau đó, với một cặp định danh hóa chất - bệnh tật tương ứng, nếu cặp này được bộ dữ liệu CDR gán nhãn là CID thì tiến hành gán nó vào lớp Positive (1), ngược lại thì sẽ là lớp Negative (0).

Trong bộ dữ liệu CDR, các quan hệ bệnh do hóa chất gây ra (CID) chỉ được gán cho các cặp thực thể hóa chất - bệnh tật cụ thể nhất. Lấy ý tưởng của [11], luận văn thực hiện lọc tất cả các cặp hóa chất - bệnh tật chứa hypernyms sử dụng tính phân cấp của bộ từ điển MESH (Medical Subject Headings). Bảng 4.2 mô tả số lượng các cặp hóa chất - bệnh tật lọc ra được bởi MESH.

Bảng 4.2 Số lượng cặp hóa chất - bênh tật được lọc ra bởi MESH.

Subset	**Number of filtered negative**
Training	192
Development	174
Test	201

Thực hiện huấn luyện mô hình với tập huấn luyện và điều chỉnh các siêu tham số bằng tập phát triển. Cuối cùng, sử dụng cả tập huấn luyện lẫn tập phát triển để huấn luyện mô hình cùng các siêu tham số đã tìm được trước đó. Mô hình sau khi được huấn luyện sẽ được đánh giá ở trên tập Test.

4.4 Cài đặt thực nghiệm

4.4.1 Thư viện sử dụng

Luận văn tiến hành cài đặt mô hình bằng Pytorch - một thư viện mã nguồn mở để phát triển các mô hình Học Sâu. Đối với vector nhúng từ theo ngữ cảnh BioELMo, sử dụng mô hình đã được huấn luyện từ thư viện AllenNLP [17]. Ngoài ra, luận văn sử dụng thư viện ScispaCy [18] với một tập các tính năng hoàn chỉnh cho xử lý văn bản y sinh học, bao gồm tách từ, phân tích cú pháp phụ thuộc và gán nhãn từ loại. Một số thư viện phụ trợ khác cũng được sử dụng khi cài đặt mô hình là Pandas, Numpy, và Sklearn. Mô hình được huấn luyện trên 1 GPU Tesla T4 với 15GB bộ nhớ.

4.4.2 Các siêu tham số của mô hình

Luận văn đặt số chiều của vector từ loại là 10. Trong khi, số chiều của vector nhúng position và nhúng từ theo ngữ cảnh (BioELMo) được luận văn đặt là 30 và 1024 tương ứng. Đối với mạng LSTM, luận văn đặt số chiều của trạng thái ẩn xuôi và ngược là 100,

vì vậy biểu diễn đầu ra của mô hình LSTM sẽ có chiều là 200. Việc biểu diễn khoảng cách tương đối giữa hai đề cập của thực thể bằng một vector 50 chiều.

Với huấn luyện mô hình, luận văn sử dụng thuật toán tối ưu AdamW [19]. Mô hình được huấn luyện với tốc độ học 7e-4 và kích cỡ của minibatch là 8. Để hạn chế vấn đề quá khớp, luận văn sử dụng L2-Regularization với hệ số λ là 0.001 và tốc độ học được giảm một nữa sau mỗi một epoch.

4.4.3 Kết quả thực nghiệm

Trong phần này, luận văn báo cáo các kết quả thực nghiệm đã làm được. Các thí nghiệm tập trung vào việc nghiên cứu ảnh hưởng của biểu diễn đầu vào. Thực hiện so sánh kết quả của mô hình đề xuất (mạng nơ-ron tích chập CNN kết hợp với mạng hồi quy LSTM) với nhiều phương pháp tiên tiến gần đây trên thế giới cho bài toán trích xuất quan hệ CID. Với mỗi thí nghiệm, luận văn sử dụng kết quả trung bình trên 10 lần chạy với các random seed khác nhau làm kết quả cuối cùng.

So sánh mô hình kết hợp giữa CNN và LSTM với các phương pháp tiên tiến gần đây trên thế giới cho bài toán trích xuất quan hệ CID trên cùng bộ dữ liệu BioCreative V CDR. Các nghiên cứu đó được liệt kê sau đây.

