**TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN**

**KHOA CÔNG NGHỆ THÔNG TIN**

**CHƯƠNG TRÌNH CHẤT LƯỢNG CAO**

**ĐẶNG NGUYỄN DUY – NGUYỄN NHỰT LINH**

**MÔ HÌNH GIÀN GIÁO NGỮ PHÁP CHO TÁC VỤ RÚT TRÍCH CẤU TRÚC ĐỐI SỐ VỊ NGỮ TRONG VĂN BẢN Y SINH**

**KHÓA LUẬN TỐT NGHIỆP CỬ NHÂN CNTT**

**TP. HCM, 2023**

**TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN**

**KHOA CÔNG NGHỆ THÔNG TIN**

**CHƯƠNG TRÌNH CHẤT LƯỢNG CAO**

**ĐẶNG NGUYỄN DUY – 19127372**

**NGUYỄN NHỰT LINH – 19127460**

**MÔ HÌNH GIÀN GIÁO NGỮ PHÁP CHO TÁC VỤ RÚT TRÍCH CẤU TRÚC ĐỐI SỐ VỊ NGỮ TRONG VĂN BẢN Y SINH**

**KHOÁ LUẬN TỐT NGHIỆP CỬ NHÂN CNTT**

**GIÁO VIÊN HƯỚNG DẪN**

**ThS. TUẤN NGUYÊN HOÀI ĐỨC**

**TP. HCM, 2023**

# **LỜI CẢM ƠN**

Trước tiên, chúng em xin gửi lời cảm ơn và biết ơn sâu sắc đến ThS. Tuấn Nguyên Hoài Đức – Giảng viên Chuyên ngành Hệ thống thông tin – Khoa Công nghệ thông tin – Trường Đại học Khoa học tự nhiên, người đã tận tình hướng dẫn, chỉ bảo, giúp đỡ em trong suốt thời gian em nghiên cứu khóa luận. Và cũng là người đưa ra những ý tưởng, kiểm tra sự phù hợp của luận văn.

Chúng em cũng xin gửi lời cảm ơn đến toàn thể các thầy cô Khoa Công nghệ thông tin trường Đại học Khoa học tự nhiên đã giảng dạy, và tạo điều kiện cho em trong quá trình học tập và nghiên cứu tại trường. Những kiến thức mà chúng em nhận được sẽ là hành trang giúp chúng em vững bước trong tương lai.

Bên cạnh đó, chúng em cũng xin gửi lời cảm ơn đến gia đình, bạn bè, người thân đã luôn ở bên để động viên và là nguồn cổ vũ lớn lao, là động lực giúp em hoàn thành luận văn này.

Do kiến thức và khả năng lý luận còn nhiều hạn chế nên khóa luận vẫn còn những thiếu sót nhất định. Em rất mong nhận được những đóng góp của các thầy giáo, cô giáo để khóa luận tốt nghiệp của em được hoàn thiện hơn.

Cuối cùng, em xin kính chúc các thầy cô giáo Ban lãnh đạo và các phòng ban chức năng Trường Đại học Khoa học tự nhiên dồi dào sức khỏe và luôn đạt được thành công trong sự nghiệp.

TP. Hồ Chí Minh, tháng 3 năm 2023

Nhóm sinh viên thực hiện

Đặng Nguyễn Duy – Nguyễn Nhựt Linh

# **MỤC LỤC**

[LỜI CẢM ƠN i](#_Toc131016202)

[MỤC LỤC ii](#_Toc131016203)

[Danh mục các bảng có trong khóa luận v](#_Toc131016204)

[Danh mục hình ảnh có trong khóa luận vi](#_Toc131016205)

[TÓM TẮT KHÓA LUẬN TỐT NGHIỆP vii](#_Toc131016206)

[Chương 1: GIỚI THIỆU ĐỀ TÀI 1](#_Toc131016207)

[1.1. Lý do chọn đề tài 1](#_Toc131016208)

[1.2. Giới thiệu đề tài 2](#_Toc131016209)

[1.3. Thách thức của bài toán và hướng giải quyết của đề tài 3](#_Toc131016210)

[1.4. Mục tiêu đề tài 5](#_Toc131016211)

[Chương 2: CẤU TRÚC ĐỐI SỐ VỊ NGỮ 6](#_Toc131016212)

[2.1. Cấu trúc đối số vị ngữ 6](#_Toc131016213)

[2.2. Cấu trúc đối số vị ngữ trong Y sinh 8](#_Toc131016214)

[2.3. Các khung đối số phổ biến 8](#_Toc131016215)

[2.4. Các khung đối số trong Y sinh 13](#_Toc131016216)

[2.4.1. BioProp 13](#_Toc131016217)

[2.4.2. PASBio 15](#_Toc131016218)

[2.4.3. GREC 21](#_Toc131016219)

[Chương 3: NHỮNG NGHIÊN CỨU VỀ RÚT TRÍCH ĐỐI SỐ VỊ NGỮ 24](#_Toc131016220)

[3.1. Hướng tiếp cận dựa luật (rule – base) và khớp mẫu (pattern matching) 24](#_Toc131016221)

[3.1.1. Hướng tiếp cận dựa luật (rule – base) 25](#_Toc131016222)

[3.1.2. Hướng tiếp cận khớp mẫu 26](#_Toc131016223)

[3.2. Hướng tiếp cận học máy 26](#_Toc131016224)

[3.2.1. Học có giám sát 27](#_Toc131016225)

[3.2.2. Học không giám sát 27](#_Toc131016226)

[3.2.3. Học bán giám sát 28](#_Toc131016227)

[3.3. Deep learning 29](#_Toc131016228)

[3.3.1. BIO 30](#_Toc131016229)

[3.3.2. SPAN 35](#_Toc131016230)

[3.3.3. Hướng tiếp cận nhận biết cú pháp 39](#_Toc131016231)

[4.1. Tổng quan phương pháp thực hiện 45](#_Toc131016232)

[4.2. Đánh giá mô hình học sâu của Zhixing Tan 46](#_Toc131016233)

[4.2.1. Giới thiệu tổng quan về mô hình được đề xuất bởi Zhixing Tan 46](#_Toc131016234)

[4.2.2. Cơ sở lý thuyết 46](#_Toc131016235)

[4.2.3. Mô hình được đề xuất 49](#_Toc131016236)

[4.3. Đánh giá mô hình học đa tác vụ chủ động do F.Ikhwantri đề xuất: 50](#_Toc131016237)

[4.3.1. Lý do lựa chọn mô hình 50](#_Toc131016238)

[4.3.2. Cơ sở lý thuyết 51](#_Toc131016239)

[4.4. Đề xuất mô hình “Rút trích đối số vị ngữ trong văn bản Y sinh” 53](#_Toc131016240)

[4.4.1. Cơ sở lý thuyết 53](#_Toc131016241)

[4.4.2. Giới thiệu mô hình 63](#_Toc131016242)

[4.4.3. Xây dựng mô hình 64](#_Toc131016243)

[Chương 5: THỰC NGHIỆM VÀ KẾT QUẢ 69](#_Toc131016244)

[5.1. Mô tả dữ liệu thực nghiệm 69](#_Toc131016245)

[5.1.1. Dữ liệu cho tác vụ SRL 69](#_Toc131016246)

[5.1.2. Bộ dữ liệu của tác vụ Grammar 69](#_Toc131016247)

[5.1.3. Bộ dữ liệu của tác vụ Dependency 69](#_Toc131016248)

[5.2. Kịch bản thử nghiệm 70](#_Toc131016249)

[5.2.1. Chia tập dữ liệu 70](#_Toc131016250)

[5.2.2. Phương pháp đánh giá 71](#_Toc131016251)

[5.2.3. Kết quả 72](#_Toc131016252)

[5.3. Thảo luận 77](#_Toc131016253)

[Chương 6: KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN 80](#_Toc131016254)

[6.1. Kết luận 80](#_Toc131016255)

[6.1.1. Các lý thuyết đã tìm hiểu 80](#_Toc131016256)

[6.1.2. Đóng góp của khóa luận 80](#_Toc131016257)

[6.2. Hướng phát triển 80](#_Toc131016258)

[TÀI LIỆU THAM KHẢO 81](#_Toc131016259)

[Phụ lục A: THÔNG SỐ CẤU HÌNH CHO MÔ HÌNH HỌC ĐA TÁC VỤ 90](#_Toc131016260)

# **Danh mục các bảng có trong khóa luận**

[Bảng 2-1: Khung đối số cho động từ "run" trong PropBank 10](#_Toc129379086)

[Bảng 2-2: Cấu trúc đối số vị ngữ của Probank, VerbNet và FrameNet 12](#_Toc129379087)

[Bảng 2-3: Khung động từ Y Sinh trong BioProp 15](#_Toc129379088)

[Bảng 2-4: Bốn nhóm động từ trong PASBio 15](#_Toc129379089)

[Bảng 2-5: Cấu trúc đối số của "mutate" trong PASBio và PropBank 17](#_Toc129379090)

[Bảng 2-6: Cấu trúc đối số của "block" trong PASBio và PropBank 18](#_Toc129379091)

[Bảng 2-7: Cấu trúc đối số của "confer" trong PASBio và PropBank 19](#_Toc129379092)

[Bảng 2-8: Cấu trúc đối số của "express" trong PASBio và PropBank 20](#_Toc129379093)

[Bảng 2-9: Nhãn ngữ nghĩa của GREC 23](#_Toc129379094)

[Bảng 4-1: Các loại cơ chế attention 48](#_Toc129811731)

[Bảng 5-1: Thống kê số lượng câu trong mỗi tập 70](#_Toc129811741)

[Bảng 5-2: Thống kê số lượng câu của từng động từ 71](#_Toc129811742)

[Bảng 5-3: Kết quả thực nghiệm đối với mô hình đơn tác vụ SRL trên tập DEV và TEST 72](#_Toc129811743)

[Bảng 5-4: Kết quả thực nghiệm đối với mô hình đa tác vụ SRL + Dependency trên tập DEV và TEST 73](#_Toc129811744)

[Bảng 5-5: Kết quả thực nghiệm đối với mô hình đa tác vụ SRL + Grammar trên tập DEV và TEST 74](#_Toc129811745)

[Bảng 5-6: Kết quả thực nghiệm đối với mô hình đa tác vụ SRL + Dependency + Grammar trên tập DEV và TEST 75](#_Toc129811746)

[Bảng 5-7: Thời gian huấn luyện và thời gian thực thi mô hình đơn tác vụ SRL 76](#_Toc129811747)

[Bảng 5-8: Thời gian huấn luyện và thời gian thực thi mô hình đa tác vụ SRL + Dependency 76](#_Toc129811748)

[Bảng 5-9: Thời gian huấn luyện và thời gian thực thi mô hình đa tác vụ SRL + Grammar 76](#_Toc129811749)

[Bảng 5-10: Thời gian huấn luyện và thời gian thực thi mô hình đa tác vụ SRL + Grammar + Dependency 76](#_Toc129811750)

# **Danh mục hình ảnh có trong khóa luận**

[Hình 3-1: Kiến trúc DB-LSTM 31](#_Toc131016050)

[Hình 3-2: Kiến trúc mô hình BERT cho tác vụ rút trích mối quan hệ 33](#_Toc131016051)

[Hình 3-3: Kiến trúc mô hình BERT cho tác vụ SRL 34](#_Toc131016052)

[Hình 3-4: Tổng quan kiến trúc của mô hình BiLSTM-span 36](#_Toc131016053)

[Hình 3-5: So sánh giữa biểu đồ khoảng đối số và BIO-base SRL 37](#_Toc131016054)

[Hình 3-6: Ví dụ cho việc phân tích cú pháp của một câu 38](#_Toc131016055)

[Hình 3-7: Ví dụ về dependency parsing tree 39](#_Toc131016056)

[Hình 3-8: Ví dụ về grammar parsing tree 39](#_Toc131016057)

[Hình 3-9: Dự đoán đối số và nhãn của nó dựa vào bộ mã hóa LSTM 40](#_Toc131016058)

[Hình 3-10: Tương quan giữa phụ thuộc ngữ nghĩa (trên) và cú pháp (dưới) 41](#_Toc131016059)

[Hình 3-11: Minh họa quá trình chuyển đổi SRL 🡪 Tree 42](#_Toc131016060)

[Hình 3-12: Minh họa quá trình chuyển đổi Tree 🡪 SRL 42](#_Toc131016061)

[Hình 3-13: Ví dụ về Full-C và SRL-C 44](#_Toc131016062)

[Hình 4-1: Mô hình self-attention 47](#_Toc131159988)

[Hình 4-2:Minh họa mô hình deep attentional neural network 49](#_Toc131159989)

[Hình 4-3: Kiến trúc. Soft Parameter Sharing 52](#_Toc131159990)

[Hình 4-4: Mô hình Hard Parameter Sharing 53](#_Toc131159991)

[Hình 4-5: Kiến trúc transformer 54](#_Toc131159992)

[Hình 4-6: Ví dụ biểu diễn input vector của các từ gần nghĩa với nhau trên hệ trục tọa độ 55](#_Toc131159993)

[Hình 4-7: Minh họa thành phần Positional Encoding 56](#_Toc131159994)

[Hình 4-8: Minh họa cho việc Tạo ra 3 vector đầu vào của Encoder 57](#_Toc131159995)

[Hình 4-9: Minh họa cách tạo self-attention layer 58](#_Toc131159996)

[Hình 4-10: Minh họa thành phần Residual 60](#_Toc131159997)

[Hình 4-11: Kiến trúc tổng quát của mô hình do khóa luận đề xuất 67](#_Toc131159998)

# **TÓM TẮT KHÓA LUẬN TỐT NGHIỆP**

Y Sinh là một ngành học cung cấp kiến thức về sức khỏe của con người. Hiện nay, lĩnh vực này ngày càng được phát triển mạnh mẽ, lý do cho sự phát triển này là nhờ những đóng góp không nhỏ trong khoa học sinh học. Cũng vì vậy mà nhu cầu khai thác kho tri thức của lĩnh vực Y Sinh phục vụ cho việc chăm sóc sức khỏe là ngày một cao.

Tuy nhiên, việc khai thác dựa vào sức người là không khả thi do khối lượng tri thức là quá lớn. Để giải quyết vấn đề này chúng ta cần sử dụng sức mạnh của máy tính để hỗ trợ cho việc khai khoáng dữ liệu lĩnh vực Y Sinh.

Trước khi máy tính có thể khai thác được thông tin của văn bản thì máy tính cần hiểu ngữ nghĩa trong câu mà văn bản đang đề cập đến. Để máy tính đọc hiểu được văn bản, trước tiên nó cần hiểu được sự kiện chính được nói đến trong mỗi câu. Đó chính là vai trò của tác vụ Gán nhãn Ngữ nghĩa cho văn bản. Vì vậy, khóa luận sẽ nghiên cứu và đề xuất một giải pháp gán nhãn ngữ nghĩa cho văn bản Y sinh.

Khóa luận đề xuất mô hình rút trích tự động Cấu trúc Đối số Vị ngữ cho văn bản thuộc lĩnh vực Y sinh sử dụng bộ ngữ liệu PASBio. Khóa luận giải quyết vấn đề thiếu ngữ liệu huấn luyện bằng cách sử dụng mô hình ngôn ngữ tiền huấn luyện BioBERT, đồng thời fine-tuning BioBERT để chuyên biệt hóa về bài toán Gán nhãn ngữ nghĩa cho lĩnh vực Y sinh. Nhận thấy tri thức ngữ pháp góp phần nâng cao hiệu suất cho mô hình, Khóa luận đã xây dụng giàn giáo ngữ pháp và sử dụng học đa tác vụ với tác vụ chính SRL và hai tác vụ phụ là Phân tích cú pháp và Phân tích quan hê phụ thuộc để đưa tri thức ngữ pháp vào mô hình.

# **Chương 1: GIỚI THIỆU ĐỀ TÀI**

* 1. **Lý do chọn đề tài**

Kỹ thuật Y sinh (Biomedical) là ngành khoa học ứng dụng Công nghệ sinh học để đưa ra các giải pháp trong lĩnh vực Y khoa, cũng như các mục đích chăm sóc sức khỏe con người khác [1]. Trong những năm qua, lĩnh vực này đã được mở rộng một cách đáng kể. Sự phát triển vượt bậc này có được nhờ những tiến bộ trong nghiên cứu, góp phần tạo ra các cơ hội mới để phát triển các công cụ chẩm đoán và điều trị bệnh cho con người [2]. Cũng vì lý do này mà kho tri thức của lĩnh vực Y sinh ngày càng được tích lũy và phát triển một cách đồ sộ.

Quá trình phát triển nhanh chóng của tri thức trong lĩnh vực này dẫn đến sự gia tăng về số lượng tài liệu Y sinh từ các nghiên cứu mới. Vì thế, nhu cầu về việc khai thác hợp lý kho tri thức dồi dào này để tìm kiếm tư liệu cho các công trình nghiên cứu ở từng lĩnh vực cụ thể và phục vụ cho việc chăm sóc sức khỏe là ngày một cao [3]. Nhưng với số lượng tri thức ngày một đồ sộ ấy thì việc khai thác thủ công là một điều bất khả thi, gây khó khăn trong việc rút trích nguồn dữ liệu này một cách hiệu quả nhất để phục vụ cho đời sống cũng như quá trình nghiên cứu. Từ đó, áp dụng máy tính trong nhiệm vụ khai khoáng kho tri thức Y sinh là một giải pháp bắt buộc. Máy tính có khả năng xử lý một cách chính xác với tốc độ nhanh. Nhờ vậy, máy tính có thể giúp con người thực hiện khai thác dữ liệu Y sinh trong khoảng thời gian có thể chấp nhận được với độ chính xác cao. Để máy tính có thể làm được điều này thì cần huấn luyện cho máy tính hiểu ngữ nghĩa của câu trong văn bản đang được khai thác. Vì vậy, khóa luận chọn nghiên cứu cũng như đề xuất một giải pháp rút trích ngữ nghĩa từ văn bản Y sinh.

* 1. **Giới thiệu đề tài**

Ngôn ngữ tự nhiên (natural language) là bất kỳ ngôn ngữ nào đã được phát triển tự nhiên, thông qua việc sử dụng và lặp lại mà không có kế hoạch hoặc dự định trước một cách có ý thức. Điển hình là một số ngôn ngữ mà con người sử dụng để có thể giao tiếp với nhau, dù là ngôn ngữ nói, ngôn ngữ ký hiệu, hay chữ viết. Những ngôn ngữ này khác với ngôn ngữ được xây dựng, chẳng hạn như ngôn ngữ lập trình [4]. Vì thế, đối với máy tính, hiểu được ngôn ngữ tự nhiên không hề đơn giản.

Trước khi ta có thể sử dụng máy tính để khai khoáng tri thức từ một văn bản bất kì, máy tính cần phải hiểu nội dung của từng câu chữ trong văn bản đang nói gì. Trong một câu, nội dung được thể hiện thông qua một động từ chính được gọi là vị ngữ (predicate) và những đối số liên quan đến ngữ nghĩa của động từ chính đó (argument). Cũng vì thế mà một trong những bài toán giúp máy tính thực hiện được nhiệm vụ đọc hiểu văn bản hiệu quả chính là bài toán rút trích Cấu trúc Đối số Vị ngữ (Predicate Argument Structure – PAS), hay còn có thể gọi là bài toán Gán nhãn Ngữ nghĩa (Semantic Role Labeling – SRL).

Trong lĩnh vực Xử lý ngôn ngữ tự nhiên (Natural Language Processing - NLP), Gán nhãn Ngữ nghĩa (Semantic Role Labeling - SRL) là một quá trình gán nhãn cho từ, hoặc cụm từ trong một câu để chỉ ra vai trò ngữ nghĩa của chúng trong câu. SRL đóng vai trò quan trọng trong việc máy tính đọc hiểu câu. Nó bao gồm việc phát hiện các đối số được kết hợp với vị ngữ (động từ) của một câu và phân loại chúng thành các vai trò cụ thể. Vì tầm quan trọng của tác vụ SRL, cũng như tính chất đặc thù của PAS khi áp dụng vào văn bản Y Sinh, khóa luận chọn tác vụ SRL trên văn bản Y Sinh làm đối tượng nghiên cứu của mình.

* 1. **Thách thức của bài toán và hướng giải quyết của đề tài**

Cấu trúc Đối số Vị ngữ (Predicate Argument Structure – PAS), là bộ khung ngữ nghĩa của câu, bởi nó truyển tải tất cả các sự kiện chính được đề cập trong một câu [5]. Rút trích được Cấu trúc Đối số Vị ngữ cũng đồng nghĩa với việc máy đã hiểu được sự kiện quan trọng mà câu đang muốn truyền tải. Bởi vì lý do này, bài toán rút trích Cấu trúc Đối số Vị ngữ (Predicate Argument Structure – PAS), hay còn gọi là bài toán Gán nhãn Ngữ nghĩa (Semantic Role Labeling – SRL) góp phần tạo tiền đề quan trọng cho một loạt các bài toán xử lý ngôn ngữ tự nhiên khác như rút trích sự kiện (event extraction), nhận diện thực thể (named-entity-recognition), hệ hỗ trợ ra quyết định (Decision Support System), tóm tắt văn bản (text summarization), …. [6] [7] [8] [9]

PAS phụ thuộc vào lĩnh vực của văn bản, cũng vì thế mà trong Y Sinh, cần xác định khung Đối số Vị ngữ khác biệt so với những lĩnh vực khác. Để có thể huấn luyện được máy tính hiểu được các sự kiện mà câu muốn truyền tải ở từng câu trong lĩnh vực Y Sinh, chúng ta cần một kho ngữ liệu có gắn nhãn Cấu trúc Đối số Vị ngữ với kích thước đủ lớn (ít nhất khoảng 3000 mẫu).

Tuy nhiên, thách thức với bài toán SRL cho văn bản Y Sinh đó chính là việc ngữ liệu gán nhãn sẵn vẫn còn hạn chế. Hiện tại không có kho ngữ liệu như vậy với kích thước cũng như độ phong phú ngữ nghĩa để các mô hình học máy có thể học một cách hiệu quả [10] [11] . Không những thế, việc xây dựng bộ ngữ liệu sử dụng phương pháp thủ công lại tốn nhiều nhân lực cũng như thời gian.

Bidirectional Encoder Representations from Transformers (viết tắt là BERT), là một dạng mô hình tiền huấn luyện (pre-train model) được phát triển bởi Jacob Devlin và các cộng sự từ Google và được công bố vào năm 2018 [12]. Mô hình BERT được huấn luyện từ BooksCorpus [13] với 800 triệu từ và Wikipedia Tiếng anh với 2500 triệu từ. Ngoài ra, ở mô hình BERT còn có một đặc điểm mà các mô hình trước đây chưa từng có đó là việc kết quả huấn luyện có thể fine-tuning được.

Với việc được huấn luyện trên một lượng dữ liệu lớn, mô hình BERT có thể khắc phục được vấn đề dữ liệu huấn luyện ít ỏi cho các bài toán hiện nay. Tuy nhiên, dữ liệu mà BERT sử dụng để huấn luyện là dữ liệu thuộc lĩnh vực tổng quát, bên cạnh đó mô hình này cũng chưa chuyên biệt vào một bài toán cụ thể. Vì vậy, muốn sử dụng BERT cho bài toàn Gán nhãn ngữ nghĩa cho văn bản Y sinh, Khóa luận áp dụng mô hình tương tự như BERT nhưng chuyên biệt vào lĩnh vực Y sinh đó chính là BioBert [14]. Bên cạnh đó, Khóa luận cũng sẽ fine-tuning mô hình BioBERT để chuyên biệt về bài toán Gán nhãn ngữ nghĩa.

BERT, hay BioBERT đã góp phần giải quyết thử thách về sự thiếu hụt dữ liệu để huấn luyện cho các mô hình. Hiện tại, có nhiều công trình đã chứng minh việc đưa tri thức ngữ pháp vào mô hình SRL giúp tăng hiệu quả của mô hình lên nhiều lần [15] [16] [17], … Dù vậy, việc đưa tri thức ngữ pháp một cách trực tiếp vào mô hình là một giải pháp đòi hỏi can thiệp vào kiến trúc bên trong mô hình, vì vậy mà phụ thuộc từng mô hình cụ thể và hạn chế về khả năng tổng quát hóa. Ngoài ra, can thiệp trực tiếp vào mô hình cũng là một thách thức khi phải đào sâu vào mã nguồn của những mô hình ngôn ngữ tiền huấn luyện đồ sộ và phức tạp. Cũng vì lý do này, khóa luận quyết định sử dụng phương pháp lập giàn giáo cú pháp cho tác vụ SRL thông qua học đa tác vụ (multi – task learning) để đưa tri thức ngữ pháp vào mô hình một cách gián tiếp mà không cần can thiệp sâu vào kiến trúc bên trong của BioBERT. Vì vậy, hướng tiếp cận này phần nào thể hiện tính chất Mô hình bất khả tri (Model agnostic), nhờ đó nó phù hợp để tổng quát hóa cho bất kỳ mô hình hộp đen nào Việc sử dụng học đa tác vụ giúp tiết kiệm tài nguyên tính toán bởi chúng ta không cần huấn luyện một cách riêng lẻ mỗi nhiệm vụ cho một mô hình mà có thể kết hợp các nhiệm vụ khác nhau vào một mô hình duy nhất. Bên cạnh đó, kết quả của mô hình được áp dụng multi – task learning cũng có độ chính xác cao hơn so với mô hình huấn luyện riêng lẻ bởi có sự hỗ trợ qua lại giữa các nhiệm vụ với nhau, tri thức được học từ nhiệm vụ này sẽ giúp ích cho nhiệm vụ khác. Một lợi ích quan trọng nữa của cách tiếp cận này là tri thức ngữ pháp chỉ đóng vai trò “giàn giáo” củng cố cho mô hình trong quá trình huấn luyện. Khi mô hình được huấn luyện xong, cũng như một ngôi nhà đã xây xong, thì “giàn giáo” sẽ được rút đi để giảm tải cho mô hình. Nhờ vậy, khi dùng mô hình của khóa luận (đã huấn luyện xong) để xử lý các văn bản mới thì hoàn toàn không cần khâu parsing để xây dựng cây ngữ pháp cho từng câu nhằm cung cấp dữ liệu đầu vào cho mô hình như nhiều nghiên cứu về nhúng ngữ pháp trực tiếp vẫn làm [15] [16] [18] [19]. Nói cách khác, chúng ta không cần tốn thêm thời gian và tài nguyên để phân tích ngữ pháp cho bất cứ văn bản nào trước khi đưa nó vào mô hình của khóa luận để gán nhãn ngữ nghĩa, nhưng vẫn có thể tận dụng tri thức ngữ pháp mà mô hình này đã có được từ quá trình huấn luyện bằng học đa tác vụ để nâng cao hiệu quả gán nhãn ngữ nghĩa.

* 1. **Mục tiêu đề tài**

Mục tiêu của đề tài là nghiên cứu cũng như xây dựng thử nghiệm mô hình DL dành cho bài toán SRL trên văn bản thuộc lĩnh vực Y Sinh cũng như phân tích, đánh giá kết quả mà mô hình đạt được. Các mục tiêu cụ thể như sau:

* Nghiên cứu cơ sở lý thuyết về PAS tổng quát cũng như trong lĩnh vực Y Sinh. Lựa chọn khung đối số và ngữ liệu phù hợp cho khóa luận.
* Giải quyết vấn đề thiếu ngữ liệu huấn luyện bằng cách sử dụng mô hình ngôn ngữ tiền huấn luyện thuộc lĩnh vực Y sinh là BioBERT
* Fine – tunning mô hình để chuyên biệt hóa về bài toán Gán nhãn ngữ nghĩa
* Sử dụng multi – task learning để đưa tri thức ngữ pháp vào bài toán SRL
* Phân tích, đánh giá độ hiệu quả của sự kết hợp giữa tác vụ chính SRL và hai tác vụ phụ là Phân tích ngữ pháp (grammar parsing), Phân tích quan hệ phụ thuộc (dependency parsing)

# **Chương 2: CẤU TRÚC ĐỐI SỐ VỊ NGỮ**

1. **Cấu trúc đối số vị ngữ**

Vị ngữ (Predicate) thường được đề cập với một trong hai ý nghĩa sau đây trong lý thuyết về ngữ pháp [20]. Đầu tiên, vị ngữ trong ngữ pháp truyền thống được xem như là một trong hai thành phần cơ bản của một câu, thành phần còn lại là chủ ngữ. Còn theo nghĩa thứ hai, vị ngữ của một câu ứng với vị từ (thường là động từ chính) và bất cứ thành phần nào bổ nghĩa cho vị từ.

Cấu trúc Đối số Vị ngữ (Predicate Argument Structure – PAS) có thể hiểu là một cấu trúc trong đó động từ trong câu đóng vai trò trung tâm, được gọi là vị ngữ; và các đối tượng liên quan xoay quanh vị ngữ gọi là đối số (chủ ngữ, túc từ, …). Mỗi đối số đều có kèm ngữ nghĩa của nó. Như vậy, bài toán Rút trích Cấu trúc Đối số Vị ngữ (PAS) có sự liên quan đến bài toán Gán nhãn ngữ nghĩa (SRL). Vì vậy, SRL và PAS được sử dụng là như nhau trong khóa luận này.

Như đã đề cập ở trên, vị ngữ là trung tâm cấu trúc của câu, chủ ngữ cũng chỉ đóng vai trò là đối số của vị ngữ nhưng là đối số quan trọng nhất. Mỗi vị ngữ khác nhau sẽ có đối số xoay quanh nó khác nhau tùy theo ngữ nghĩa của vị ngữ đó. Ví dụ:

* Vị ngữ không đòi hỏi đối số (arrive, cry, exist, …)
* Vị ngữ đòi hỏi một đối số (nội động từ)

VD: “The bus **stopped”**

Động từ **stopped** có một đối số bổ nghĩa cho nó

* Arg 0: Vai trò ngữ nghĩa: vật dừng lại – the bus
* Vị ngữ đòi hỏi hai đối số (ngoại động từ đơn)

VD: “Mary **moves** the chair”

Động từ **moves** có hai đối số bổ nghĩa cho nó

* Arg 0: Vai trò ngữ nghĩa: người thực hiện hành động di chuyển vật – Mary
* Arg 1: Vai trò ngữ nghĩa: vật được di chuyển – the chair
* Vị ngữ đòi hỏi ba đối số (ngoại động từ kép)

VD: “Kevin **sent** Shery a letter”

Động từ **sent** có ba đối số bổ nghĩa cho nó

* Arg 0: Vai trò ngữ nghĩa: người gửi - Kevin
* Arg 1: Vai trò ngữ ngữ nghĩa: người nhận – Sherry
* Arg 3: Vai trò ngữ nghĩa: vật được gửi – a letter
* Vị ngữ đòi hỏi bốn đối số (ngoại động từ kép có trạng từ đi kèm)

VD: “Adam will **give** Helen the report tomorrow morning”

Động từ **give** có bốn đối số bổ nghĩa cho nó

* Arg 0: Vai trò ngữ nghĩa: người cho - Adam
* Arg 1: Vai trò ngữ ngữ nghĩa: người nhận – Sherry
* Arg 3: Vai trò ngữ nghĩa: vật được cho– the report
* Arg 4: Vai trò ngữ nghĩa: thời gian cho– tomorrow moring

Thông thường, ta chỉ sử dụng những câu có hai hoặc ba đối số bổ nghĩa cho vị ngữ. Trường hợp trong câu không có hoặc có bốn đối số bổ nghĩa cho vị ngữ khá hiếm gặp trong thực tế, gần như chỉ tồn tại trong lý thuyết

Tóm lại, PAS đóng vai trò quan trọng trong bài toán xử lý ngôn ngữ tự nhiên nhờ vào đặc điểm làm nổi bật được quan hệ ngữ nghĩa giữa các thành phần trong câu. Chính vì thế, việc rút trích PAS từ văn bản thô luôn là vấn đề được quan tâm trong cộng đòng nghiên cứu.

1. **Cấu trúc đối số vị ngữ trong Y sinh**

Để hiểu rõ hơn về vai trò của PAS trong lĩnh vực Y Sinh, phân tích ví dụ sau đây:

“Both RAP1 and 2 are important vaccine candidates because it has been shown that merozoite invasion in vitro can be blocked by antibodies to RAP1”

Khi máy tính đọc hiểu câu ví dụ trên:

* Nếu không có sự hỗ trợ của PAS, do hệ thống máy tính lưu sẵn những bộ luật dựa trên ngữ nghĩa “nông” để làm căn cứ suy luận nên máy tính có thể sẽ xác nhận sự kiện chính trong câu là **shown,** dẫn tới việc xác định sai trọng tâm của câu
* Với sự hỗ trợ của PAS, hệ thống máy tính sẽ có thể xác định được sự kiện chính trong câu phải là **blocked** với các đối số là:
* Arg 0: Vai trò ngữ nghĩa: tác nhân ngăn cản - antibodies to RAP1
* Arg 1: Vai trò ngữ nghĩa: thực thể bị ngăn chặn - merozoite invasion in vitro

PAS trong lĩnh vực tổng quát không thể áp dụng vào lĩnh vực Y Sinh bởi sự khác biệt trong đặc thù về chuyên ngành. Nếu ta miễn cưỡng áp dụng PAS tổng quát vào lĩnh vực này sẽ dẫn đến việc hiểu sai ngữ nghĩa của câu. PAS trong lĩnh vực Y Sinh trong đại đa số trường hợp đều khác biệt so với PAS tổng quát (xin xem phần 2.4.2):

Nhận thấy sự khác biệt giữa PAS trong lĩnh vực Y Sinh và trong lĩnh vực tổng quát, một số bộ ngữ liệu PAS dành cho Y Sinh đã được xây dựng. Một số công trình nổi tiếng có thể kể đến như BioProp [21], PasBIO [5] và GREC [10].

1. **Các khung đối số phổ biến**

Hiện nay, có nhiều bộ ngữ liệu PAS với những hướng tiếp cận cũng như khung đối số khác nhau. Đối với lĩnh vực tổng quát, một số bộ ngữ liệu có thể kể đến như FrameNet [22], VerbNet [23] và PropBank [24]. Trong số đó, PropBank đưa ra định nghĩa cho bộ đối số một cách chi tiết và rõ ràng nhất cho từng vị ngữ.

**FrameNet** là một dự án nghiên cứu được phát triển bởi Charles J. Fillmore tại Berkeley’s International Computer Science Institute vào năm 1977. PAS trong FrameNet được định nghĩa dựa vào những gì mà người dung cũng như ứng dụng muốn để định nghĩa ra sự kiện cụ thể. Dữ liệu phân tích được từ FrameNet cho thấy câu “John sold a car to Mary” cơ bản mô tả tình huống tương tự với câu “Mary bought a car from John” chỉ là với một góc nhìn khác. FrameNet có chứa hơn 1200 khung ngữ nghĩa, 13000 đơn vị từ vựng và 202000 câu ví dụ. Dự án này đem lại sức ảnh hưởng lớn đến với cộng đồng nghiên cứu ngôn ngữ học và xử lý ngôn ngữ tự nhiên.

**VerbNet**, phát triển bởi Karrin Kipper-Schuler vào năm 2006 tại Đại học Pennsylvania. VerbNet được tổ chức thành các lớp động từ dựa vào lý thuyết thay thế của Levin (Levin, 1993), thông qua việc sàng lọc và bổ sung các lớp con để đạt được sự gắn kết về cú pháp và ngữ nghĩa giữa các thành viên của một lớp. VerbNet có 5800 động từ thuộc 270 nhóm động từ. Nó là một mạng động từ tiếng Anh trực tuyến liên kết cú pháp và ngữ nghĩa của chúng. VerbNet cũng có một phạm vi bao phủ rộng lớn, ánh xạ đến các bộ ngữ liệu khác như WordNet [25], PropBank và FrameNet.

Proposition Bank, hay **PropBank** là một kho văn bản được định nghĩa dựa vào vai trò ngữ nghĩa của từng động từ và được gán nhãn ngữ pháp theo bộ Penn [26]. Dự án PropBank áp dụng cách tiếp cận thực tế để biểu diễn ngữ nghĩa: thêm một lớp thông tin đối số vị ngữ hoặc nhãn vai trò ngữ nghĩa vào cấu trúc cú pháp của Penn Treebank. Dù PropBank được dùng để chỉ kho ngữ liệu do Martha Palmer và cộng sự đề xuất [27], thuật ngữ PropBank vẫn được sử dụng như một danh từ chung đề cập tới kho ngữ liệu nào được định nghĩa dựa trên động từ và đối số của chúng. Trong PropBank, một số động từ có nhiều nghĩa sẽ có nhiều hơn một khung PAS và đi kèm với điều này là các bộ đối số khác nhau liên quan đến ngữ nghĩa của nó.

**Bảng 2-1** mô tả các khung đối số khác nhau của động từ “**run**” tương ứng với từng ý nghĩa khác nhau.

* Ở cột (1), “**run**” mang ý nghĩa là vận hành (operate) hoặc tiến hành (proceed) và có các đối số: người vận hành (operator), thứ được vận hành (machine, procedure,…), người tuyển dụng (employer), đồng nghiệp (co-worker) và phương tiện (instrumental).
* Ở cột (2), “**run**” mang ý nghĩa là chạy và có các đối số: người chạy (runner) và các thông tin làm rõ nghĩa cho động từ chạy (course, race, distance, …) được gộp chung vào một đối số.
* Ở cột (3), “**run**” mang ý nghĩa đối mặt và có các đối số: người đối mặt (encounter) và vật đối mặt (thing encountered)

|  |  |  |
| --- | --- | --- |
| **(1)** | **(2)** | **(3)** |
| **PAS cho động từ:** RUN  **Ý nghĩa:** vận hành  **Đối số:**  0: người vận hành  1: thứ được vận hành  2: người tuyển dụng  3: đồng nghiệp  4: phương tiện  **Ví dụ**  [She]0 **runs** the [business]1 out of her home. | **PAS cho động từ:** RUN  **Ý nghĩa:** chạy  **Đối số:**  0: người chạy  1: cuộc thi chạy  **Ví dụ**  [John]0 **ran** [the Boston Marathon]1. | **PAS cho động từ:** RUN  **Ý nghĩa:** đối mặt  **Đối số:**  0: người đối mặt  1: vật đối mặt  **Ví dụ**  [He]0 **ran** into [problems with his dissertation]1 again and again |

Bảng 2-1: Khung đối số cho động từ "run" trong PropBank

Để hiểu rõ hơn về sự khác biệt giữa các bộ ngữ liệu PAS, **Bảng 2-2** mô tả cấu trúc đối số vị ngữ của PropBank, VerbNet và FrameNet

|  |  |  |
| --- | --- | --- |
| **PropBank** | **VerbNet** | **FrameNet** |
| **PAS cho động từ:** SELL  **Đối số:**  0: người bán  1: thứ được bán  2: người mua  3: giá bán  4: lợi nhuận  **Câu 1:**  [We]0 **sell** [these little old books]1 at [€2]3 each.  **PAS cho động từ:** RENT  **Đối số:**  0: người cho thuê  1: vật được cho thuê  2: người thuê  3: thuê  4: thời gian thuê  **Câu 2:**  [Mr. Tom]0 **rented** [us]2 [his spare room]1 in his house for [a month]4. | **PAS cho nhóm động từ**: GIVE  **Các động từ thành viên:**  give, sell, rent, render, refund, peddle, pass, loan, lend, lease  **Đối số:**  0: người quản lý.  1: chủ đề.  2: người nhận.  **Câu 1:**  [We]0 **sell** [these little old books]1 at €2 each.  **Câu 2:**  [Mr. Tom]0 **rented** [us]2 [his spare room]1 in his house for a month. | **PAS cho sự kiện:** Commerce\_sell  **Mô tả sự kiện:** giao dịch thương mại cơ bản từ quan điểm người bán.  **Động từ thành viên:** sell, rent, charge, lease, retail, vend  **Đối số**:  0: người bán.  1: hàng hóa.  **Câu 1:**  [We]0 **sell** [these little old books]1 at €2 each.  **Câu 2:**  [Mr. Tom]0 **rented** us [his spare room]1 in his house for a month. |

Bảng 2-2: Cấu trúc đối số vị ngữ của Probank, VerbNet và FrameNet

Tóm lại, ta có thể thấy, nếu VerbNet định nghĩa chung PAS cho các nhóm động từ có chung một cú pháp dựa vào lý thuyết của Levin, điều này giúp khung đối số của VerbNet đơn giản, dễ phát triển nhưng sẽ gặp khó khăn trong việc mô tả ngữ nghĩa một cách chi tiết khi gặp câu phức tạp. Còn đối với FrameNet, PAS được dựa vào sự kiện cụ thể, hoặc dựa vào yêu cầu mà người dùng muốn quan tâm. Cuối cùng là PropBank, từng động từ sẽ được gán nhãn một cách riêng biệt và chi tiết, điều này giúp ngữ nghĩa trong câu được thể hiện một cách rõ ràng hơn. Mặt khác, việc này khiến PropBank tốn kém nhiều thời gian hơn so với VerbNet khi phải tối ưu hóa cho từng động từ riêng biệt.

1. **Các khung đối số trong Y sinh**

Hiện nay, trong lĩnh vực Y Sinh, đã có nhiều khung đối số được xây dựng với nhiều hướng tiếp cận khác nhau. Từng công trình đều lựa chọn các động từ truyền tải sự kiện Y Sinh quan trọng như đột biến gen, mã hóa, biểu hiện tính trạng, … để xây dựng khung đối số cho các động từ ấy. Có thể kể đến một vài bộ ngữ liệu như BioProp, bộ ngữ liệu này vay mượn đối số của PropBank nên chưa được chuyên biệt vào lĩnh vực Y Sinh. Hay PASBio với số lượng đối số được gán nhãn còn quá nhỏ để được sử dụng vào huấn luyện. Với việc vị ngữ không chỉ gồm động từ chính trong câu mà còn có cả danh, động từ cùng với bộ đối số định nghĩa riêng cho lĩnh vực Y Sinh, GREC đã khắc phục được hạn chế của BioProp và PASBio, và cũng là bộ ngữ liệu mà khóa luận lựa chọn để sử dụng

1. **BioProp**

BioProp version 1.0 [28] là dự án được phát triển tại Academia Sinica, Đài Bắc, Đài Loan; bởi nhà nghiên cứu Wen-Lian Hsu cùng cộng sự vào năm 2009. Bộ ngữ liệu BioProp có 1635 câu được trích dẫn từ phần tóm tắt (abstract) của 500 bài báo về lĩnh vực Y Sinh viết bằng tiếng Anh. Các câu trích dẫn này được rút trích theo sự hướng dẫn của Penn Treebank II và được lưu trữ trong GENIA Treebank (GTB)

GTB là ngữ liệu được phát triển tại Phòng thí nghiệm Tsujii tại Đại học Tokyo, và là tập con của kho văn bản GENIA. Mục đích của dự án GENIA là tạo ra các công cụ cũng như tài nguyên giúp khai thác thông tin một cách tự động trong lĩnh vực Y Sinh. GENIA chính là kết quả của dự án này, nó là một bộ sưu tập bao gồm 2000 câu rút trích được gán nhãn cho lĩnh vực Y Sinh. BioProp version 1.0 được xây dựng nhờ vào việc bổ sung PropBank vào GTB.

Để phù hợp với các yêu cầu trong lĩnh vực Y Sinh, đối số của PropBank đã được chỉnh sửa để mô tả các vai trò ngữ nghĩa như là một thành phần của các sự kiện Y Sinh. Cụ thể, ba mươi động từ được chọn dựa trên tần suất sử dụng cũng như tầm quan trọng của chúng trong văn bản thuộc lĩnh vực Y Sinh. Vì mục tiêu trong khai thác thông tin là mối quan hệ của các thực thể có đặt tên (named entity), nên chỉ các câu có chứa tên của protein hoặc gen là được sử dụng để đếm tần suất xuất hiện của từng động từ. Các động từ thông thường sẽ được lọc ra, góp phần giúp tập trung vào động từ thuộc lĩnh vực Y Sinh. Ngoài ra, một vài động từ có tần số xuất hiện thấp nhưng lại có đống vai trò quan trọng trong việc mô tả mối quan hệ Y Sinh như “phosphorylate” và “transactivate” cũng được lựa chọn để giữ lại.