Bảng 4.3 So sánh về hiệu suất của mô hình đề xuất với một số nghiên cứu khác

Model	Precision (P)	Recall (R)	F1 score (F1)
LSTM + SVM [17]	64.9	49.3	56.0
LSTM + SVM + PP [17]	55.6	68.4	61.3
CNN +ME [2]	60.9	59.5	60.2
CNN +ME + PP [2]	55.7	68.1	61.3
GCN + Multi-Head Attn [4]	56.3	72.7	63.5
CNN +LSTM (**Ours**)	56.0	72.4	63.1

• LSTM + SVM (Zhou et al., 2016) [17]: Long short-term memory + Support vector machine.

• LSTM + SVM + PP (Zhou et al., 2016) [17]: Long short-term memory + Support vector machine + Post processing.

• CNN + ME (Gu et al., 2017) [2]: Convolutional neural network + Maximum entropy model.

• CNN + ME + PP (Gu et al., 2017) [2]: Convolutional neural network + Maximum entropy model + Post processing.

• GCN + Multi-Head Attn (Wang et al., 2020) [4]: Graph convolutional network + Multi-head self- attention mechanism.

Bảng 4.3 mô tả chi tiết các so sánh kết quả của mô hình đề xuất với một số phương pháp gần đây trên thế giới cho bài toán trích xuất quan hệ CID. Đầu tiên, luận văn thực

hiện so sánh mô hình đề xuất với các phương pháp không sử dụng đồng thời cả mô hình mạng nơ-ron tích chập CNN và mô hình hồi quy LSTM. Mô hình đề xuất đạt được kết quả tốt hơn so với mô hình LSTM + SVM, LSTM + SVM + PP (Zhou et al., 2016) [17] cũng như mô hình CNN + ME, CNN + ME + PP (Gu et al., 2017) [2]. Có thể thấy, với một vài phương pháp hậu xử lý (PP), hiệu suất của những nghiên cứu nêu trên đã tăng lên đáng kể. Điều này cho thấy rằng việc tích hợp các quy tắc dựa trên kinh nghiệm (heuristic) có thể làm tăng hiệu suất cho bài toán trích xuất quan hệ CID.

Tuy nhiên, khi so sánh với các mô hình sử dụng cấu trúc đồ thị, mô hình GCN + Multi-Head Attn (Wang et al., 2020) [4] có hiệu suất tốt hơn mô đề xuất một chút, khoảng 0.4 điểm F1. Trong phương pháp của (Wang et al., 2020) [4] đã sử dụng kết hợp mô hình GCN với cơ chế Multi-head Self-attention trên đồ thị phụ thuộc mức tài liệu, mô hình mạng tích chập đồ thị (GCN) có thể nắm bắt tốt các thông tin phụ thuộc xa khi phải xử lý các đoạn văn bản có độ dài lớn – so với mô hình hồi quy LSTM đơn thuần.

Các so sánh nêu trên đã cho thấy mô hình đề xuất (mạng nơ-ron tích chập CNN kết hợp với mạng hồi quy LSTM) đạt được những kết quả rất đáng khích lệ khi đánh giá cùng với nhiều phương pháp hiện đại khác cho bài toán trích xuất quan hệ CID.

4.5 Kết luận

Luận văn đã giới thiệu một mô hình nơ-ron kết hợp giữa mạng CNN và LSTM cho việc giải quyết bài toán trích xuất quan hệ bệnh lý do thuốc gây ra (CID). Mô hình mạng nơ ron tích chập phù hợp để nắm bắt các đặc trưng của câu ngắn, trong khi mô hình mạng nơ ron hồi quy thích hợp hơn để xử lý các câu dài và phức tạp hoặc cũng như đặc trưng giữa các câu với nhau trong văn bản. Luận văn sử dụng mô hình kết hợp các ưu điểm của mạng nơ-ron tích chập CNN và mạng nơ-ron hồi quy LSTM để trích xuất các quan hệ CID từ văn bản y sinh. Thêm nữa, luận văn nâng cấp biểu diễn đầu vào của mô hình với phương pháp nhúng từ dựa trên ngữ cảnh mạnh mẽ cho miền y sinh học (ELMo). Kết quả thực nghiệm đã cho thấy mô hình kết hợp giữa mạng tích chập CNN và mạng hồi quy LSTM đạt được 63.1 điểm F1. Phương pháp được đề xuất trong luận văn đã đạt được những kết quả với độ chính xác khá tốt, có thể áp dụng được vào thực tế cho bài toán trích xuất quan hệ CID.