**Bảng 2-3** mô tả các động từ thuộc lĩnh vực Y Sinh được lựa chọn trong BioProp cũng như loại của chúng. Động từ trong BioProp được phân thành bốn loại khác nhau, trong đó:

* Loại 1: động từ có vai trò quan trọng trong văn bản thuộc lĩnh vực Y Sinh, nhưng ít khi xuất hiện trong văn bản thuộc lĩnh vực tổng quát nên không được định nghĩa trong VerbNet
* Loại 2: động từ có xuất hiện trong VerbNet, có ý nghĩa Y Sinh và bộ khung không xác định
* Loại 3: động từ có xuất hiện trong VerbNet, nhưng khi áp dụng trong lĩnh vực Y Sinh lại mang ý nghĩa khác
* Loại 4: động từ mang ý nghĩa gần như tương tự trong lĩnh vực Y Sinh và lĩnh vực tổng quát

|  |  |
| --- | --- |
| **Loại** | **Danh sách động từ** |
| 1 | encode, interact, phosphorylate, transactivate |
| 2 | express, modulate |
| 3 | bind |
| 4 | activate, affect, alter, associate, block, decrease differentiate, encode, enhance, increase, induce, inhibit, mediate, mutate, prevent, promote, reduce, regulate, repress, signal, stimulate, suppress, transform, trigger |

Bảng 2-3: Khung động từ Y Sinh trong BioProp

1. **PASBio**

PASBio là công trình được phát triển bởi ba nhà nghiên cứu Tuangthong Wattarujeekrit, Parantu Shah và Nigel Colliertại Tokyo, Nhật Bản từ 2002 và được phát hành năm 2005. PASBio đã khắc phục được hạn chế của BioProp bằng cách chỉnh sửa toàn bộ khung đối số của PropBank cho phù hợp với lĩnh vực Y Sinh. PASBio có tổng cộng khoảng 29 vị ngữ được chia thành bốn nhóm khác nhau cụ thể như sau:

|  |  |
| --- | --- |
| **Nhóm** | **Động từ thuộc mỗi nhóm** |
| A – Cùng ý nghĩa, thiếu đối số | alter, begin, develop, disrupt, inhibit, initiate, mutate, proliferate, skip. |
| B – Cùng ý nghĩa, dư đối số | generate, block, decrease, lose, modify. |
| C – Cùng ý nghĩa, cùng cấu trúc | abolish, confer, eliminate, lead to, result, delete. |
| D – Khác ý nghĩa | splice, express, truncate, translate, encode, transform, catalyze, transcribe, recognize. |

Bảng 2-4: Bốn nhóm động từ trong PASBio

1. **Nhóm A**

Động từ thuộc nhóm này khi được sử dụng trong lĩnh vực Y Sinh mang ý nghĩa gần như tương tự với động từ sử dụng trong lĩnh vực tổng quát. Nhưng, cần nhiều đối số hơn ở khung ngữ nghĩa để phù hợp với lĩnh vực Y Sinh

Xét vị ngữ “**mutate**”, trong lĩnh vực tổng quát vị ngữ này có nghĩa là thay đổi với hai đối số: (Arg0) tác nhân gây thay đổi, (Arg1) thực thể bị thay đổi. Nhưng trong lĩnh vực Y Sinh, “mutate” lại mang ý nghĩa là đột biến. Vì thế, PASBio đã thay đối lại câu trúc đối số của vị ngữ này để phù hợp hơn về ngữ nghĩa khi áp dụng vào văn bản Y Sinh.

**Bảng 2-5** so sánh sự khác nhau giữa khung đối số vị ngữ “**mutate**” trong PASBio và PropBank

|  |  |
| --- | --- |
| **Động từ “mutate”** | |
| **Đối số trong sinh học** | **Đối số của PropBank** |
| **Arg 0**: Vị trí exon hoặc nitron nơi xảy ra đột biến  **Arg 1**: Gene bị đột biến  **Arg 2**: Thay đổi ở mức độ phân tử  **Arg 4:** Thay đổi ở mức độ kiểu hình | **Arg 0:** Tác nhân gây ra sự thay đổi  **Arg 1:** Thực thể bị thay đổi |
| **Ví dụ** | |
| Mutating amino acids EE (437 and 438) to AA has generated the DNA-binding mutant (Stat5aEE-AA) of Stat5a   * **Arg 0**: - * **Arg 1**: amino acids EE (437 and 438) * **Arg 2**: AA * **Arg 3**: the DNA-binding mutant (Stat5aEE-AA) of Stat5a * **Arg 4**: - | |

Bảng 2-5: Cấu trúc đối số của "mutate" trong PASBio và PropBank

1. **Nhóm B**

Về cơ bản các động từ thuộc nhóm này khá giống với trường hợp ở nhóm A, điều khác biệt là động từ ở nhóm B được sử dụng trong văn bản Y Sinh có ngữ nghĩa tương tự như trong PropBank nhưng có một vài đối số không phù hợp với ngữ cảnh sinh học. Vì thế, PASBio thay đổi lại khung đối số và chỉ thêm vào những đối số cần thiết.

Xét vị ngữ “**block**”, trong lĩnh vực tổng quát, vị ngữ này mang ý nghĩa là chặn với bốn đối số là (Arg 0) tác nhân, (Arg 1) hành động hoặc đối tượng bị chăn, (Arg 2) sở hữu, (Arg 3) phương tiện. Đối với ngữ cảnh sinh học, vị ngữ này có nghĩa là ngăn chặn với hai đối số là (Arg 0) tác nhân, (Arg 1) thực thể hoặc quá trình bị ngừng lại

**Bảng 2-6** so sánh sự khác nhau giữa khung đối số vị ngữ “**block**” trong PASBio và PropBank

|  |  |
| --- | --- |
| **Động từ “block”** | |
| **Đối số trong sinh học** | **Đối số của PropBank** |
| **Arg 0**: Tác nhân  **Arg 1**: Thực thể hoặc quá trình bị chặn | **Arg 0:** Tác nhân  **Arg 1:** Hành động hoặc đối tượng bị chặn  **Arg 2:** Sở hữu  **Arg 3:** Phương tiện |
| **Ví dụ** | |
| Labeling with antibodies to the Y553 phosphopeptide is specific since the action of a postulated repressor is blocked by Alanine.   * **Arg 0**: Alanine * **Arg 1**: the action of a postulated repressor | |

Bảng 2-6: Cấu trúc đối số của "block" trong PASBio và PropBank

1. **Nhóm C**

Động từ thuộc nhóm này mang ngữ nghĩa tương tự nhau ở cả trong lĩnh vực Y Sinh cũng như trong lĩnh vực tổng quát. Vì thế, khung đối số ở các động từ này trong PASBio và PropBank gần như là tương tự nhau về số lượng và cả ý nghĩa

**Bảng 2-7** so sánh sự khác nhau giữa khung đối số vị ngữ “**confer**” trong PASBio và PropBank

|  |  |
| --- | --- |
| **Động từ “confer”** | |
| **Đối số trong sinh học** | **Đối số của PropBank** |
| **Arg 0**: Tác nhân  **Arg 1**: Tính chất sinh học  **Arg 2:** Đơn vị nhận tính chất sinh học | **Arg 0:** Người cho  **Arg 1:** Vật được cho  **Arg 2:** Người được cho |
| **Ví dụ** | |
| Nonsense or frameshift alterations in the coding sequence or by splice variants were not able to confer loss-of-function mutations   * **Arg 0**: Nonsense or frameshift alterations in the coding sequence or by splice variants * **Arg 1**: loss-of-function mutations | |

Bảng 2-7: Cấu trúc đối số của "confer" trong PASBio và PropBank

1. **Nhóm D**

Động từ thuộc nhóm này mang ý nghĩa khác biệt hoàn toàn giữa lĩnh vực Y Sinh và lĩnh vực tổng quát. Cũng vì lý do này mà PASBio phải tạo một khung đối số riêng biệt cho những động từ này do khung đối số PropBank đã không còn phù hợp

Xét ví dụ vị ngữ “express”, trong lĩnh vực tổng quát, vị ngữ này mang ý nghĩa là biểu lộ hoặc chuyển phát nhanh. Còn đối với ngữ cảnh trong Y Sinh, vị ngữ này lại mang ý nghĩa là biểu hiện sự tồn tại của gene hoặc sản phẩm gene

**Bảng 2-8** so sánh sự khác nhau giữa khung đối số vị ngữ “**express**” trong PASBio và PropBank

|  |  |
| --- | --- |
| **Động từ “express”** | |
| **Đối số trong sinh học** | **Đối số của PropBank** |
| **Arg 0**: Tên thực thể  **Arg 1:** Tính chất của thực thể  **Arg 2:** Bộ phận hoặc cơ quan mang biểu hiện | EXPRESS – 01  Ý nghĩa: nói chuyện  **Arg0:** Người nói  **Arg1:** Nội dung  **Arg2:** Người nghe  EXPRESS – 02  Ý nghĩa: chuyển phát nhanh  **Arg0:** Người gửi  **Arg1:** Vật gửi  **Arg2:** Người nhận |
| **Ví dụ** | |
| BNIP3alpha was shown to be cloned and expressed in in most human tissues, and two alternative precursors of the protein have been imported into isolated mitochondria and processed to the mature protein   * **Arg 0**: BNIP3alpha * **Arg 1**: in most human tissues | |

Bảng 2-8: Cấu trúc đối số của "express" trong PASBio và PropBank

Bộ ngữ liệu mà khóa luận lựa chọn sử dụng là PASBio+. Tiền thân của bộ ngữ liệu này là 317 câu đã được gán nhãn của bộ ngữ liệu PASBio. Với việc sử dụng PASBio+, khóa luận có được ngữ liệu huấn luyện gán nhãn PASBio kích thước hơn 10000 câu. PASBio+ cũng được gia tăng độ phong phú của mẫu câu bằng các biến thể ngữ pháp của các câu gốc, giúp ngữ liệu có được độ phủ rộng trên các cách hành văn tự nhiên đa dạng. Chi tiết về bộ ngữ liệu sẽ được trình bày ở chương 5.

1. **GREC**

GREC là một kho ngữ liệu được nghiên cứu và phát triển bởi Paul Thompson và các cộng sự của mình tại Trung tâm khai thác văn bản quốc gia (NaCTeM), thuộc khoa Khoa học Máy tính, đại học Manchester tại Anh. Công trình được chỉnh sửa, bổ sung và hoàn thiện từ năm 2008 đến tháng 8 năm 2011.

Mục tiêu của GREC chính là gán nhãn đối số vị ngữ cho sự kiện được nêu ra trong một câu, với vị ngữ được thể hiện qua động từ (verb) và động từ danh nghĩa (nominalised verb). Không những thế, GREC còn được gán nhãn thực thể có tên (nhãn NER – Named Entity Recognition).

Bộ ngữ liệu GREC được gán nhãn một cách thủ công về mặt ngữ nghĩa dựa vào 240 bản tóm tắt MEDLINE (167 chủ đề về E. Coli và 73 chủ đề về loài Người).

Kho ngữ liêu của GREC bao gồm có 677 bản tóm tắt và tổng cộng có 4770 sự kiện khác nhau. Ngoài ra GREC còn được tác giả chia thành 13 đối số, cụ thể được mô tả ở **Bảng 2-9**

|  |  |  |
| --- | --- | --- |
| **Nhãn ngữ nghĩa** | **Ý nghĩa** | **Ví dụ (tiếng Anh)**  **\*in đậm:** cụm được gán nhãn  *\*in nghiêng:* vị ngữ chính |
| AGENT | Đối tượng gây ra/xúc tác phản ứng | **The narL gene product** *activates* the nitrate reductase operon. |
| THEME | Ảnh hưởng/kết quả bởi phản ứng.  Trạng thái phản ứng. | **recA protein** was *induced* by UV radiation.  **The FNR** **protein** *resembles* CRP. |
| MANNER | Phương pháp/cách phản ứng xảy ra | cpxA gene *increases* the levels of csgA transcription by **dephosphorylation** of CpxR. |
| INSTRUMENT | Chất xúc tác phản ứng | EnvZ *functions* through **OmpR** to control NP porin gene expression in Escherichia coli K-12. |
| LOCATION | Nơi diễn ra phản ứng | Phosphorylation of OmpR *modulates* expression of the ompF and ompC genes in **Escherichia coli.** |
| SOURCE | Điểm bắt đầu | A transducing lambda phage was *isolated* from **a strain** harboring a glpD’’lacZ fusion. |
| DESTINATION | Điểm kết thúc | Transcription of gntT is activated by *binding* of the cyclic AMP (cAMP)-cAMP receptor protein (CRP) complex to **a CRP binding site.** |
| TEMPORAL | Liên quan với phản ứng khác | The Alp protease activity is *detected* in cells **after introduction** of plasmids carrying the alpA gene. |
| CONDITION | Điều kiện hay sự thay đổi điểu kiện để xảy ra phản ứng | Strains carrying a mutation in the crp structural gene fail to *repress* ODC and ADC activities in response to **increased cAMP**. |
| RATE | Sự thay đổi số liệu | marR mutations *elevated* inaA expression by **10-to 20-fold** over that of the wild-type. |
| DESCRIPTIVE- AGENT | Thông tin của AGENT | It is likely that HyfR *acts* as **a formate-dependent regulator** of the hyf operon. |
| DESCRIPTIVE- THEME | Thông tin của THEME | The FNR protein *resembles* **CRP**. |
| PURPOSE | Mục đích/lý do sự kiện xảy ra | The fusion strains were *used* **to study** the regulation of the cysB gene by assaying the fused lacZ gene product. |

Bảng 2-9: Nhãn ngữ nghĩa của GREC

# **Chương 3: NHỮNG NGHIÊN CỨU VỀ RÚT TRÍCH ĐỐI SỐ VỊ NGỮ**

Rút trích Cấu trúc Đối số Vị ngữ chủ yếu được dùng để giúp máy tính hiểu được vai trò ngữ nghĩa của các từ trong một câu [29], điều này góp phần mang lại lợi ích cho lĩnh vực Xử lý ngôn ngữ tự nhiên khi bài toán này không chỉ cần hiểu ý nghĩa của mỗi từ mà còn là cách chúng được sử dụng trong những câu khác nhau [30].

Nhờ vào những đóng góp quan trọng với lĩnh vực Xử lý ngôn ngữ tự nhiên mà hiện nay, có rất nhiều công trình nghiên cứu về bài toán Rút trích Cấu trúc Đối số Vị ngữ. Nhìn chung, các công trình trên tiếp cận bài toán theo một trong ba hướng chính sau:

* Hướng tiếp cận dựa luật (rule – base) và khớp mẫu (pattern matching)
* Hướng tiếp cận học máy (machine learning)
* Hướng tiếp cận học sâu (deep learning)

Mỗi hướng tiếp cận sẽ có từng đặc điểm khác nhau. Ở chương này, Khóa luận sẽ trình bày về từng hướng tiếp cận cũng như ưu điểm, khuyết điểm của chúng. Học sâu là một nhánh của học máy, nhưng do có mức độ phát triển vượt bậc nên khóa luận sẽ trình bày về Học sâu như một hướng tiếp cận độc lập.

1. **Hướng tiếp cận dựa luật (rule – base) và khớp mẫu (pattern matching)**

Dựa luật (rule – base) và Khớp mẫu (pattern matching) là hai hướng tiếp cận cổ điển và xa xưa trong lịch sử về việc giải quyết bài toán gán nhãn ngữ nghĩa. Mặc dù ở bối cảnh hiện nay, đã có nhiều hướng tiếp cận mới đem lại nhiều ưu điểm nổi bật (hướng tiếp cận học máy, học sâu, …). Song, chúng ta vẫn không thể phủ nhận những đóng góp mà chúng mang lại trong việc giải quyết bài toán gán nhãn ngữ nghĩa cho văn bản.

1. **Hướng tiếp cận dựa luật (rule – base)**

Đây là hướng tiếp cận sớm nhất trong việc giải quyết các bài toán trong lĩnh vực Gán nhãn ngữ nghĩa cho văn bản. Hướng tiếp cận dựa luật sử dụng bộ luật được viết thủ công bởi các chuyên gia để nhận biết vị ngữ cũng như đối số của nó trong văn bản, từ đó gán nhãn ngữ nghĩa cho từng đối số đó.

Một số công trình tiêu biểu cho hướng tiếp cận này có thể kể đến như: công trình *Head – Driven Phrase Structure Grammar (HPSG)* [31] đã đề xuất thuyết Ngữ pháp Dẫn xuất Cấu trúc Đầu tố ngữ cũng như xây dựng bộ luật viết tay từ HPSG cho việc rút trích tự PAS; công trình *From Tree to Predicate Agrument Structure* [32]đã khai thác trực tiếp Penn Treebank [26] để xây dựng luật cho việc rút trích PAS, công trình góp phần khắc phục các trường hợp đối số rỗng như trong câu khuyết túc từ hoặc chủ ngữ ngầm định; *Hệ thống cơ sở của CoNLL 2004* [33] *và CoNLL 2005* [34] xử lý SRL bằng tập 7 luật kinh nghiệm (heuristic rule) đơn giản. Đối với lĩnh vực Y sinh, nhiều công cụ SRL đã được xây dựng [35] [36] [37], sử dụng luật heuristic dựa vào cây cú pháp để rút trích Cấu trúc đối số vị ngữ.

Hạn chế của hướng tiếp cận này nằm ở việc các bộ luật cần được viết thủ công bởi các chuyên gia ngôn ngữ, vì thế khi chuyển sang lĩnh vực hay ngôn ngữ khác thì cần xây dựng lại bộ luật mới. Ngoài ra, mặt dù mang lại độ chính xác cao nhưng độ bao phủ của hướng tiếp cận này cũng không cao, do các chuyên gia không thể bao hàm hết tát cả cấu trúc ngữ pháp có thể có trong văn bản. Tuy nhiên, đây cũng là ưu điểm của phương pháp này khi nó có thể hoạt động tốt với những ngôn ngữ không thông dụng hoặc các lĩnh vực hạn chế về tài nguyên ngôn ngữ khi mà khối lượng ngữ liệu là quá ít cho việc huấn luyện máy tính. Ví dụ như các công trình SRL cho tiếng Nhật, tiếng Hà Lan [38] [39].

1. **Hướng tiếp cận khớp mẫu**

Hướng tiếp cận này chủ yếu thịnh hành vào thập niên 90. Hướng khớp mẫu sử dụng mẫu được định nghĩa sẵn để so khớp với văn bản, từ đó có thể rút trích được vị ngữ cũng như đối số của nó. Hướng tiếp cận này có hai phương pháp thực hiện: mẫu được viết thủ công bởi chuyên gia hoặc được khai thác tự động từ dữ liệu.

Hầu hết các công trình thuộc lĩnh vực tổng quát, bộ mẫu có được là nhờ vào việc khai thác tự động từ dữ liệu. Ví dụ như hệ thống AutoSlog và AutoSlog – TS của E.Riloff cùng cộng sự đã rút trích các sự kiện từ văn bản mô tả các cuộc tấn công khủng bố. Đối với lĩnh vực Y sinh, do kích thước ngữ liệu còn hạn chế, nên các công trình cần chuyên gia xây dựng thủ công bộ mẫu [35] [40].

Đối với phương pháp mời chuyên gia viết bộ mẫu thủ công, tuy mang lại độ chính xác cao nhưng cũng tương tự với hướng dựa luật, phương pháp này cũng tốn kém chi phí và độ bao phủ không cao nên chỉ phù hợp với các lĩnh vực hoặc ngôn ngữ hạn chế về kích thước ngữ liệu. Đối với phương pháp khai thác mẫu tự động từ dữ liệu, hạn chế của phương pháp này nằm ở việc khó có thể kiểm soát các mẫu thu được do độ nhiễu cao. Cũng vì lý do này mà phương pháp này cần có sự kết hợp của phương pháp thủ công để rà soát lại bộ mẫu.

1. **Hướng tiếp cận học máy**

Học máy (machine learning) là hướng tiếp cận mới hơn so với hai hướng tiếp cận nêu trên (trong đó học sâu – deep learning là kỹ thuật mới nhất), nhưng hướng tiếp cận này mang lại hiệu quả cao hơn so với hướng dựa luật và khớp mẫu. Học máy là một lĩnh vực của trí tuệ nhân tạo (AI) liên quan đến việc nghiên cứu và xây dựng kỹ thuật cho phép máy tính có khả năng “học” để cải thiện chính nó dựa vào mẫu (training data) hoặc dựa vào kinh nghiệm đã được học (experience).

Các thuật toán học máy được chia làm 3 nhóm dựa trên phương thức học: học có giám sát (supervised learning), học không giám sát (unsupervised learning), và học bán giám sát (semi – supervised learning)

1. **Học có giám sát**

Học có giám sát (supervised learning) là một kỹ thuật của ngành học máy trong đó, máy tính được “học” bằng cách dựa vào bộ dữ liệu đào tạo đã được gán nhãn (well labelled training data) để máy tính dự đoán đầu ra. Dữ liệu gán nhãn là dữ liệu đầu vào (input data) đã được gắn thẻ với đầu ra chính xác.

Học máy có giám sát cần sử dụng bộ ngữ liệu có kích thước đủ lớn và đã có gán sẵn nhãn ngữ nghĩa để có thể huấn luyện máy tính. Một số công trình trong lĩnh vực tổng quát đã sử dụng bộ ngữ liệu Propbank, FrameNet, Penn Treebank [41] [42] [43] [44] để huấn luyện hệ thống SRL. Đối với lĩnh vực Y sinh, vào năm 2006, khi bộ ngữ liệu BioProp được xây dựng, BIOSMILE [45] là công trình SRL hoàn chỉnh đầu tiên sử dụng bộ ngữ liệu này và được huấn luyện bằng MaxEnt.

Nhược điểm của phương pháp này là việc xây dựng bộ ngữ liệu đã được gán nhãn sẵn tốn nhiều chi phí cũng như công sức. Từ đó, học máy bán giám sát hoặc học máy không giám sát được các công trình sử dụng như một giải pháp để giải quyết vấn đề này.

1. **Học không giám sát**

Không giống như phương pháp học máy có giám sát khi dữ liệu huấn luyện có vai trò như “người giám sát” để dạy cho máy tính dự đoán đầu ra một cách chính xác. Các mô hình của phương pháp học máy không giám sát (unsupervised learning) không được giám sát bằng tập dữ liệu huấn luyện mà chúng sẽ tự tìm ra mẫu ẩn và thông tin từ dữ liệu đã cho.

Năm 2021 [46] đã sử dụng kết hợp phương pháp phân tích thành phần chính (principal component analysis) trên các biến của bộ dữ liệu COVID – 19 để giảm kích thước cũng như tìm ra biến quan trọng nhất và học máy không giám sát để tìm ra các cấu trúc cộng động (community structure) bị ẩn của tập dữ liệu. Một số công trình như [47] [48] [49] cũng đã sử dụng phương pháp học không giám sát để giải quyết bài toán SRL. Trong lĩnh vực y sinh, có thể kể đến một số công trình tiêu biểu như [50] [51] [52].

1. **Học bán giám sát**

Học máy bán giám sát (semi – supervised learning) là phương pháp đứng giữa học có giám sát (toàn bộ dữ liệu huấn luyện đều được gán nhãn) và học không giám sát (dữ liệu huấn luyện không được gán bất kỳ nhãn nào). Học bán giám sát sử dụng cả dữ liệu đã và chưa gán nhãn để thực hiện huấn luyện, điển hình là một lượng nhỏ dữ liệu đã được gán nhãn cùng với lượng lớn dữ liệu chưa được gán nhãn.

Một số công trình đã sử dụng phương pháp học bán giám sát để đề xuất giải pháp cho bài toán SRL, có thể kể đến như [53] [54]. Hai công trình này rút trích PAS bằng cách lặp lại quá trình chọn ứng viên trên bộ dữ liệu thô, bắt đầu từ một vài PAS làm hạt giống ứng viên. Mặc dù [53] [54] sử dụng phương pháp học không giám sát nhưng lại dựa trên bộ dữ liệu Verbnet – bộ tài nguyên không kém gì dữ liệu được gán nhãn. Nên có thể coi hai công trình này đã áp dụng phương pháp học bán giám sát.

Ưu điểm của phương pháp này là không tốn chi phí cho chuyên gia trong việc xây dựng bộ ngữ liệu gán nhãn sẵn, tạo thuận lợi khi muốn chuyển sang lĩnh vực mới. Bên cạnh đó, nhược điểm của học bán giám sát nằm ở việc lặp lại và phân kì của các thực thể ứng viên làm cho độ chính xác không tốt khi so sánh với học có giám sát.

1. **Deep learning**

Ban đầu, các nhà nghiên cứu trong lĩnh vực trí tuệ nhân tạo (artificial intelligence) đã thử nghiệm việc mô phỏng trí thông minh của con người trong một vài công việc cụ thể, ví dụ như chơi game. Họ đã đưa ra một lượng lớn những quy tắc mà máy phải tuân theo. Máy tính cần có một danh sách cụ thể các hành động có thể được thực hiện và nó cần quyết định đưa ra hành động nào dựa trên những quy tắc này.

Học máy (machine learning) lại hướng đến việc máy tính có thể học cách sử dụng lượng lớn các tập dữ liệu (dataset) thay vì sử dụng các quy tắc được quy định sẵn. Học máy cho phép máy tính cải thiện bằng cách học từ chính nó. Phương pháp học này lợi dụng sức mạnh xử lý lượng lớn dữ liệu từ các máy tính hiện đại.

Học sâu (hay deep learning) là một nhánh nghiên cứu chuyên sâu của học máy, phương pháp này dựa vào mạng thần kinh nhân tạo (artificial neural network – ANN) với nhiều lớp. Vì ANN mô phỏng lại cách thức hoạt động neuron thần kinh trong bộ não của con người, nên học sâu cũng là một phương pháp lấy cảm hứng từ cấu trúc và chức năng của bộ não con người.

Các mô hình học sâu được thiết kế để học từ lượng lớn dữ liệu theo cả hai cách học có giám sát hoặc không giám sát. Các kiến trúc thường được sử dụng trong học sâu có thể kể đến như mạng neuron chuyển tiếp (Feed Forward Neuron Network – FFNN), mạng neuron tích chập (Convolutional Neural Network – CNN), mạng neuron hồi quy (Recurrent Neural Network – RNN).

Năm 2012, tại cuộc thi nhận dạng ảnh quy mô lớn (ImageNet Large-Scale Visual Recognition), Alex Krizhevsky cùng cộng sự đã tham gia và giành được chiến thắng với kết quả top 5 error rate 16%, làm sững sờ giới nghiên cứu lúc đó. Mô hình này là một CNN và cũng là mô hình Deep CNN đầu tiên, sau này nó được gọi là AlexNet [55].

Nhân tố quan trọng góp phần giúp cho AlexNet thành công chính là nhờ vào việc sử dụng GPU để huấn luyện mô hình. GPU với khả nặng chạy song song nhiều lõi, giúp tăng tốc thuật toán lên rất nhiều lần khi so với CPU. Đây là một bước đột phá lớn, góp phần giúp cho học sâu phát triển và mang lại nhiều thành tựu. Sau AlexNet, các mô hình giành giải cao trong những năm sau đó đều là các deep networks: ZFNET (2013), GoogLeNet (2014), VGG (2014), ResNet (2015). Các công ty về công nghệ cũng đã bắt đầu để ý đến việc phát triển phòng nghiên cứu về học sâu trong khoảng thời gian này. Cũng từ năm 2012, số lượng bài báo khoa học, công trình nghiên cứu về học sâu cũng tăng lên theo hàm số mũ, blog về DL cũng gia tăng theo từng ngày.

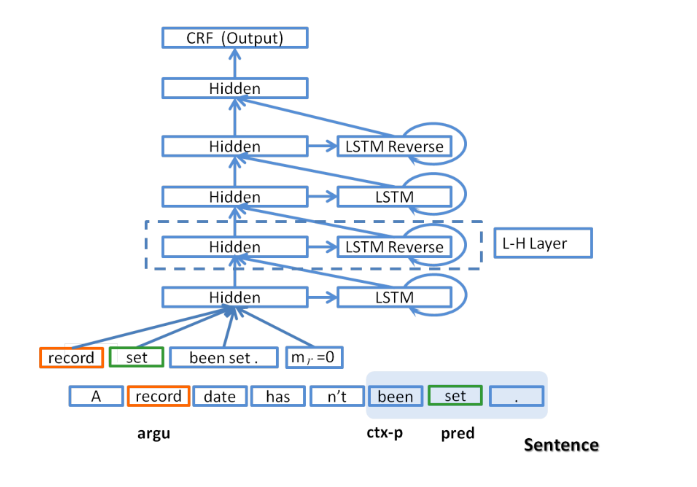
Ở thời điểm hiện tại, học sâu được chia làm 3 hướng nghiên cứu: BIO, SPAN và hướng nhận biết cú pháp.

1. **BIO**

BIO (hoặc IOB) [56] là từ viết tắt của B – Begin (bắt đầu), I – Inside (bên trong) và O – Outside (bên ngoài) là một định dạng thường được dùng để gắn nhãn các mã thông báo (token) trong các bài toán phân tích thành phần câu hoặc thành phần văn bản.

Năm 2015, trong khi các hệ thống SRL thành công đều được xây dựng dựa vào kết quả của việc phân tích cú pháp [57] [58] [59], và các nỗ lực trong việc xây dựng hệ thống SRL end-to-end mà không sử dụng kiến thức ngữ pháp không đạt được nhiều thành công [60]. Jie Zhou và Wei Xu đã đề xuất một hệ thống end-to-end sử dụng deep BiLSTM [61] để giải quyết vấn đề trên. Trong bài báo này, tác giả sử dụng thông tin hai chiều (bi-directional information) khác với công trình [62], đầu ra của lớp LSTM này sẽ là đầu vào của lớp LSTM tiếp theo với trình tự xử lý ngược lại. Sau đó, cặp LSTM này sẽ được xếp chồng lên để có được mô hình deep LSTM, cấu trúc này gọi là deep bi-directional LSTM network (DB – LSTM) (Hình 3-1). Văn bản gốc chỉ được sử dụng làm đầu vào mà không dùng bất cứ kiến thức ngữ pháp nào.

Mô hình đạt state-of-the-art với F1 là 81.07 và 81.27 lần lượt với bộ ngữ liệu coNLL – 2005 và coNLL – 2012.



Hình 3-: Kiến trúc DB-LSTM [61]

Năm 2018, Sanket Vaibhav Mehta cùng cộng sự đã xây dựng mô hình SRL dưới dạng bài toán gắn thẻ BIO kết hợp với bộ nhớ ngắn – dài hai chiều nhiều lớp (multilayer highway bi-directional long short-term memory) [63]. Khác với [61], phương pháp được đề cập trong bài báo dựa vào việc thực thi các ràng buộc cú pháp (syntactic constraint) bằng cách tăng các mục tiêu đào tạo nhờ vào các thành phần mất tính không nhất quán về cú pháp (syntactic-inconsistency loss component) để đào tạo LSTM có mục tiêu chung (joint-objective LSTM).

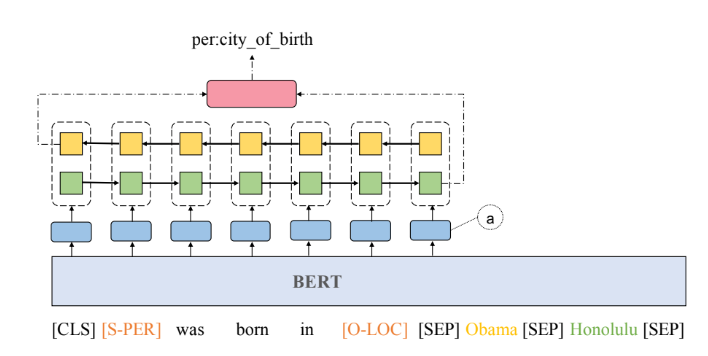
Với mỗi cặp (x,) đã cho trước, dựa vào phương trình được công bố bởi Luheng He cùng cộng sự [64], xét là thẻ tốt nhất có thể tại epoch t trong quá trình huấn luyện mô hình. Trường hợp lý tưởng nhất, nếu mô hình có thể nhận biết cú pháp (syntax-aware) thì ta sẽ có được . Loss component dựa vào syntactic inconsistency (hay SI – loss) được định nghĩa như sau:

Dựa vào phương trình (1), [64] đã định nghĩa một mục tiêu đào tạo chung (joint training objective). Cho trước một cặp câu – vị ngữ (x, ) và thẻ SRL y, joint training objective tại epoch t được định nghĩa:

Đối với ngôn ngữ tiếng anh của bộ ngữ liệu coNLL – 2012, mô hình đề xuất đạt +1.58, +0.78 F1 với mô hình trước đó với ELMo trên cùng một dữ liệu gán nhãn SRL. Ngoài ra, với việc sử dụng SI – loss, mô hình được đề xuất đạt được +3,67, +2,1 F1 so với mô hình được đào tạo trước đó.

Năm 2019, Peng Shi và Jimmy Lin đã đề xuất một mô hình dựa vào BERT cho hai bài toán rút trích mối quan hệ (extraction relation) và gán nhãn ngữ nghĩa [65]. Trong bài báo này, các thử nghiệm trên bộ dữ liệu cho cả hai tác vụ này chỉ ra rằng không cần sử dụng bất kì tính năng nào khác, mô hình BERT này có thể đạt được hiệu suất state-of-the-art. Mô hình này góp phần cung cấp cơ sở vũng chắc cho những nghiên cứu khác trong tương lai.

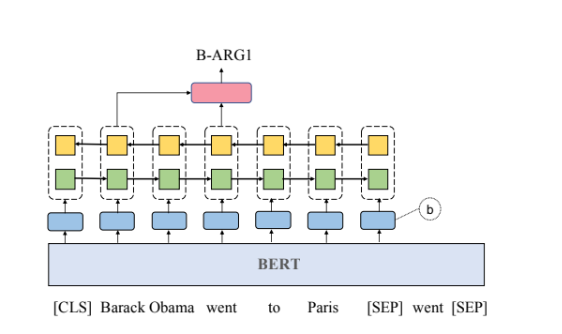
Rút trích mối quan hệ là nhiệm vụ trích xuất mối quan hệ ngữ nghĩa trong văn bản, các mối quan hệ được trích xuất thường xảy ra giữa hai hoặc nhiều thực thể khác nhau. Để mã hóa câu theo cách nhận biết thực thể, bài báo đề xuất một mô hình dựa trên BERT. Chuỗi đầu vào sẽ được mã hóa bằng WordPiece tokenizer [66] và được đưa vào BERT. Kết quả dự đoán dựa trên sự kết hợp của trạng thái ẩn (hidden state) cuối cùng theo mỗi hướng từ BiLSTM, được cung cấp thông qua MLP.



Hình 3-: Kiến trúc mô hình BERT cho tác vụ rút trích mối quan hệ [65]

Kết quả trên tập thử nghiệm TACRED (TAC Relation Extraction Dataset) [67] (Hình 3-2) cho thấy mô hình được đề xuất bởi Peng Shi và Jimmy Lin vượt trội hơn công trình [68] và [69] khi sử dụng đồ thị mạng tích chập (Graph Convolutional Networks – GCNs) và các biến để mã hóa thông tin của cây cú pháp dưới dạng tính năng ngoài (external features). Về F1, hệ thống đạt được điểm số được biết đến nhiều nhất trong số các mô hình riêng lẻ nhưng vẫn thấp hơn mô hình nội suy của [68] vì recall thấp hơn.

Thông thường, nhiệm vụ của bài toán gán nhãn ngữ nghĩa bao gồm: phát hiện vị từ (predicate detection), định hướng ý nghĩa của vị từ (predicate sense disambiguation), xác định đối số (argument identification) và phân loại đối số (argument classification). Có hai loại biểu diễn cho đối số: span-based (sử dụng trong PropBank) và dependency – base (được sử dụng trong coNLL 2008 và 2009). Trong bài báo này, đối với mô hình BERT cho tác vụ SRL, tác giả áp dụng công trình [70] để kết hợp cả hai loại biểu diễn mà không cần bất kỳ ràng buộc khai báo nào để giải mã. Ngoài ra, tác giả cũng chỉ thảo luận về định hướng vị từ và xác định và phân loại đối số.



Hình 3-: Kiến trúc mô hình BERT cho tác vụ SRL [65]

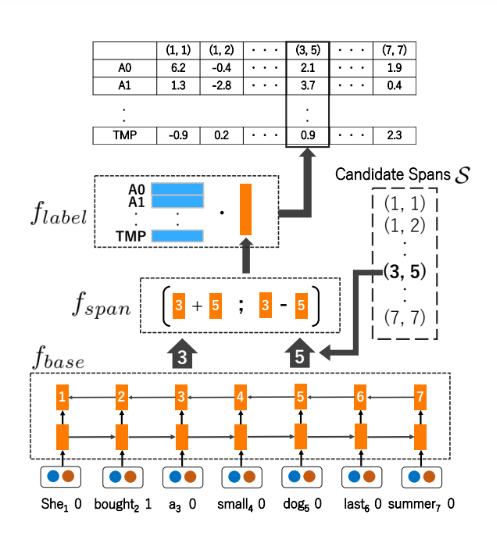
Với nhiệm vụ định hướng cho vị ngữ, dù không sử dụng bất kì đặc trưng ngôn ngữ và ràng buộc khai báo nào, mô hình vẫn cho ra kết quả vượt trội hơn so với mô hình state-of-the-art tại thời điểm đó. Còn đối với nhiệm vụ phân loại đối số, cũng cho ra kết quả vượt trội hơn so với các mô hình riêng lẻ và với công trình [71] trên bộ ngữ liệu coNLL – 2005. Tuy nhiên với bộ ngữ liệu coNLL – 2012 chưa thể vượt qua [71] do công trình có độ chính xác rất cao bởi họ có lớp giải mã phức tạp hơn với các ràng buộc được thiết kế bởi con người như “Overlap Constraint” và “Number Constraint”

1. **SPAN**

Đối với bài toán Gán nhãn vai trò ngữ nghĩa, với đầu vào là một câu và vị ngữ, hệ thống SRL phải dự đoán đối số ngữ nghĩa của vị ngữ đó. Mỗi đối số chính là một khoảng (span) – một đơn vị bao gồm một hoặc nhiều từ. Để có thể dự đoán các khoảng đối số hiệu quả chính là dựa vào cách biểu diễn và mô hình hóa các khoảng.

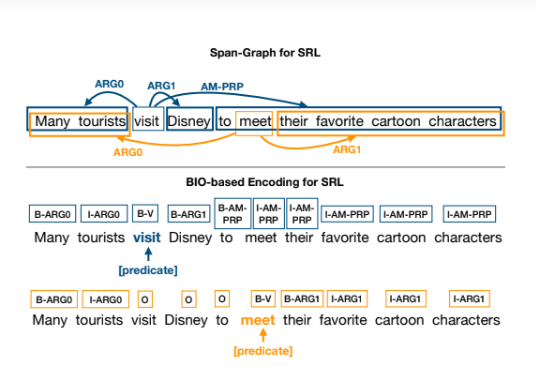
Cách tiếp cận phổ biến nhất chính là dựa vào BIO – tagging, những từ bắt đầu và bên trong khoảng đối số sẽ được gắn nhãn “B” và “I”, và những từ nằm ngoài khoảng đối số sẽ được gắn thẻ “O”. Mặc dù mang lại độ chính xác cao, nhưng thay vì dự đoán trực tiếp các khoảng, cách tiếp cận này lại tái tạo lại các khoảng đối số từ thẻ BIO. Một cách tiếp cận khác chính là dựa vào sự dự đoán cho khoảng được gán nhãn (labeled span prediction). Cách tiếp cận này đánh giá khoảng dựa vào nhãn của nó. Điểm mạnh của phương pháp này là nó cho phép thiết kế mô hình ở cập độ khoảng đối số, điều mà BIO khó có thể thực hiện. Mặt khác, cách tiếp cận này lại cho ra kết quả kém hơn so với BIO.

Để khắc phục nhược điểm này, năm 2018, [72] đã đề xuất mô hình dựa trên khoảng đối số đơn giản và chính xác. Mô hình được đề xuất xem bài toán gán nhãn ngữ nghĩa như một tác vụ lựa chọn khoảng đối số. Điều này nghĩa là mô hình sẽ chấm điểm trực tiếp trên tất cả khoảng có thể được gắn nhãn dựa vào các biểu diễn khoảng đối số được tạo ra từ mạng thần kinh (neuron network). Tại thời điểm giải mã, mô hình lựa chọn các khoảng được gán nhãn có điểm số cao hơn. Các tham số mô hình được học bởi việc tối ưu hóa log-likehood của những khoảng được gán nhãn chính xác.

Hình 3-: Tổng quan kiến trúc của mô hình BiLSTM-span [72]

Mô hình được đánh giá trên bộ ngữ liệu coNLL-2005 và 2012. Kết quả thực nghiệm cho thấy mô hình được đề xuất vượt trội hơn so với mô hinhg BiLSTM-CRF. Ngoài ra, nhờ việc sử dụng các biểu diễn ngữ cảnh (contextualized word representations), mô hình đã đạt state-of-the-art với F1 là 87.4 và 87.0 lần lượt trên coNLL-2015 và 2012.

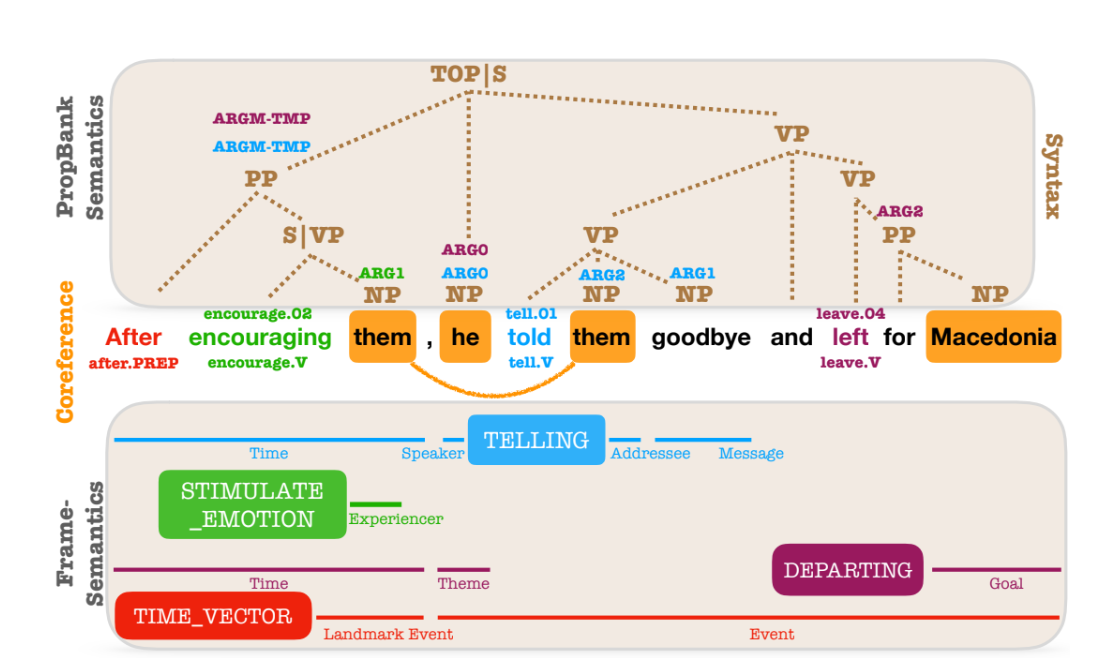
Trong khi các mô hình trước đó đánh giá dựa vào các vị ngữ được đưa vào mô hình. Lần đầu tiên, Luheng He cùng cộng sự đã đề xuất một mô hình SRL dựa trên khoảng mà không cần giả định rằng các vị ngữ đã được cung cấp sẵn. [73] đề xuất cách tiếp cận end-to-end để dự đoán tất cả các vị từ và khoảng đối số của chúng. Mô hình sử dụng các biểu diễn khoảng đối số theo ngữ cảnh (contextualized span representations) để dự đoán biểu đồ SRL (SRL graph) trên các đoạn văn bản. Biểu đồ cuối cùng (Hình 3-5) là sự kết hợp của các vai trò ngữ nghĩa được dự đoán (các cạnh) và các khoảng văn bản (các nút) được liên kết của chúng.

****

Hình 3-: So sánh giữa biểu đồ khoảng đối số và BIO-base SRL [73]

Mô hình đạt state-of-the-art trên bộ ngữ liệu PropBank. Nó cũng góp phần khẳng định hiệu suất mạnh mẽ của các mô hình mã hóa khoảng đối số (span embedding) tương tự, cho thấy rằng hướng tiếp cận này có thể sử dụng cho các nhiệm vụ liên quan đến khoảng đối số khác, chẳng hạn như phân tích cú pháp (syntatic parsing), trích xuất quan hệ (relation extraction), …

Tri thức cú pháp, tuy mang lại lợi ích cho bài toán SRL, nhưng việc phân tích cú pháp lại tốn kém nhiều chi phí, liệu có cách nào để bộ phân tích ngữ nghĩa thu được lợi ích từ cú pháp mà không phải tốn chi phí cho quá trình trên? Để trả lời cho câu hỏi này, Swabha Swayamdipta cùng cộng sự đã đề xuất mô hình học đa tác vụ [74] để kết hợp thông tin cú pháp vào các mô hình ngữ nghĩa thần kinh (neural semantics models). Mô hình được đề xuất, giàn giáo cú pháp (syntactic scafold) sử dụng thông tin cú pháp như một tác vụ phụ trợ cho tác vụ chính SRL. Vì bản thân tác vụ giàn giáo không phải là mục đích cuối cùng của mô hình, tác giả đã nới lỏng vấn đề phân tích cú pháp thành một tập hợp các dự đoán cấp độ khoảng đối số độc lập (independent span-level predictions), không có ràng buộc rằng chúng sẽ tạo thành một cây phân tích cú pháp hợp lệ. Điều này có nghĩa là mô hình không cần phải chạy thuật toán phân tích cú pháp.