4.6 Hướng nghiên cứu trong tương lai

Trong tương lai, tác giả sẽ tiếp tục thu thập và bổ sung các phương pháp khác cũng như áp dụng thêm một số đặc trưng khác để, để cải tiến hiệu năng dự đoán của thuật toán cũng như tìm cách tối ưu các tham số của thuật toán tự động để đạt được kết quả cao hơn.

Tài liệu tham khảo

Tiếng Anh

[1] Wahiba Ben Abdessalem Karaa, Eman H. Alkhammash, A, Drug Disease Relation Extraction from Biomedical Literature Using NLP and Machine Learning, Mobile Information Systems, vol. 2021, Article ID 9958410, 10 pages, 2021.

[2] Gu, Jinghang and Sun, Fuqing and Qian, Longhua and, Chemical-induced disease relation extraction via convolutional neural network, 2017.

[3] Mingbo Ma et all, Dependency-based Convolutional Neural Networks for Sentence Embedding, 2015.

[4] Wang J, Chen X, Zhang Y, et al, Document-Level Biomedical Relation Extraction Using Graph Convolutional Network and Multihead Attention: Algorithm Development and Validation, JMIR Med Inform, 2020.

[5] Yoon Kim, Convolutional neural networks for sentence classification. In Proceedings of EMNLP, 2014.

[6] Dean, Tomas Mikolov and Ilya Sutskever and Kai Che, Distributed Representations of Words and Phrases and their Compositionality, 2013.

[7] Pennington, Jeffrey and Socher, Richard and Mannin, GloVe: Global Vectors for Word Representation, 2020.

[8] Zettlemoyer, Matthew E. Peters and Mark Neumann an, Deep contextualized word representations, 2018.

[9] Lu, Qiao Jin and Bhuwan Dhingra and William W. Coh, Probing Biomedical Embeddings from Language Models, 2019.

[10] Jun Xu and Y. Wu and Y. Zhang and J. Wang and Hee-, CD- REST: a system for extracting chemical-induced disease relation in literature, 2016.

[11] Jinghang Gu and Longhua Qian and Guodong Zhou, Chemical-induced disease relation extraction with various linguistic features, 2016.

[12] Huiwei Zhou and Huijie Deng and Jiao He, Chemical-disease Relations Extraction Based on The Shortest Dependency Path Tree, 2015.

[13] Nguyen, Dat Quoc and Verspoor, Karin, Convolutional neural networks for chemical-disease relation extraction are improved with character-based word embeddings, 2018.

[14] Ashish Vaswani and Noam Shazeer and Niki Parmar an, Attention Is All You Need, 2017.

[15] Sahu, Sunil Kumar and Christopoulou, Fenia and Miw, Inter-sentence Relation Extraction with Document-level Graph Convolutional Neural Network, 2019.

[16] Wei, Chih-Hsuan and Peng, Yifan and Leaman, Robert, Assessing the state of the art in biomedical relation extraction: Overview of the BioCreative V chemical-disease relation (CDR) task, 2016.

[17] Zettlemoyer, Matt Gardner and Joel Grus and Mark N, AllenNLP: A Deep Semantic Natural Language Processing Platform, 2017.

[18] Neumann, Mark and King, Daniel and Beltagy, Iz and, ScispaCy: Fast and Robust Models for Biomedical Natural Language Processing, 2017.

[19] Ilya Loshchilov, Frank Hutter, Decoupled Weight Decay Regularization, 2017.

[20] Huang, Huiwei Zhou and Huijie Deng and Long Chen a, Exploiting syntactic and semantics information for chemical–disease relation extraction, 2016.

[21] Patrick Verga and Emma Strubell and Andrew McCallu, Simultaneously Self-Attending to All Mentions for Full-Abstract Biological Relation Extraction, 2018.

[22] Andrej Kastrin et all, Predicting potential drug-drug interactions on topological and semantic similarity features using statistical learning, 2018.

[23] Chunyun Zhang et all, Multi-Gram CNN-Based Self-Attention Model for Relation Classification, vol. 7, IEEE Access, 2019, p. 5343 – 5357.

[24] François Chollet, Deep Learning with Python, 2018.

[25] Yijia Zhang, Hongfei Lin, Zhihao Yang, et all, A hybrid model based on neural networks for biomedical relation extraction, vol. 81, Journal of Biomedical Informatics, 2018, pp. 83-92.

[26] Keiron O'Shea, Ryan Nash, An Introduction to Convolutional Neural Networks, 2015.