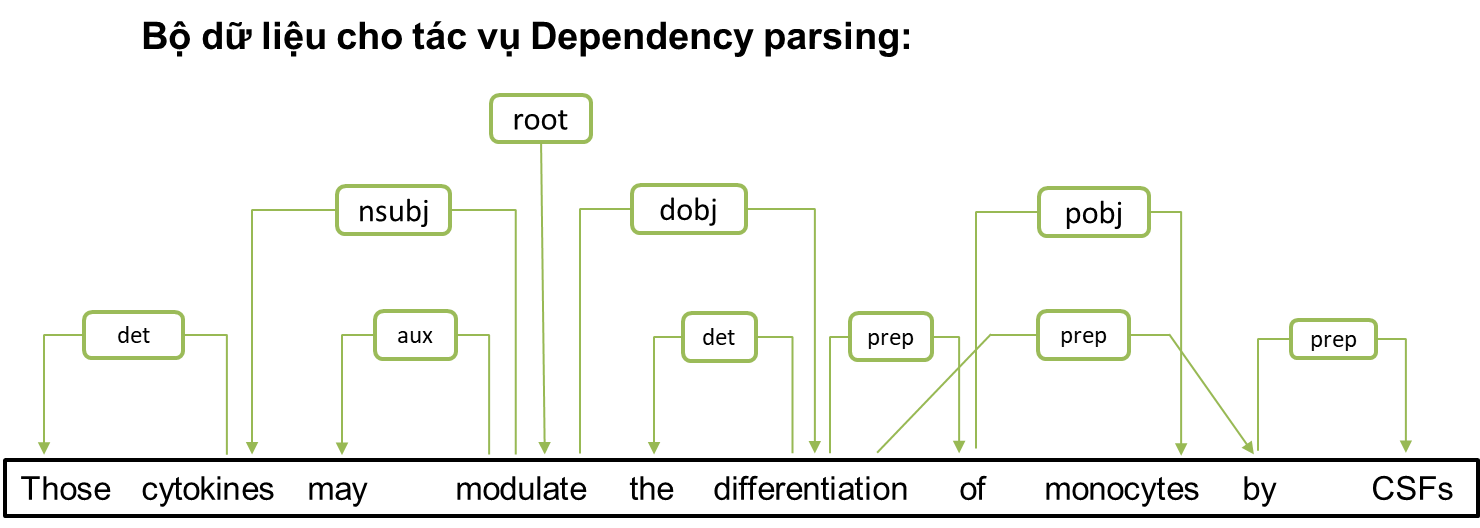


Hình 3-: Ví dụ cho việc phân tích cú pháp của một câu [74]

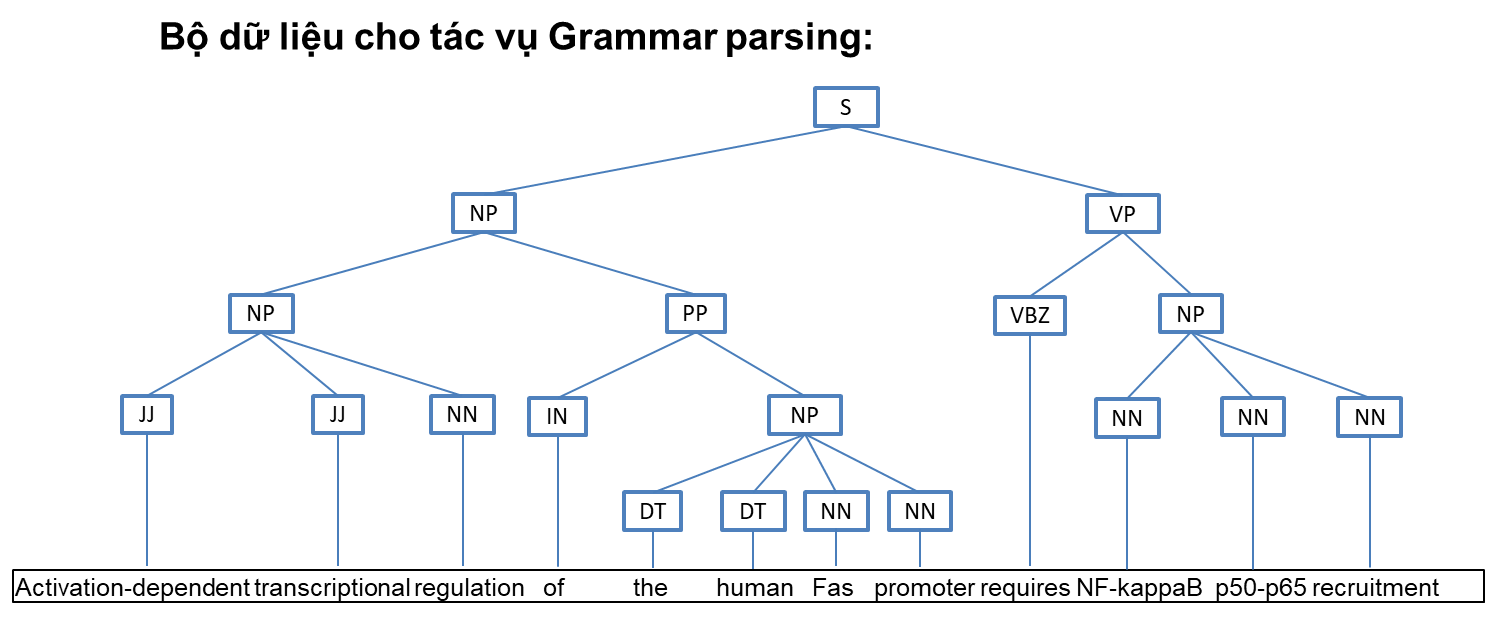
Thực nghiệm cho thấy giàn giáo cú pháp mang lại sự thúc đầy đáng kể cho hai bài toán SRL và phân giải đồng cú pháp (coreference resolution) state-of-the-art hiện tại. Mô hình sử dụng kiến trúc mạng thần kinh mạnh nhất thời điểm đó cho những tác vụ này, tích hợp học sâu cũng như dự đoán có cấu trúc ở cấp độ khoảng đối số. Mô hình giàn giáo cú pháp vượt trội hơn so với mô hình trước đó với F1 tăng 3.6 trên bộ ngữ liệu FrameNet SRL; 1.1 trên bộ ngữ liệu PropBank SRL; và 0.6 cho tác vụ phân giải đồng cú pháp.

1. **Hướng tiếp cận nhận biết cú pháp**

Thuật ngữ phân tích cú pháp (syntax parsing) đề cập đến quả trình xây dựng cây phân tích cú pháp (parsing tree) từ một câu đã cho trước. Cây cú pháp có vai trò làm nổi bật cấu trúc cú pháp của một câu theo một vài dạng ngữ pháp chính thức (formal grammar). Tùy thuộc vào dạng ngữ pháp được sử dụng, cây cú pháp sẽ có các tính năng cũng như cấu trúc khác nhau. Mặc dù có cùng mục tiêu là trích xuất thông tin cú pháp, nhưng phân tích ngữ pháp (grammar parsing, hay constituent parsing) và phân tích sự phụ thuộc (dependency parsing) sử dụng hai dạng ngữ pháp khác nhau; vì thế, cây phân tích cú pháp của hai tác vụ này cũng sẽ khác nhau.



Hình 3-: Ví dụ về dependency parsing tree

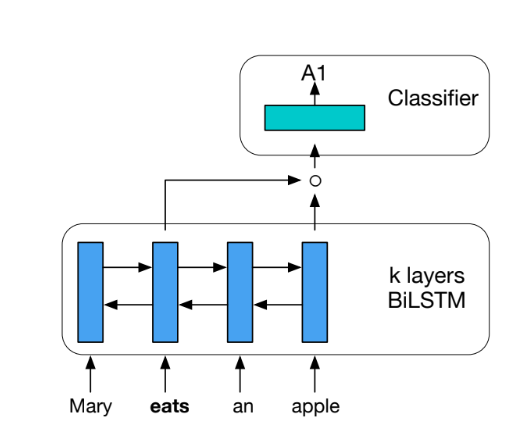


Hình 3-: Ví dụ về grammar parsing tree

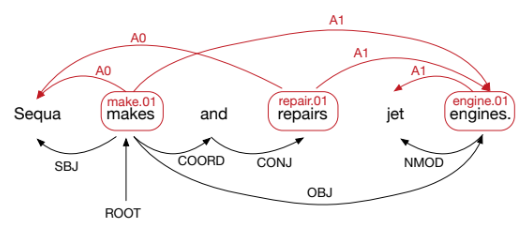
1. **Phân tích sự phụ thuộc**

Phân tích phụ thuộc cú pháp là một kỹ thuật trong xử lý ngôn ngữ tự nhiên, được sử dụng để phân tích cấu trúc ngữ pháp của câu. Quá trình này liên quan đến việc xác định chủ ngữ, tân ngữ, cũng như các yếu tố ngữ pháp khác trong một câu, sau đó xác định mối quan hệ của các thành phần này trong câu. Trình phân tích cú pháp sẽ sử dụng tập các quy tắc ngữ pháp và mô hình ngữ pháp để phân tích và xây dựng cây phụ thuộc hoặc biểu đồ thể hiện mối quan hệ giữa các từ trong câu (Hình 3-7).

Năm 2017, lần đầu tiên, Diego Marcheggiani cùng cộng sự đã đề xuất mô hình bất khả tri về ngữ pháp (syntax – agnostic) [75] hiệu quả cho bài toán gắn nhãn ngữ nghĩa dựa trên sự phụ thuộc (dependency – based). Ban đầu, mô hình dựa đoán các phụ thuộc đối số vị ngữ dựa trên trạng thái của BiLSTM encoder, và đạt được kết quả cạnh tranh ngay cả khi không có bất kì thông tin ngữ pháp nào mà chỉ sử dụng các suy luận cục bộ. Tuy nhiên, khi các thẻ gán nhãn từ loại (part-of-speech tag) được dự đoán tự động được cung cấp làm đầu vào, mô hình được đề xuất đã cho ra kết quả vượt trội hơn so với các mô hình trước đây trên bộ ngữ liệu coNLL 2009.

Để xác định và phân loại các đối số, [75] đề xuất mô hình bao gồm ba thành phần (Hình 3-5): thành phần biểu diễn từ (word representation), bộ mã hóa BiLSTM và cuối cùng là thành phần phân loại (classifier).

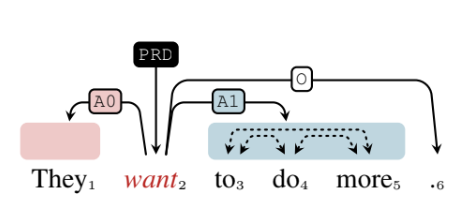
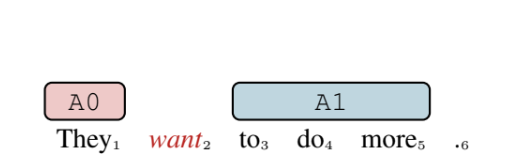
Hình 3-: Dự đoán đối số và nhãn của nó dựa vào bộ mã hóa LSTM [75]

Cũng trong năm 2017, Diego Marcheggiani đã đề xuất cải tiến mô hình LSTM state-of-the-art hiện tại cho bài toán SRL bằng cách sử dụng một phiên bản của mạng tích chập đồ thị (graph convolutional networks – GCNs) kết hợp với LSTM [76]. Ngoài ra, nhận thấy thông tin về cú pháp cũng có liên quan chặt chẽ tới cấu trúc ngữ nghĩa, tác giả cũng đã khai thác thông tin về cú pháp trong mô hình của mình. Đây là công trình đầu tiên chứng minh rằng GCNs có mang lại hiệu quả cho bài toán xử lý ngôn ngữ tự nhiên, cũng như chứng minh một cách tổng quát rằng GCN phù hợp để mã hóa thông tin cú pháp ở cấp độ từ. Mô hình thu được kết quả state-of-the-art trên cả tiếng Anh và tiếng Trung với bộ ngữ liệu coNLL 2009.

Hình 3-: Tương quan giữa phụ thuộc ngữ nghĩa (trên) và cú pháp (dưới) [76]

Năm 2018, các nghiên cứu trước đó đã chứng minh thông tin cú pháp có đóng góp trong hiệu suất của bài toán SRL. Nhận định này đã gặp phải thách thức khi các công trình mới nhất đã cho ra kết quả ấn tượng ngay cả khi không cần thông tin về cú pháp [77]. Shexia He cùng cộng sự đã đề xuất mô hình giúp khẳng định lại tầm quan trọng của cú pháp đối với bài toán SRL dựa trên phụ thuộc. Để đánh giá định lượng về sự đóng góp của cú pháp vào bài toán SRL, công trình sử dụng tỷ lệ giữa điểm F1 cho các phụ thuộc ngữ nghĩa đã được gán nhãn (Sem-F1) và điểm gắn nhãn (labeled attachment score) cho các phụ thuộc cú pháp làm chỉ số đánh giá. Tỷ lệ này mang lại sự đánh giá công bằng hơn dựa các hệ thống SRL dựa trên cú pháp khi các trình phân tích cú pháp khác nhau đóng góp các đầu vào cú pháp khác nhau với nhiều mức chất lượng khác nhau. Mô hình đạt state-of-the-art với bộ ngữ liệu coNLL trên cả tiếng Anh và tiếng Trung, cho thấy tầm quan trọng của cú pháp đối với bài toán SRL.

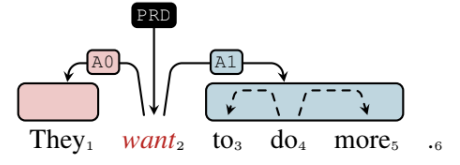
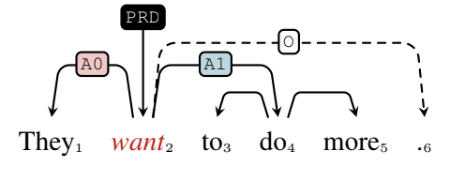
Mặc dù các hướng tiếp cận BIO – based và SPAN – based của các công trình SRL vào năm 2021 mang lại hiệu quả đáng kể, nhưng chúng có cùng một nhược điểm là không xem xét rõ ràng cấu trúc đối số bên trong. Để khắc phục vấn đề này, [78] đã đề xuất đưa SRL thành tác vụ phân tích sự phụ thuộc cú pháp và coi các khoảng đối số phẳng (flat argument spans) là các cây con tiềm ẩn (latent subtree).



(1): cấu trúc gốc

(2): quá trình huấn luyện

Hình 3-: Minh họa quá trình chuyển đổi SRL 🡪 Tree và quá trình nghịch đảo của nó [78]



(3): giải mã

(4): khôi phục

Hình 3-: Minh họa quá trình chuyển đổi SRL 🡪 Tree và quá trình nghịch đảo của nó [78]

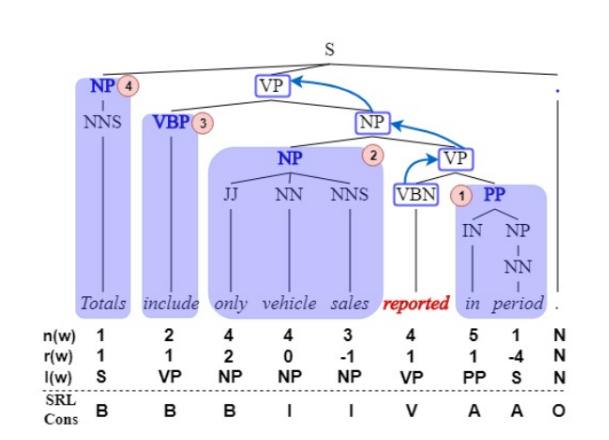
Cụ thể, công trình sử dụng TreeCRF [79] [80] để tìm hiểu về cây nhận diện từng phần (partially observed tree) và loại bỏ cấu trúc tiềm ẩn (latent structure) trong quá trình huấn luyện. Vì TreeCRF thông thường sẽ liệt kê tất cả các cây có thể có, nên [78] cần phải áp đặt ràng buộc khoảng (span constraint) lên các subtree nhằm phản ánh chính xác ranh giới của các đối số. Để giải quyết vấn đề này, công trình đã thiết kế một ràng buộc khoảng mới dành cho TreeCRF để điều chỉnh nó theo quy trình huấn luyện của mô hình. Các thử nghiệm trên bộ ngữ liệu coNLL05 và coNLL12 đã cho thấy mô hình được đề xuất đã vượt trội hơn so với các phương pháp trước đó và đạt được state-of-the-art.

1. **Grammar**

Khác với Phân tích sự phú thuộc cú pháp, Phân tích ngữ pháp (hay Phân tích cấu trúc thành phần) liên quan đến việc xác định cấu trúc cú pháp của câu thông qua việc phân tích các từ và cụm từ (cụm danh từ, cụm động từ, …) của nó. Đầu ra của trình phần tích cú pháp thường là một cây phân tích cú pháp đại diện cho cấu trúc phân cấp của câu (Hình 3-8).

Có nhiều cách khác nhau để đưa tri thức ngữ pháp vào bài toán SRL, năm 2019 Yufei Wang cùng cộng sự đã đề xuất phương pháp hiệu quả nhất cho công việc này [81]. Công trình đánh giá ba phương pháp khác nhau để mã hóa thông tin cú pháp cũng như ba hướng khác nhau để đưa chúng vào mô hình SRL dựa trên ELMo state-of-the-art hiện tại. Cụ thể, [81] đã đề xuất ba phương pháp mã hóa cú pháp:

* Cây cú pháp thành phần (full constituency tree representation – Full-C) (Hình 3-7): điểu chỉnh biểu diễn phân tích cú pháp (constituency parsing representation) từ công trình [82] và mã hóa cấu trúc cây thành một tập hợp các đặc trưng cho các cặp từ.
* Khoảng SRL cụ thể (SRL specific span – SRL-C) (Hình 3-7): sử dụng các biểu diễn khoảng phân loại (representation of the constituency spans) phù hợp nhất với các nhiệm vụ của SRL dựa vào công trình [83]
* Cây phụ thuộc (dependency tree – DEP): đề xuất biểu diễn vector rời rạc (discrete vector representation) mã hóa các mối quan hệ danh từ chính – từ bổ nghĩa (head-modifier relationship) trong các cây phụ thuộc



Hình 3-: Ví dụ về Full-C và SRL-C [81]

Kết quả thực nghiệm cho thấy việc sử dụng biểu diễn thành phần làm đầu vào sẽ giúp cải thiện hiệu suất nhiều nhất, mô hình cũng đạt được state-of-the-art cho tác vụ SRL trên bộ ngữ liệu coNLL’05 và coNLL’12.

Năm 2020, có nhiều nghiên cứu trong lĩnh vực SRL thường nhúng tri thức ngữ pháp dựa vào các biểu diễn sự phụ thuộc cú pháp hơn là sử dụng cây cú pháp thành phần. Diego cùng cộng sự, đã đề xuất mô hình dựa trên mạng tích chập đồ thị (graph convolutional networks) gọi là SpanGCN [84]. Nhóm tác giả đã chỉ ra rằng, so với việc sử dụng cây phụ thuộc cú pháp, khai thác cây cấu trúc thành phần có phần tự nhiên hơn và có thể mang lại nhiều tính năng dự đoán hơn cho các vai trò ngữ nghĩa. [84] cho thấy cách GCN có thể được áp dụng để mã hóa cấu trúc thành phần cú pháp và cung cấp nó cho hệ thống SRL. Việc tính toán được thực hiện trong 3 giai đoạn. Đầu tiên, các biểu diễn nút ban đầu được tạo ra bằng cách kết hợp các biểu diễn của từ đầu tiên và từ cuối cùng trong một thành phần của câu. Sau đó, các tích chập đồ thị (graph convolution) được thực hiện dựa trên cây cú pháp thành phần, mang lại biểu diễn constituent được cung cấp thông tin cú pháp. Cuối cùng, các biểu diễn constituent được phân tách trở lại thành các biểu diễn từ, được sử dụng làm đầu vào cho bộ phân loại SRL. Thực nghiệm cho thấy mô hình được đề xuất cho ra kết quả khá cao trên bộ dữ liệu coNLL 2005 và coNLL 2012.

# Chương 4: PHƯƠNG PHÁP THỰC HIỆN

1. **Tổng quan phương pháp thực hiện**

Khóa luận xây dựng giải pháp cho bài toán rút trích Cấu trúc đối số vị ngữ trong lĩnh vực Y sinh dựa vào giải pháp được đề xuất từ:

* Công trình: “Deep Semantic Role Labeling with Self-Attention” [85] của Zhixing Tan và các cộng sự được viết vào năm 2018
* Công trình: “Multi-Task Active Learning for Neural Semantic Role Labeling on Low Resource Conversational Corpus” [86] của Fariz Ikhwantri và cộng sự được viết vào năm 2018.

Bài báo của Zhixing Tan đã trình bày một mô hình học sâu đơn giản và hiệu quả cho tác vụ SRL góp phần giải quyết thách thức của mạng thần kinh hồi quy (recurrent neural networks) trong việc xử lý thông tin cấu trúc và các phụ thuộc tầm xa (long range dependencies). Mô hình đạt được F1 = 82.7 trên bộ ngữ liệu coNLL 2015 và F1 = 83.4 trên bộ ngữ liệu coNLL 2005 và vượt trội hơn mô hình state-of-the-art trước đó với F1 lần lượt là 1.0 và 1.8.

Về bài báo của Fariz Ikhwantri, tác giả đã đề xuất mô hình học đa tác vụ (multi-task learning) giữa gán nhãn ngữ nghĩa là tác vụ chính và nhận dạng thực thể là tác vụ phụ để sử dụng thông tin từ tác vụ này phụ trợ cho tác vụ chính SRL. Mô hình được đánh giá dựa vào bộ gữ liệu có kích thước hạn chế là dữ liệu đàm thoại của Indonesia. Nhờ vào việc mô hình tiếp cận với bộ ngữ liệu có kích thước thấp, cũng như cho ra kết quả vượt trội hơn so với học tích cực (active learning) cho một tác vụ, Khóa luận quyết định xem xét và áp dụng mô hình này cho bài toán Gán nhãn ngữ nghĩa cho văn bản trong lĩnh vực Y sinh.

Các bước thực hiện đề đưa ra mô hình đề xuất cho Khóa luận:

* Bước 1: Đánh giá mô hình học sâu được đề xuất bởi Zhixing Tan [85].
* Bước 2: Đánh giá mô hình đa tác vụ được đề xuất bởi Fariz Ikhwantri [86].
* Bước 3: Đề xuất mô hình kết hợp và cái tiến giữa hai mô hình trên.

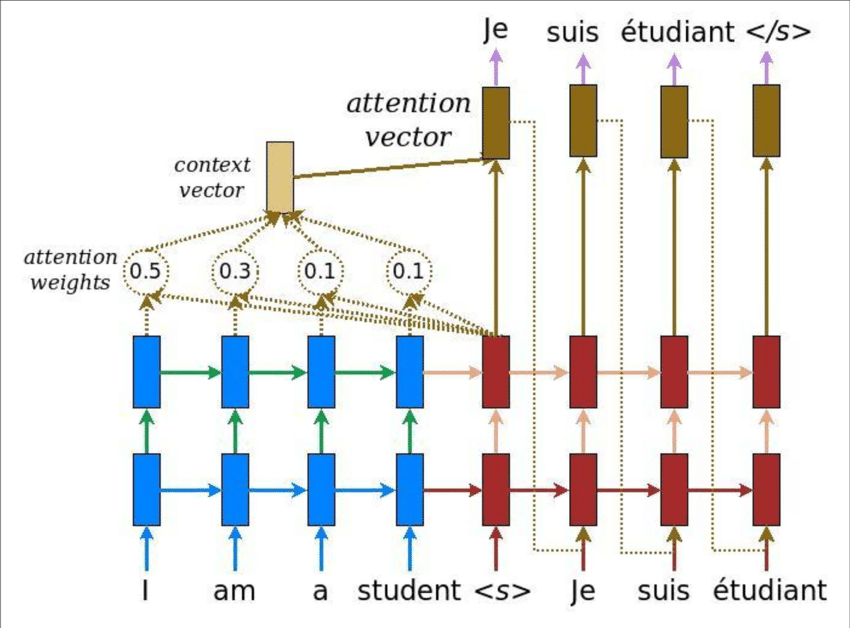
1. **Đánh giá mô hình học sâu của Zhixing Tan**
2. **Giới thiệu tổng quan về mô hình được đề xuất bởi Zhixing Tan**

Năm 2017, các hệ thống end-to-end cho bài toán SRL với việc sử dụng mạng thần kinh hồi quy (recurrent neural networks – RNN) ngày càng được chú ý bởi các công trình nghiên cứu. Tuy nhiên, việc xử lý các thông tin cấu trúc (structural information) cũng như các phụ thuộc tầm xa (long range dependencies) vẫn là một thách thức lớn đối với RNN. Để giải quyết vấn đề này, Zhixing Tan cùng cộng sự đã đề xuất mô hình deep attentional neural network (DEEPATT) [85] cho tác vụ SRL. Mô hình này dựa trên cơ chế self-attention, cơ chế này sẽ tự động rút trích sự phụ thuộc giữa các từ khác nhau trong một câu của dữ liệu đầu vào. Khác với RNN, self-attention thực hiện kết nối trực tiếp hai token bất kì trong một câu. Ngoài ra, self-attention cũng cung cấp một cách linh hoạt hơn để lựa chọn, trình bày và tổng hợp thông tin của các câu đầu vào và bổ sung cho các mô hình dựa trên RNN.

1. **Cơ sở lý thuyết**
2. **Giới thiệu về cơ chế Attention**

Trong thực tế, bộ não con người cũng có cơ chế attention của riêng nó. Ví dụ, khi đọc một câu văn hay đoạn văn, tại thời điểm nào đó chúng ta chỉ tập trung vào các từ khóa của câu, hay câu mang ý chính của đoạn văn rồi từ đó mới đưa ra thông tin cần thiết. Chính cơ chế này giúp bộ não của chúng ta tiết kiệm năng lượng vì bỏ qua thành phần không đem lại nhiều giá trị, giúp đưa ra kết quả nhanh hơn cùng với độ chính xác cao.

Dựa trên ý tưởng đó, cơ chế Attention được ra đời để giải quyết những vấn đề của mô hình seq2seq gặp phải, với một ý tưởng là tạo ra một vector bối cảnh có thể tương tác với toàn bộ vector trạng thái ẩn của encoder thay vì chỉ sử dụng vector trạng thái ẩn cuối cùng để tạo ra biểu diễn cho decoder. Cụ thể khi áp dụng attention cho mô hình seq2seq, attention sẽ có kiến trúc như sau:



Hình 4-: Mô hình self-attention [87]

Các khối màu xanh dương là đại diện cho thành phần encoder, khối màu đỏ là thành phần decoder.

Tại mỗi time-step ở phía decoder:

* Đầu tiên, nhận vector trạng thái ẩn của decoder ht và tất cả các vector trạng thái ẩn của decoder hs.
* Tiếp theo, tính điểm attention. Với mỗi vector trạng thái ẩn của encoder thì ta cần tính điểm thể hiện sự liên quan tới vector trạng thái ẩn ht của decoder. Cụ thể, áp dụng một phương trình tính điểm với đầu vào là vector trạng thái ẩn decoder ht và vector trạng thái ẩn của encoder hs, hàm này sẽ trả về một giá trị vô hướng, tạm gọi là score (ht, hs).
* Kế đến, tính trọng số attention. Áp dụng hàm softmax với đầu vào là điểm attention.
* Tiếp đến, tính toán vector bối cảnh ct là tổng của các trọng số attention nhân với vectơ trạng thái ẩn của decoder tại time step tương ứng
* Cuối cùng, các vector attention at dùng để đưa đầu ra được tính toán dựa trên vector bối cảnh ct và vector trạng thái ẩn ở decoder ht

Các loại cơ chế Attention: Tùy theo cách tính hàm score mà có những loại attention khác nhau, có thể kể đến như bảng dưới đây:

|  |  |
| --- | --- |
| **Tên cơ chế Attention** | **Hàm tính score** |
| Content-base attention |  |
| Additive |  |
| Dot-Product |  |
| General |  |

Bảng 4-: Các loại cơ chế attention

Xét rộng hơn, có thể chia cơ chế attention thành ba loại như sau:

* Self-attention
* Global/Soft attention
* Local/Hard attention

Tuy nhiên trong phạm vi khóa luận chỉ nói đến Self-Attention. Self-Attention hay còn được gọi là Intra Attention là cơ chế chỉ dùng cho một câu. Ý tưởng của cơ chế này là tạo ra một ma trận với hàng và cột đều cùng một câu, từ đó giúp hiểu được thành phần nào sẽ có liên hệ với thành phần nào (từ nào trong câu sẽ có mối quan hệ với từ nào)

Điểm mạnh của cơ chế này là không bị giới hạn trong việc tìm kiếm sự tương quan giữa hai thành phần. Self-Attention cho phép mô hình quan sát câu đang đề cập một cách trọn vẹn nhất và nối các từ liên quan tới ngữ cảnh của chúng. Bằng cách này, Self-Attention có thể sử dụng thông tin tạo bởi mối quan hệ giữa các từ bất kể chúng nằm cách xa bao nhiêu, điều vốn dĩ không thể làm được ở các mô hình truyền thống như RNN, LSTM.

1. **Mô hình được đề xuất**

Diagram

Description automatically generated with medium confidenceTrong bài báo này, tác giá xem SRL như là một tác vụ gắn thẻ BIO (BIO tagging). Đầu tiên, các câu đầu vào cùng vị ngữ của chúng sẽ được mã hóa thành các vector giá trị thực (real value vector). Sau đó, deep attentional neural network sẽ nhận vào các vector đã được mã hóa làm đầu vào để nắm bắt các cấu trúc lồng nhau (nested structures) của câu và các mối quan hệ phụ thuộc tiềm ẩn (latent dependency relationships) giữa các nhãn. Ở giai đoạn suy luận, chỉ kết quả thu được ở tầng trên cùng của attention-sublayer mới được đưa đến lớp hồi quy logistic (logistic regression) để đưa ra quyết định cuối cùng.

Hình 4-:Minh họa mô hình deep attentional neural network [85]

Mô hình đề xuất đã vượt qua mô hình state-of-the-art tại thời điểm đó trên bộ ngữ liệu coNLL-2015 và 2012 với F1 tăng lần lượt là 1.8 và 1.0. Ngoài ra, mô hình cũng đạt được kết quả cải thiện hơn mô hình end-to-end trước đó [64] với F1 tăng 2.0. DEEPATT cho ra tốc độ phân tích cú pháp 50 nghìn tokens mỗi giây trên một GPU Titan X

1. **Đánh giá mô hình học đa tác vụ chủ động do F.Ikhwantri đề xuất:**
2. **Lý do lựa chọn mô hình**

Theo phương pháp tiếp cận truyền thống, các phương pháp học sâu thường chỉ tập trung giải quyết một nhiệm vụ duy nhất mà không quan tâm tới những nhiệm vụ khác có liên quan tới nhiệm vụ mà chúng ta đang quan tâm. Điều này có thể khiến chúng ta bỏ qua một số thông tin hữu ích từ những nhiệm vụ phụ đó mà có thể giúp chúng ta thực hiện tốt hơn nhiệm vụ chính của mình.

Ví dụ, dưới góc độ học tập huấn luyện của con người, để có được một kỹ năng tốt trong một lĩnh vực vụ thể nào, lúc này không những đòi hỏi người học phải có kiến thức, kỹ năng trong lĩnh vực đó mà cần phải có kiến thức kỹ năng của các lĩnh vực liên quan để bổ trợ cho kỹ năng chính.

Hay một vận động viên bóng rổ chỉ có kỹ năng qua người, ném rổ tốt mà thiếu đi những nhân tố bổ trợ như thể lực, tâm lý thi đấu thì không thể trở thành một vận động viên xuất sắc được.

Với ý tưởng đó, năm 2017 Mô hình học chủ động đa tác vụ (Multi-Task Active Learning) [86] của nhóm tác giả người Indonesia ra đời, đứng đầu trong đó là F.Ikhwantri. Mô hình giới thiệu phương pháp xây dựng một mô hình Deep Learning cho nguồn tài liệu ngữ liệu hạn chế: gồm 6057 câu là bộ ngữ liệu trò chuyện của người Indonesia.

1. **Cơ sở lý thuyết**
2. **Học đa tác vụ (multi-task learning)**

Quay lại với góc độ học sâu, từ những ý tưởng “những tác vụ, huấn luyện liên quan đến tác vụ chính sẽ bổ sung, hỗ trợ cho tác vụ chính có hiệu quả tốt hơn” đã hình thành nên học đa tác vụ.

Học đa tác vụ (Multi-Task Learning) là thuật toán cho phép huấn luận các tác vụ trên một mô hình hay nhiều mô hình cùng nhau để hỗ trợ lẫn nhau. Từ đó giúp cho nhiệm vụ chính sẽ được cải thiện đáng kể do tận dụng thông tin kiến thức từ những tác vụ phụ.

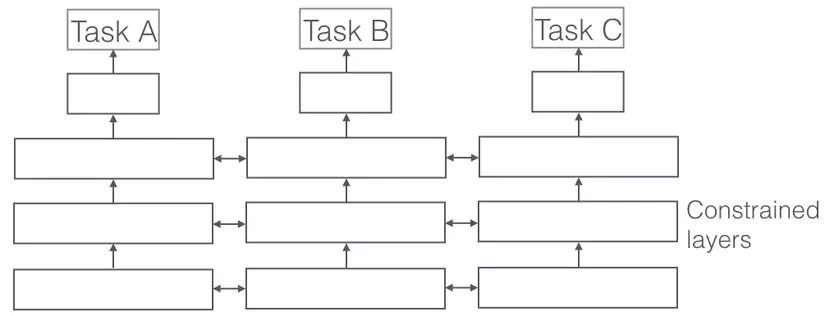
Học đa tác vụ (Multi - Task Learning) hay còn được gọi với những cái tên khác như: Học kết hợp (joint-learning), Học để học (learning to learn), học cùng tác vụ phụ (learning with auxiliary task). Hiện nay, học đa tác vụ đã được biết đến rộng rãi và đạt được những kết quả tích cực, có thể kể đến như: xử lý ngôn ngữ tự nhiên, thị giác máy tính và nhận dạng giọng nói, …

Tuy nhiên, không phải bất cứ tác vụ nào cũng có thể huấn luyện cùng nhau. Nếu như tác vụ phụ và tác vụ không liên quan (không có mối quan hệ với nhau) được huấn luyện cùng nhau, tác vụ phụ sẽ làm ảnh hưởng xấu đến kết quả của tác vụ chính, không cải thiện mà còn giảm sút hơn so với việc huấn luyện đơn tác vụ. Do đó, việc chọn tác vụ phụ để huấn luyện cùng tác vụ chính cần phải hết sức cẩn thận: chọn tác vụ phụ sao cho có sự tương đồng, hoặc bổ trợ với tác vụ chính.  Ví dụ trong lĩnh vực NLP, Intent Classification và Named Entity Recognition, Part of Speech và Semantic Role Labeling, … thường được huấn luyện cùng nhau.

Học đa tác vụ có nhiều cách sử dụng, tuy nhiên trong phạm vi khóa luận chỉ giới thiệu hai phương pháp là: Soft Parameter Sharing và Hard Parameter Sharing

1. **Soft Parameter Sharing**

Đối với Soft Parameter Sharing, mỗi tác vụ sẽ có một mô hình riêng, cũng như tham số riêng của nó. Tuy nhiên để đảm bảo mối quan hệ giữa các tác vụ, khoảng cách tham số giữa các tác vụ sẽ được ràng buộc lẫn nhau để khiến các tham số này có mức độ tương đồng.

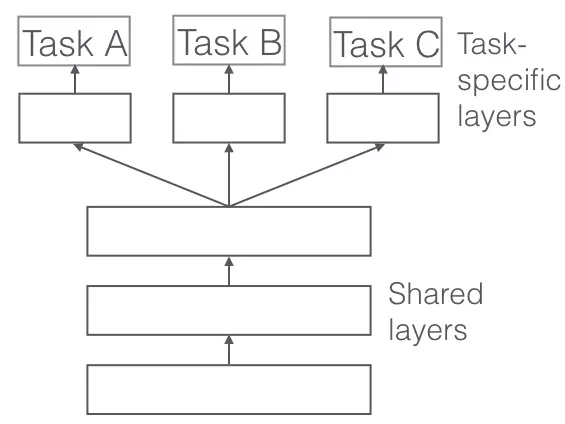


Hình 4-: Kiến trúc. Soft Parameter Sharing [88]

Và cách tiếp cận để áp đặt sự tương đồng của các tham số trong các task này tương tự như phương pháp Hồi quy Ridge: trong đó Ridge ràng buộc tham số còn với Soft Parameter Sharing thì ràng buộc khoảng cách.

1. **Hard Parameter Sharing**

Đây là phương pháp phổ biến được sử dụng trong mạng neural. Ý tưởng của nó là chia sẻ tất cả các lớp hidden của tất cả tác vụ, và output layer là khác nhau cho các tác vụ.

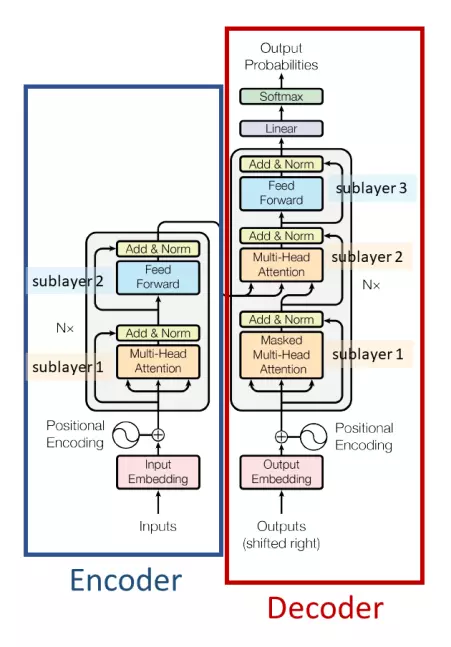


Hình 4-: Mô hình Hard Parameter Sharing [88]

Do chia sẻ chung các lớp Hidden layer nên bắt buộc mô hình phải học những biểu diễn tổng quát ở tất cả các tác vụ, không ưu tiên học tác vụ này là chính hay là phụ, do đó đây là một phương pháp giảm overfitting vào tác vụ chính rất tốt.

1. **Đề xuất mô hình “Rút trích đối số vị ngữ trong văn bản Y sinh”**
2. **Cơ sở lý thuyết**
3. **Tranformer**

Tương tự như RNN, Transformer cũng sử dụng 2 thành phần chính là Encoder và Decoder.  Tuy nhiên đối với RNN, input sẽ được đẩy vào tuần tự, tính toán lâu do không tận dụng được tính toán song song của GPU thì Transformer lại đưa tất cả input vào cùng một lúc. Tức là lúc này, Transformer không sử dụng khái niệm time-step nữa. Để thực hiện được cơ chế này, Transformer đã sử dụng Self Attention.

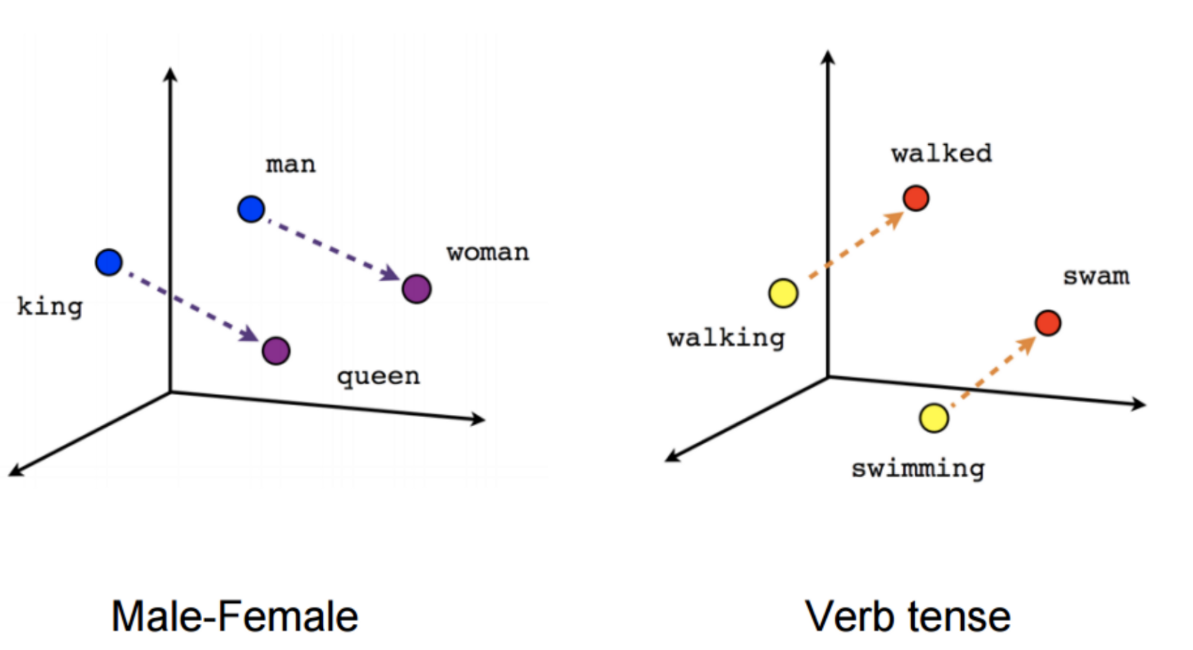


Hình 4-: Kiến trúc transformer [89]

1. **Khối Encoder**
2. **Thành phần Input Embedding**

Đối với máy tính, nó không thể hiểu được câu chữ như con người chúng ta mà nó chỉ hiểu được con số, vector, ma trận. Do đó, chúng ta phải tìm cách biểu diễn câu, chữ thành dạng số, vector, ma trận để máy tính có thể hiểu được. Việc này được gọi là Input Embedding. Tuy nhiên, việc biểu diễn này không phải chỉ đơn giản là biến câu, chữ thành những vector, ma trận mà còn phải đảm bảo sao cho việc biểu diễn chúng vẫn mang một phần ý nghĩa thông tin của chúng, ví dụ những từ gần nghĩa với nhau phải có biểu diễn gần nhau.

Hiện tại đã có nhiều pretrained model cho việc này mà ta có thể sử dụng: Glove, Word2Vec, … Những pretrained model này đảm bảo việc những từ gần nghĩa với nhau sẽ có biểu diễn gần nhau



Hình 4-: Ví dụ biểu diễn input vector của các từ gần nghĩa với nhau trên hệ trục tọa độ [90]

1. **Thành phần Positional Encoding**

Words Embedding phần nào giúp ta hiểu rõ ngữ nghĩa của một từ, tuy nhiên trong thực tế ngữ nghĩa của một từ còn phụ thuộc vào vị trí của nó trong câu.

Ví dụ câu: Học sinh học Sinh học

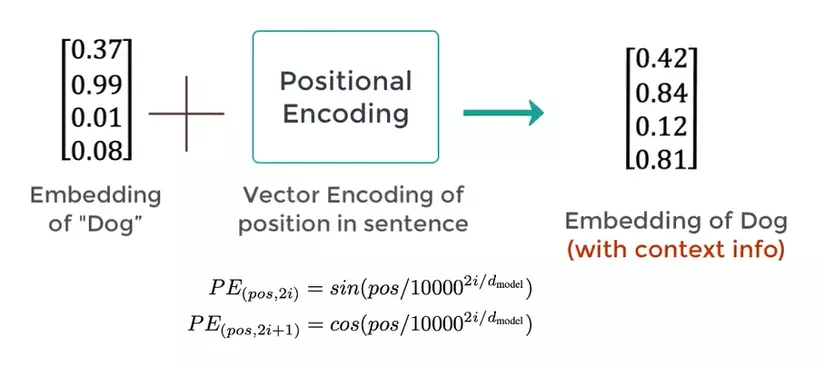
Có 3 từ Học và 2 từ Sinh. Nếu chúng ta dùng word Embedding thì cả 3 từ học đều mang cùng biểu diễn, 2 từ sinh cũng tương tự như vậy.

Những có thể dễ dàng nhận thấy rằng, 3 từ học đều mang ý nghĩa khác nhau. Do đó, Transformers cần có thêm 1 thành phần nữa là Positional Encoding để có thêm thông tin về vị trí của từ đó.

Trong đó,

* pos chính là vị trí của từ trong câu
* PE là giá trị phần tử thứ i trong embedding.

Sau đó ta cộng PE vector với Embedding vector:



Hình 4-: Minh họa thành phần Positional Encoding [89]

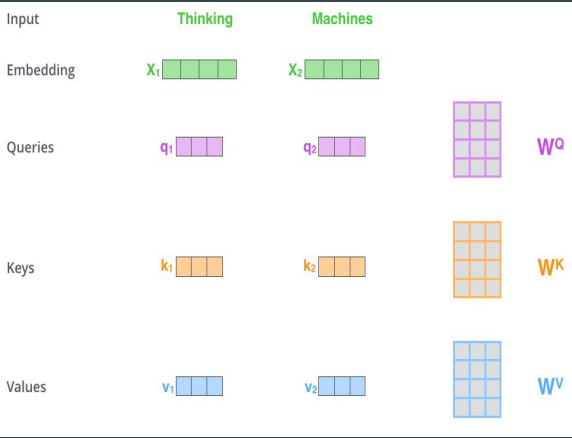
1. **Thành phần Self-Attention**

Như đã trình bày ở trên, Self-Attention là cơ chế giúp Transformers hiểu được sự liên quan giữa các từ trong câu.

Cụ thể cơ chế được giải thích như sau:

* Bước 1: Tạo ra 3 vector đầu vào của Encoder đối với mỗi từ (tương ứng với 3 mũi tên ở phía trên mô hình)
* Một vector Query (Q)
* Một vector Key (K)
* Một vector Value (V)

Và để có được 3 vector này, input embeddings được nhân với 3 ma trận trọng số tương ứng WQ, WK, QV (3 ma trận này được tune trong quá trình huấn luyện mô hình).



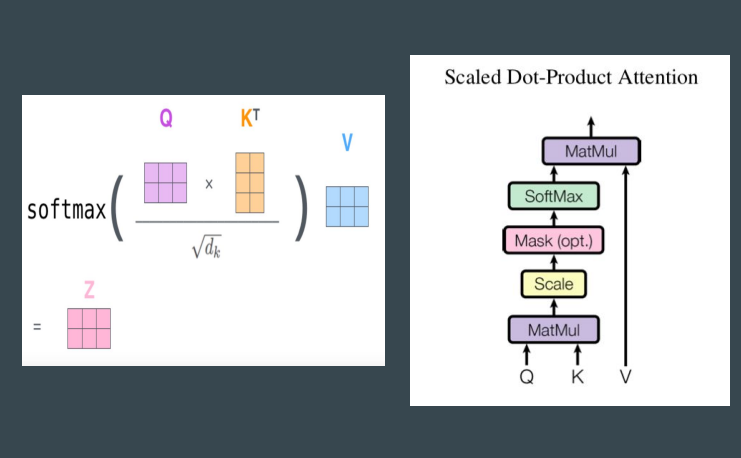
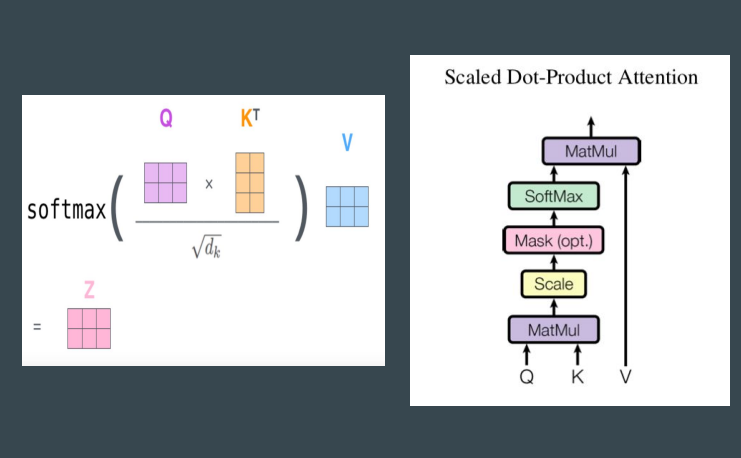
Hình 4-: Minh họa cho việc Tạo ra 3 vector đầu vào của Encoder [89]

* Bước 2: Tính score: Score được tính bằng cách lấy vector Query q1 nhân với vector Key k1
* Bước 3: Lấy điểm Score vừa tính ra chia căn bậc hai của kích thước vector Key.
* Bước 4: Đưa kết quả vừa tính ra qua một hàm Softmax

Tóm lại từ bước 2 đến bước 6 ở trên có thể viết gọn lại như sau để tạo ra một self-attention layer

Tính toán ma trận trong Self-Attention:

* Bước đầu tiên là tính toán Query, Key, và ma trận Value. Chúng ta làm điều đó bằng cách đóng gói imput embeddings của chúng ta vào Ma trận X, và nhân nó với các ma trận trọng lượng mà chúng ta đã được đào tạo (WQ, WK, WV).
* Mỗi hàng trong ma trận X tương ứng với một từ trong câu đầu vào.



Hình 4-: Minh họa cách tạo self-attention layer [89]

1. **Thành phần multi-head attention:**

Có thể dễ dàng nhận biết rằng đối với self- attention, một từ sẽ luôn đặt sự chú ý nhiều nhất vào nó. Bởi với mỗi từ, thì từ liên quan tới nó nhiều nhất chính là nó.

Nhưng trong bài toán NLP, chúng ta lại không mong muốn điều này xảy ra, thứ chúng ta thật sự mong muốn là sự liên quan giữa các từ trong câu. Do đó, ý tưởng multi-head Attention đã ra đời với mong muốn, mỗi thành phần self-attention sẽ chú ý vào đến một thành phần khác nhau trong câu.

Với Multi-head Attention, chúng ta sẽ duy trì các Ma trận Q/K/V riêng biệt cho mỗi phần self- attention dẫn đến có nhiều ma trận trận Q/K/V khác nhau. Như đã làm trước đây, chúng ta nhân x với WQ/WK/WV ma trận để tạo ra ma trận q/k/v.

Tuy nhiên chúng ta gặp phải một vấn đề là lớp feed-forward không thể xử lý nhiều ma trận, mà chỉ có thể một ma trận duy nhất. Do đó, chúng ta liên kết các ma trận lại thành một ma trận duy nhất bởi một ma trận trọng số W0.

Quá trình kết hợp kết quả từ các multi-head attention:

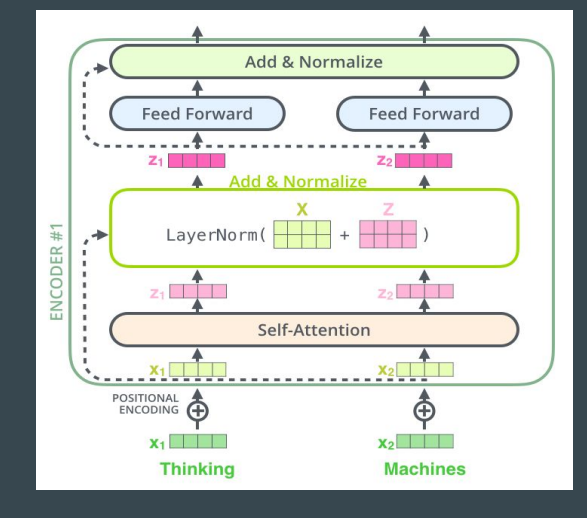
* Kết hợp (concatenate) tất cả các multi-head attention.
* Nhân với Ma trận trọng số W0.
* Sau đó, chúng ta nhận được một ma trận Z chứa thông tin từ tất cả các multi-head attention

Tổng kết lại các bước trong multi-head attention

* Mã hóa từng từ trong câu đầu vào
* Xây dựng 8 multi-head attention: xây dựng 3 ma trận WQ, WK, WV tương ứng với từng head attention
* Tính attention dựa trên 3 ma trận Q, K, V của từng head attention
* Kết hợp (concatenate) tất cả các multi-head attention.
* Nhân với Ma trận trọng số W0.
* Sau đó, chúng ta nhận được một ma trận Z chứa thông tin từ tất cả các multihead attention

1. **Thành phần Residual:**

Trong mô hình trên thì mỗi sub-layer đều là một residual block. Skip connections trong Transformers cho phép thông tin đi qua sub-layer trực tiếp. Thông tin này (x) được cộng với attention (z) của nó và thực hiện Layer Normalization.



Hình 4-: Minh họa thành phần Residual [89]

1. **Thành phần feed forward:**

Sau khi được normalize, các vector z sẽ được đưa qua mạng fully connected trước khi được đẩy qua Decoder.

1. **Thành phần Decoder:**

Quá trình decode về cơ bản là giống với encode, chỉ khác là Decoder decode từng từ một và input của Decoder.

1. **BERT**

Việc tiền huấn luyện các mô hình ngôn ngữ (language model pretraining) đã được chứng minh là mang lại hiệu quả cho bài toán xử lý ngôn ngữ tự nhiên [91] [92]. Có hai chiến lược để áp dụng các biểu diễn ngôn ngữ đã được tiền huấn luyện: dựa vào tính năng (featured – based) và dựa vào tinh chỉnh (fine – tunning). Cả hai chiến lược này có chung một mục tiêu huấn luyện đó chính là sử dụng các mô hình ngôn ngữ đơn hướng (unidirectional language) để tìm hiểu các biểu diễn ngôn ngữ chung (general language representations). Năm 2019, Jacob Devlin cùng cộng sự nhận ra kỹ thuật tại thời điểm đó làm hạn chế sức mạnh của các biểu diễn tiền huấn luyện, đặc biệt là với hướng tiếp cận fine-tuning. Điểm hạn chế lớn nhất nằm ở việc các mô hình ngôn ngữ là đơn hướng, điều này hạn chế quá trình lựa chọn các kiến trúc có thể được sử dụng cho quá trình tiền huấn luyện.

Để giải quyết vấn đề trên, Jacob Devlin đã đề xuất cải tiến cho phương pháp fine-tuning bằng mô hình BERT (Bidirectional Encoder Representations from Transformers) [12]. BERT giúp giảm bớt ràng buộc về tính đơn hướng bằng cách sử dụng mô hình ngôn ngữ ẩn (masked language model - MLM). Không giống với việc tiền huấn luyện từ trái sang phải (left-toright language model pre-training), MLM cho phép bối cảnh của từ được học từ cả 2 phía bên trái và bên phải cùng một lúc từ những bộ dữ liệu, điều này cho phép mô hình tiền huấn luyện một deep bidirectional Transformer.

Mô hình BERT đạt được state-of-the-art trên mười một tác vụ xử lý ngôn ngữ tự nhiên, bao gồm tăng điểm GLUE lên 80.5% (cải thiện 7.7%), độ chính xác MultiNLI lên 86.7% (cải thiện 4.6%). Với F1 cho tác vụ question answering cho tập test SQuAD v1.1 và SQuAD v2.0 lần lượt là 93.2 (tăng 1.5) và 83.1 (tăng 5.1).

1. **BIOBert**

Số lượng tài liệu Y sinh ngày càng tăng, dẫn đến việc khai thác văn bản y sinh ngày càng quan trọng. Cùng với sự tiến bộ trong lĩnh vực xử lý ngôn ngữ tự nhiên (NLP), việc trích xuất thông tin từ văn bản Y sinh trở nên phổ biến trong giới nghiên cứu đã góp phần thúc đẩy sự phát triển của các mô hình khai thác văn bản Y sinh hiệu quả. Tuy nhiên, việc áp dụng trực tiếp các giải pháp NLP tiên tiến để khai thác cho văn bản Y sinh còn gặp nhiều hạn chế. Trước tiên, do các mô hình biểu diễn từ như Word2Vec, ELMo và BERT được huấn luyện trên bộ ngữ liệu trong lĩnh vực tổng quát, dẫn đến khó khăn trong việc ước tính hiệu suất của chúng đối với lĩnh vực Y sinh. Ngoài ra, sự khác biệt giữa Cấu trúc Đối số Vị ngữ giữa văn bản trong lĩnh vực tổng quát và lĩnh vực Y sinh cũng là một vấn đề lớn đối với các mô hình khai thác văn bản Y sinh.

Năm 2020, Jinhyuk Lee cùng cộng sự đã đề xuất mô hình dựa trên giả thuyết mô hình BERT có nhu cầu huấn luyện với kho ngữ liệu thuộc lĩnh vực Y sinh để xử lý tác vụ khai thác văn bản Y sinh. Dù BERT đã chứng minh được tính hiệu quả của các contextualized word representations, nhưng nó không thể đạt được hiệu suất cao do được tiền huấn luyện trên kho ngữ liệu thuộc lĩnh vực tổng quát. Vì BERT đưa ra kết quả tốt đối với tác vụ NLP khác nhau trong khi sử dụng cấu trúc gần như tương tự cho các tác vụ, nên công trình lựa chọn xem xét việc điều chỉnh BERT cho lĩnh vực Y sinh có thể mang lại lời ích cho nhiều nghiên cứu xử lý ngôn ngữ nhiên cho lĩnh vực này.

Mô hình được đề xuất là BioBERT [14] (Bidirectional Encoder Representations from Transformers for Biomedical Text Mining). Với việc sử dụng kiến trúc gần như tương tự nhau giữa các tác vụ, BioBERT đa phần vượt trội hơn so với mô hình BERT state-of-the-art trước đây. BioBERT đạt hiệu suất vượt trội hơn so với BERT trên ba tác vụ khai thác văn bản Y sinh tiêu biểu sau: biomedical named entity recognition (F1 tăng 0.62%), biomedical relation extraction (F1 tăng 2.80%) và biomedical question answering (MRR tăng 12.24%)

1. **Giới thiệu mô hình**

Thách thức mà bài toán SRL cho văn bản Y sinh đang gặp phải nằm ở việc ngữ liệu dược gán nhãn sẵn vẫn còn hạn chế. Để có thể huấn luyện được máy tính hiểu được các sự kiện mà câu muốn truyền tải ở từng câu trong lĩnh vực Y Sinh, chúng ta cần một kho ngữ liệu có gắn nhãn Cấu trúc Đối số Vị ngữ với kích thước đủ lớn (ít nhất khoảng 3000 mẫu). Để khắc phục vấn đề thiếu ngữ liệu huấn luyện, Khóa luận lựa chọn sử dụng bộ ngữ liệu PASBio+. Với việc sử dụng PASBio+, khóa luận có được ngữ liệu huấn luyện gán nhãn PASBio kích thước hơn 10000 câu. Ngoài ra, với việc được huấn luyện trên bộ ngữ liệu có kích thước lớn, Khóa luận cũng lựa chọn BERT như một giải pháp cho vấn đề thiếu hụt ngữ liệu để huấn luyện cho mô hình khi nó được huấn luyện trên một lượng dữ liệu lớn. Tuy nhiên, do BERT được huấn luyện trên bộ ngữ liệu thuộc lĩnh vực tổng quát, Khóa luận cũng cần phải thực hiện thêm một bước fine-tuning BioBERT để chuyên biệt về bài toán SRL cho lĩnh vực Y sinh.

Các công trình trước đó cũng đã chứng minh đóng góp của tri thức ngữ pháp vào hiệu suất của bài toán SRL, cũng chính vì lý do này đã là động lực thúc đẩy Khóa luận quyết định lựa chọn đưa tri thức ngữ pháp vào mô hình. Tuy nhiên, việc nhúng trực tiếp ngữ pháp vào mô hình là một thách thức lớn khi phải đào sâu vào mã nguồn của mô hình tiền huấn luyện đồ sộ và phức tạp. Nhưng đó không phải lý do chính khiến cho khóa luận không lựa chọn nhúng trực tiếp tri thức ngữ pháp vào mô hình SRL. Lý do chính lại liên quan đến khâu triển khai. Nếu nhúng trực tiếp tri thức ngữ pháp vào mô hình học sâu, việc chạy mô hình trên dữ liệu mới lúc triển khai sẽ đòi hỏi bất cứ dữ liệu đầu vào nào cũng phải được phân tích cú pháp sẵn, bao gồm việc phân tích ngữ pháp và phân tích quan hệ phụ thuộc. Điều này tạo một gánh nặng lớn cho việc ứng dụng mô hình trên văn bản mới và làm giảm đáng kể hiệu suất của mô hình về mặt thời gian xử lý. Vì thế, Khóa luận quyết định sử dụng phương pháp xây dựng giàn giáo cú pháp thông qua học đa tác vụ để đưa tri thức ngữ pháp vào mô hình một cách gián tiếp. Nhờ vào việc sử dụng giàn giáo cú pháp, giờ đây tri thức ngữ pháp chỉ đóng vai trò “giàn giáo” giúp cũng cố mô hình trong quá trình huấn luyện, Bên cạnh đó, Khóa luận cũng thực hiện phân tích, đánh giá độ hiệu quả khi thực hiện học đa tác vụ giữa tác vụ chính SRL và hai tác vụ phụ Phân tích cú pháp và Phân tích quan hệ phụ thuộc.

1. **Xây dựng mô hình**
2. **Data transformation**

Chuyển đổi dữ liệu là bước đầu tiên trong mô hình multi-task. Bởi dữ liệu ở nhiều định dạng khác nhau, ở trong phạm vi khóa luận là 3 nguồn dữ liệu cho 3 tác vụ mà mỗi tác vụ lại có dữ liệu đầu vào khác nhau. Cụ thể:

* Đối với tác vụ SRL, dữ liệu đầu vào sẽ là câu và danh sách các vị ngữ của nó.
* Với tác vụ Dependency, thì đầu vào lại là từ và sự phụ thuộc của nó với từ khác trong câu, do đó tác vụ này sẽ chứa 2 nhãn: mối quan hệ phụ thuộc và vị trí mà từ mà nó phụ thuộc.
* Còn đối với tác vụ Grammar thì đầu vào là cấu trúc ngữ pháp của câu, thường biểu diễn dưới dạng cây.

Do đó, chúng ta phải chuyển dữ liệu từ những định dạng khác nhau thành một dạng duy nhất để sử dụng trong mô hình multi-task. Nhóm quyết định chuyển dữ liệu sang dạng BIO. Sau đó, dữ liệu dưới dạng BIO được ghi trong những file text, được chuyển đối sang định dạng tsv để đưa vào mô hình huấn luyện.

1. **Shared Encoder**

Khái niệm về bộ mã hóa chia sẻ là bắt nguồn từ mong muốn huấn luyện một mô hình duy nhất cho nhiều tác vụ. Để đạt được điều này, mô hình sử dụng một bộ transformer-based encoder centre. Dữ liệu của tất cả các tác vụ sẽ đi qua bộ mã hóa trung tâm này. Bộ mã hóa này được gọi là chia sẻ vì nó chịu trách nhiệm cho phần lớn việc huấn luyện trên tất cả các tác vụ. Hơn nữa, các tiêu đề cụ thể của tác vụ được hình thành trên bộ mã hóa được chia sẻ.

Bộ mã hóa chia sẻ này sử dụng một khái niệm gọi là Task specific headers, chính là các trạng thái ẩn của bộ mã hóa được sử dụng bởi các lớp cụ thể của tác vụ được xác định để xuất nhật ký theo định dạng mà tác vụ yêu cầu. Chuyển tiếp cho lô dữ liệu thuộc về tác vụ A nói trên xảy ra thông qua tiêu đề và bộ mã hóa dùng chung cho tác vụ A. Hàm mất mát được tính toán (được gọi là 'task loss') được lan truyền ngược qua cùng một đường dẫn.

1. **Predicate Indicator Embedding**

Tầm quan trọng của vị ngữ trong bài toán gán nhãn cấu trúc đối số vị ngữ là không thể bàn cãi. Việc mô hình biết được chính xác vị trí của vị ngữ ở đâu trong câu sẽ giúp phần nào làm cho mô hình dễ dàng nhận biết chính xác hơn cấu trúc đối số vị ngữ của cả câu. Tuy nhiên, hiện tại mô hình lại không biết trước vị trí của đối số mà cần phải được huấn luyện để biết được đâu là đối số, lúc này mô hình sẽ chịu hai gánh nặng là vừa xác định đối số vừa xác định vị ngữ.

Ví dụ như câu: I saw a saw

Lúc này có 2 từ saw và sẽ có thể gây nhầm lẫn trong quá trình nhận diện vị từ “saw” vì nó xuất hiện ở hai vị trí khác nhau với 2 nghĩa khác nhau.

Để nhấn mạnh tầm quan trọng của vị ngữ trong việc xác định đối số, khóa luận đề xuất sử dụng Nhúng chỉ định vị ngữ (Predicate Indicator Embedding) để tăng cường sự đóng góp của vị ngữ. Nói rõ hơn là sẽ tạo một vector tương tự như vector input\_mask (vector dùng để xác định những vị trí có thông tin của từ) nhưng thay vì đánh dấu vị trí của các từ, vector này sẽ đánh dấu vị trí của vị ngữ trong câu.

Ví dụ, với một câu có vị từ chính là abolished: A G-to-A transition at the first nucleotide of intron 2 of patient 1 abolished normal splicing

Ta có Vector nhúng vị từ là:

0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0

Ta có Vector input mask là:

1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1

Sau đó cộng 2 vector này lại với nhau sẽ được vector:

1 1 1 1 1 1 1 1 1 1 1 1 1 2 1 1

Cuối cùng, sau khi dùng hàm matmul, sẽ có được vector sau:

0 0 0 0 0 0 0 0 0 0 0 0 0 1000 0 0

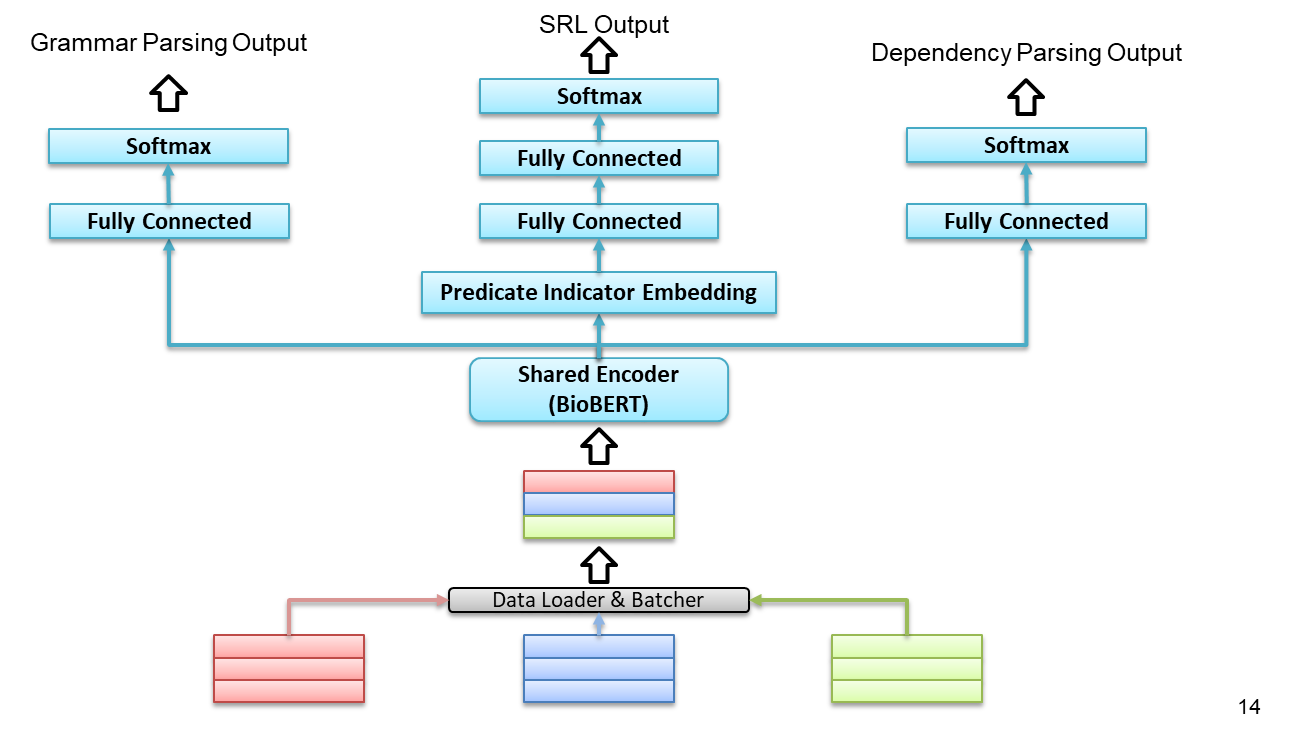
Có thể thấy, với số điểm cao hơn thì vị từ tỉ lệ phát hiện vị từ trong câu cũng sẽ cao hơn. Với thông tin này, quá trình mã hóa có thể tạo ra các bộ biểu diễn cho câu một cách cụ thể cho mỗi vị từ, từ đó giúp tăng hiệu quả cho mô hình.

1. **Cách định nghĩa mô hình multi-task**

Đối với neuron network, cần phải xác định trước các tầng (layer) mà các tác vụ có thê chia sẻ và dùng chung. Hiện tại, có nhiều cách để ó thể xác định tham số của mô hình hay tầng mạng được chia sẻ, nhưng nhìn chung, có hai hướng tiếp cận sau:

* Chia sẻ hoàn toàn (Fully – shared): là loại mô hình chia sẻ hoặc dùng chung tham số của mô hình, trừ tầng đầu ra. Với mỗi tác vụ riêng việt sẽ có một tầng đầu ra khác nhau.
* Phân cấp (Hierarchical): là loại mô hình chia các tác vụ thành các cấp. Sự phân cấp này được thể hiện thông qua việc số tầng mà mỗi tác vụ được huấn luyện. Mặc dù có chung một tầng đầu vào, nhưng tác vụ cấp cao hơn sẽ được huấn luyện nhiều tầng hơn so với tác vụ cấp thấp. Mục đích của việc này là để phân biệt rõ vai trò của tác vụ phụ và chính, khi tác vụ cấp thấp chỉ đóng vai trò bổ trợ cho tác vụ cấp cao.

Trong phạm vi khóa luận này, tác vụ chính mà khóa luận quan tâm là Rút trích cấu trúc đối số vị ngữ trong câu. Do đó, mô hình được đề xuất sử dụng là mô hình Phân cấp, tức là mô hình sẽ dùng chung bộ encoder BioBert, tuy nhiên sau đó mô hình sẽ tập trung chăm chút hơn cho tác vụ SRL bằng cách sử dụng số lớp layer nhiều hơn, cụ thể ở đây là 2 lớp fully connected layer và 1 lớp softmax, cùng với sự hỗ trợ của thành phần Predicate Indicator Embedding.



Hình 4-: Kiến trúc tổng quát của mô hình do khóa luận đề xuất

Bối cảnh mà khóa luận quan tâm có 3 tác vụ: Tác vụ A, Tác vụ B và Tác vụ C để đào tạo cùng nhau. Tác vụ A là gán nhãn cấu trúc đối số vị ngữ, Tác vụ B là phân tích quan hệ phụ thuộc và Tác vụ C là phân tích cấu trúc ngữ pháp câu. Chúng ta có thể xác định một task file đề cập đến các chi tiết bắt buộc về các tác vụ theo định dạng YAML (xin xem phần phụ lục). Task file dùng cho mô hình học đa tác vụ có một số những tính chất mà người xây dựng mô hình cần phải lưu ý:

* Trong một task file có thể có một tác vụ hoặc nhiều tác vụ. Do đó, sức mạnh của mô hình học đa tác vụ không chỉ giới hạn trong 1, 2 hay 3 tác vụ mà còn có thể mở rộng ra thêm nhiều tác vụ khác, miễn là tên các tác vụ phải phân biệt vì mô hình xem đó là định danh cho các tác vụ. Trong trường hợp task file chỉ đề cập đến một tác vụ duy nhất, mô hình sẽ hoạt động như mô hình một nhiệm vụ thông thường.
* Model type cho tất cả các tác vụ được đề cập trong task file phải giống nhau vì mô hình sử dụng một bộ mã hóa dùng chung duy nhất được chia sẻ cho tất cả các tác vụ này. Do đó, không thể nào mà tác vụ task A sử dụng BioBERT mà tác vụ task B lại sử dụng BERT.

1. **Traning**

Sau khi đã chuẩn bị dữ liệu, cũng như cấu hình các tác vụ xong thì lúc này chúng ta có thể huấn luyện mô hình học đa tác vụ của mình. Đây là bước cuối cùng trong việc xây dựng mô hình giàn giáo ngữ pháp cho tác vụ SRL. Chi tiết về các thông số cần thiết lập khi huấn luyện mô hình xin xem trong phần phụ lục của khóa luận.

# **Chương 5: THỰC NGHIỆM VÀ KẾT QUẢ**

1. **Mô tả dữ liệu thực nghiệm**
2. **Dữ liệu cho tác vụ SRL**

Bộ ngữ liệu mà khóa luận lựa chọn sử dụng là PASBio+. Tiền thân của bộ ngữ liệu này là 317 câu đã được gán nhãn của bộ ngữ liệu PASBio. Với việc sử dụng PASBio+, khóa luận có được ngữ liệu huấn luyện gán nhãn PASBio kích thước 28439 câu. Việc có được bộ ngữ liệu gán nhãn lớn này là nhờ vào Phương Pháp Ví Dụ Ảo. Dữ liệu sinh bởi phương pháp này được quy định bằng hai quy tắc tuần tự giúp đảm bảo rằng tri thức lĩnh vực Y sinh vẫn luôn được giữ tính đúng đắn. Bộ ngữ liệu có độ phủ rộng trên các cách hành văn tự nhiên do được tăng cường độ phong phú mẫu câu bằng các biến thể ngữ pháp của câu gốc.

Kết quả cho thấy PASBio+ với vai trò là ngữ liệu huấn luyện đã giúp tăng F1 cho mô hình học sâu lên 52.2% và 22.5% lần lượt với mô hình huấn luyện bằng ngữ liệu gốc chưa tăng cường và ngữ liệu thuộc lĩnh vực tổng quát.

1. **Bộ dữ liệu của tác vụ Grammar**

Khóa luận sử dụng bộ dữ liệu Genia Dependency gồm 16984 câu chứa nhãn của tác vụ Grammar. Nhãn của tác vụ này được viết dưới dạng xml theo cấu trúc cây, chứa thông tin cấu trúc ngữ pháp của câu

1. **Bộ dữ liệu của tác vụ Dependency**

Bộ dữ liệu mà nhóm sử dụng là bộ dữ liệu là bộ dữ liệu Genia Dependency Tree do AlienAi xây dựng. Bộ ngữ liệu chứa 20194 câu được viết trên file text.

1. **Kịch bản thử nghiệm**
2. **Chia tập dữ liệu**

Đối với cả 3 bộ dữ liệu cho 3 tác vụ, nhóm tiến hành chia tỉ lệ thành 60% cho train, 20% cho test và 20% cho dev.

Vậy đối với mỗi tác vụ ta có:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Tác vụ | Tổng số câu | Số câu Train | Số câu Dev | Số câu Test | Bộ dữ liệu |
| SRL | 28439 | 17063 | 5867 | 5509 | PasBIO+ |
| Dependency | 20194 | 12116 | 4038 | 4040 | Genia Dependency |
| Grammar | 16984 | 10190 | 3396 | 3398 | Genia Dependency |

Bảng 5-: Thống kê số lượng câu trong mỗi tập

Và đối với chi tiết từng động từ cho tác vụ SRL

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Động từ | Tổng số câu | Số câu Train | Số câu Dev | Số câu Test | Số biến thể ngữ pháp | Tỷ lệ % biến thể ngữ pháp |
| Abolish | 1079 | 647 | 216 | 216 | 185 | 17 |
| Alter | 872 | 523 | 174 | 174 | 154 | 17,5 |
| Begin | 1861 | 1117 | 372 | 372 | 449 | 24 |
| Block | 1295 | 777 | 259 | 259 | 116 | 9 |
| Catalyse | 478 | 287 | 96 | 96 | 125 | 26 |
| Confer | 945 | 567 | 189 | 189 | 245 | 26 |
| Decrease | 654 | 392 | 131 | 131 | 525 | 80,5 |
| Delete | 857 | 514 | 171 | 171 | 183 | 21,5 |
| Develop | 204 | 122 | 41 | 41 | 165 | 81 |
| Disrupt | 439 | 263 | 88 | 88 | 167 | 38 |
| Eliminate | 706 | 424 | 141 | 141 | 91 | 13 |
| Encode | 718 | 431 | 144 | 144 | 699 | 97,5 |
| Express | 1472 | 883 | 294 | 294 | 1248 | 85 |
| Generate | 1393 | 836 | 279 | 279 | 1199 | 86 |
| Inhibit | 1200 | 720 | 240 | 240 | 190 | 16 |
| Initiate | 534 | 320 | 107 | 107 | 176 | 33 |
| Lead | 1796 | 1078 | 359 | 359 | 110 | 6 |
| Lose | 838 | 503 | 168 | 168 | 198 | 23,5 |
| Modify | 376 | 226 | 75 | 75 | 344 | 91,5 |
| Mutate | 839 | 503 | 168 | 168 | 483 | 57,5 |
| Proliferate | 512 | 307 | 102 | 102 | 174 | 34 |
| Recognize | 991 | 595 | 198 | 198 | 177 | 18 |
| Result | 1739 | 1043 | 348 | 348 | 315 | 18 |
| Skip | 275 | 165 | 55 | 55 | 205 | 74,5 |
| Splice | 1358 | 815 | 272 | 272 | 434 | 70,5 |
| Transcribe | 903 | 542 | 181 | 181 | 368 | 41 |
| Transform | 1188 | 713 | 238 | 238 | 348 | 29,5 |
| Translate | 2399 | 1439 | 480 | 480 | 1809 | 75,5 |
| Truncate | 518 | 311 | 104 | 104 | 295 | 57 |

Bảng 5-: Thống kê số lượng câu của từng động từ

Sau khi đánh giá và đề xuất mô hình phù hợp (Chương 4), Khóa luận tiến hành thực nghiệm và so sánh kết quả giữa các mô hình học đa tác vụ giữa tác vụ chính SRL và hai tác vụ phụ để có thể nhìn nhận đánh giá khách quan hơn. Cụ thể như sau:

* Mô hình đơn tác vụ SRL sử dụng pretrained model BioBERT
* Mô hình đa tác vụ kết hợp giữa 2 tác vụ: SRL với Dependency sử dụng BioBERT
* Mô hình đa tác vụ kết hợp giữa 2 tác vụ: SRL với Grammar sử dụng BioBERT
* Mô hình đa tác vụ kết hợp giữa 3 tác vụ: SRL với Dependency và Grammar sử dụng BioBERT

Với thông số cấu hình của mỗi mô hình là số epoch = 50, batch size = 5 và learning rate = 0.00002.

1. **Phương pháp đánh giá**

Khóa luận tiến hành đánh giá model trên 29 động từ trên tập dữ liệu Dev và Test.

Và để có cái nhìn chi tiết hơn đối với từng động từ, Khóa luận cũng tiến hành đánh giá đối với từng động từ trên tập Dev và Test của động từ đó mà Khóa luận xây dựng.

1. **Kết quả**

Kết quả huấn luyện mô hình đơn tác vụ SRL



Bảng 5-: Kết quả thực nghiệm đối với mô hình đơn tác vụ SRL trên tập DEV và TEST

Kết quả huấn luyện mô hình đa tác vụ SRL + Dependency và so sánh độ tăng F1 so với mô hình đơn tác vụ SRL



Bảng 5-: Kết quả thực nghiệm đối với mô hình đa tác vụ SRL + Dependency trên tập DEV và TEST

Kết quả huấn luyện mô hình đa tác vụ SRL + Grammar và so sánh độ tăng F1 so với mô hình đơn tác vụ SRL



Bảng 5-: Kết quả thực nghiệm đối với mô hình đa tác vụ SRL + Grammar trên tập DEV và TEST

Kết quả huấn luyện mô hình đa tác vụ SRL + Grammar + Dependency và so sánh độ tăng F1 so với mô hình đơn tác vụ SRL



Bảng 5-: Kết quả thực nghiệm đối với mô hình đa tác vụ SRL + Dependency + Grammar trên tập DEV và TEST

Thời gian thực thi của các mô hình:

|  |  |  |
| --- | --- | --- |
| Thời gian huấn luyện  1 epoch | Thời gian huấn luyện  50 epochs | Thời gian thực thi 1 câu |
| 12 phút 27 giây | 10 giờ 22 phút 30 giây | 0.04 giây |

Bảng 5-: Thời gian huấn luyện và thời gian thực thi mô hình đơn tác vụ SRL

|  |  |  |
| --- | --- | --- |
| Thời gian huấn luyện  1 epoch | Thời gian huấn luyện  50 epochs | Thời gian thực thi 1 câu |
| 16 phút 55 giây | 14 giờ 5 phút 50 giây | 0.04 giây |

Bảng 5-: Thời gian huấn luyện và thời gian thực thi mô hình đa tác vụ SRL + Dependency

|  |  |  |
| --- | --- | --- |
| Thời gian huấn luyện  1 epoch | Thời gian huấn luyện  50 epochs | Thời gian thực thi 1 câu |
| 16 phút 4 giây | 13 giờ 23 phút 20 giây | 0.04 giây |

Bảng 5-: Thời gian huấn luyện và thời gian thực thi mô hình đa tác vụ SRL + Grammar

|  |  |  |
| --- | --- | --- |
| Thời gian huấn luyện  1 epoch | Thời gian huấn luyện  50 epochs | Thời gian thực thi 1 câu |
| 21 phút 29 giây | 17 giờ 54 phút 10 giây | 0.04 giây |

Bảng 5-: Thời gian huấn luyện và thời gian thực thi mô hình đa tác vụ SRL + Grammar + Dependency

Với các mô hình được chạy trên máy có cấu hình như sau: Intel(R) Xeon(R) CPU @ 2.30GHz, CPU 1 core, 2 thread, RAM 13GB, GPU Tesla T4

1. **Thảo luận**

**Bảng 5-3** trình bày kết quả thực nghiệm của mô hình khi chạy đơn tác vụ. Có thể thấy F1 của toàn mô hình khá tốt, đạt F1 81.8 trên tập Dev và 81.64 trên tập Test. Trong đó có thể thấy có một số động từ có F1 rất tốt (trên 85%) ví dụ như begin, catalyse, confer... Tuy nhiên cũng có một số động từ có F1 khá thấp (dưới 70%) như block, eliminate, encode … Điều này có thể lý giải là do:

* Thứ nhất số lượng câu để huấn luyện của các động từ có sự chênh lệch. Ví dụ đối với động từ Begin là 1861 câu trong khi động từ decrease lại chỉ có 654
* Thứ hai, đến từ bộ ngữ liệu PASBio+, do bộ ngữ liệu này sử dụng các biến thể của câu gốc để làm tăng độ phong phú mẫu câu. Vì vậy, những động từ có sự đa dạng về đối số hơn sẽ có lợi hơn trong quá trình huấn luyện.

**Bảng 5-4** trình bày kết quả thực nghiệm của mô hình đa tác vụ kết hợp của SRL + Dependency. F1 của mô hình đạt được F1 86.83 trên tập Dev và 86.74 trên tập Test. Đầu tiên có thể thấy kết quả của mô hình này tốt hơn so với mô hình đơn tác vụ SRL, cụ thể: đối với tập Dev, F1 đã tăng 5.03 (86.83 so với 81.8) và 5.1 trên tập Test (86.74 so với 81.64). Điều này chứng minh, Dependency đã mang lại tác động tích cực cho mô hình, giúp cải thiện tác vụ SRL. Thêm vào đó, một số động từ, ví dụ như động từ eliminate đã hưởng lợi rất lớn từ tác vụ Dependency, F1 đã tăng 13.91 trên tập DEV và 15.87 trên tập TEST. Bên cạnh đó, cũng có những động từ điểm F1 giảm so với mô hình đơn tác vụ SRL. Khóa luận nhận định trường hợp này có thể do bộ ngữ liệu cho tác vụ Dependency không hỗ trợ, hoặc hỗ trợ ít cho những động từ này. Điều này làm cho trong quá trình huấn luyện, động từ không nhận được nhiều tri thức ngữ pháp, trái lại còn gây rối cho động từ đó trong quá trình huấn luyện, dẫn đến việc F1 giảm.

**Bảng 5-5** trình bày kết quả thực nghiệm của mô hình đa tác vụ kết hợp giữa SRL + Grammar. F1 của mô hình đạt được F1 86.26 trên tập DEV và 86.03 trên tập TEST Cũng như tác vụ Dependency, tác vụ Grammar cũng đã mang lại nhiều thông tin, từ đó cải thiện chất lượng của tác vụ chính SRL. Cụ thể, F1 trên tập DEV đã tăng 6.89 và 6.78 trên tập Test đối với động từ translate. Tuy nhiên, so sánh giữa mô hình SRL + Dependency (2) và mô hình SRL + Grammar (3), có thể nhận thấy rằng mô hình (2) tốt hơn mô hình (3). Điều này có thể lý giải là do tác vụ Grammar phức tạp hơn tác vụ Dependency rất nhiều, nhưng lại không có mối liên hệ chặt chẽ với tác vụ SRL như tác vụ Dependency.  Thật vậy, trong khi tác vụ Dependency chỉ phải học mối liên kết giữa các từ trong câu thì tác vụ Grammar phải làm nhiều hơn thế. Tri thức Dependency thể hiện rất rõ rệt mối quan hệ phụ thuộc giữa các đối sối với vị ngữ của chúng, nhờ đó trở thành một chỉ điểm rất tốt cho tác vụ SRL. Ngược lại, tác vụ Grammar phải học được tri thức ngữ pháp trong câu, thường biểu diễn bằng dạng cây phức tạp và phủ trên cả cấu trúc đối số vị ngữ và những thành phần khác trong câu không liên quan đến PAS. Từ thực nghiệm, khóa luận nhận định có thể chính điều này đã phần nào tạo nhiễu khiến tác vụ Grammar không hỗ trợ cho tác vụ SRL tốt bằng tác vụ Dependency. Việc huấn luyện tác vụ Dependency là đơn giản và dễ dàng hơn, nhưng Dependency lại cung cấp nhiều tri thức cú pháp trực tiếp hữu ích hơn cho tác vụ chính so với tác vụ Grammar.

**Bảng 5-6** trình bày kết quả thực nghiệm của mô hình đa tác vụ kết hợp giữa 3 task SRL + Grammar + Dependency. Mô hình này đạt được F1 89.2 trên tập Dev và 89.1 trên tập Test, cao hơn 7.41 và 7.47 so với mô hình đơn tác vụ SRL lần lượt trên tập DEV và TEST. Thêm vào đó, mô hình cũng đạt điểm F1 cao hơn so với mô hình (2), (3). Do đó, chúng ta có thể thấy tác động tích cực của 2 tác vụ Grammar và Dependency đối với tác vụ SRL, cả 2 tác vụ đã cùng phối hợp, tạo thành “giàn giáo” để giúp tác vụ SRL “trưởng thành”.

Qua bảng kết quả về thời gian huấn luyện, mô hình (4) có thời gian huấn luyện là lâu hơn so với mô hình (1), (2), (3), điều này được lý giải bởi mô hình (4) phải huấn luyện cả 3 tác vụ, bộ dữ liệu huấn luyện là lớn hơn nhiều so với 3 mô hình (1), (2), (3).

Thông thường, để có sự chính xác càng cao thì phải đánh đổi thời gian xử lý, tuy nhiên mô hình mà khóa luận đề xuất lại không có sự đánh đổi đó. Cụ thể thời gian để thực thi 1 câu đầu vào để cho ra output là không có sự chênh lệch nhiều giữa mô hình với (4) so với mô hình (1), (2), (3).

Điều này chứng tỏ mô hình mà khóa luận đề xuất là không có sự đánh đổi giữa thời gian và chất lượng mô hình. Chính điều này làm nổi bật giá trị của “Giàn giáo” mà khóa luận đã mô tả trong chương 4: Sau khi huấn luyện xong thì giàn giáo được tháo gỡ đi, không còn gánh nặng về thời gian xử lý cho tác vụ chính nữa.

# **Chương 6: KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN**

1. **Kết luận**
2. **Các lý thuyết đã tìm hiểu**

* Các công trình Deep Learning cho bài toán “Rút trích cấu trúc đối số vị ngữ’.
* Bộ ngữ liệu PASBio+.
* Mô hình ngôn ngữ tiền huấn luyện BERT cho lĩnh vực tổng quát và BioBERT cho lĩnh vực Y sinh.
* Mô hình học đa tác vụ (multi-tasks learning).

1. **Đóng góp của khóa luận**

Xây dựng mô hình Deep Learning cho tác vụ SRL trên lĩnh vực Y sinh và giải quyết vấn đề thiếu ngữ liệu huấn luyện bằng cách sử dụng mô hình ngôn ngữ tiền huấn luyện BioBERT. Đồng thời Khóa luận cũng thực hiện đưa tri thức ngữ pháp vào bài toán SRL bằng cách xây dựng mô hình giàn giáo cú pháp.

1. **Hướng phát triển**

Ứng dụng mô hình cho các lĩnh vực khác ngoài Y sinh khi có bộ ngữ liệu lớn.

Nghiên cứu thêm các chiến lược khác để cải tiến mô hình nhằm nâng cao độ đo F1.

# **TÀI LIỆU THAM KHẢO**

|  |  |
| --- | --- |
| [1] | Enderle, J. and Bronzino, J., Introduction to Biomedical Engineering, Elsevier Science, 2011, pp. 16-21. |
| [2] | W. Saltzman, Biomedical Engineering: Bridging Medicine and Technology, Cambridge University Press, 2009, p. xiii. |
| [3] | Hagit Shatkay and Ronen Feldman, Mining the Biomedical Literature in the Genomic Era: An Overview, 2003, pp. 821 - 823. |
| [4] | J. Lyons, Natural Language and Universal Grammar, Cambridge University Press, 1991, pp. 68-70. |
| [5] | Tuangthong Wattarujeekrit, Parantu K Shah & Nigel Collier, "PASBio: predicate-argument structures for event extraction in molecular biology," pp. 155-174, 2004. |
| [6] | Jae-Hong Eom & Byoung-Tak Zhang , "PubMiner: Machine Learning-Based Text Mining System for Biomedical Information Mining International Conference on Artificial Intelligence: Methodology, Systems, and Applications," pp. 216-225, 2004. |
| [7] | L. Tanabe, U. Scherf, L.H. Smith, J.K. Lee, L. Hunter & J.N. Weinstein, "MedMiner: An Internet Text-Mining Tool for Biomedical Information, with Application to Gene Expression Profiling," 2018. |
| [8] | Svetlana Novichkova, Sergei Egorov, Nikolai Daraselia, "MedScan, a natural language processing engine for MEDLINE abstracts," p. 1699–1706, 2003. |
| [9] | T Sekimizu, HS Park, J Tsujii, "Identifying the Interaction between Genes and Gene Products Based on Frequently Seen Verbs in Medline Abstracts," pp. 62-71, 1998. |
| [10] | Paul Thompson, Philip Cotter, John McNaught, Sophia Ananiadou, Simonetta Montemagni, Andrea Trabucco, Giulia Venturi, "Building a Bio-Event Annotated Corpus for the Acquisition of Semantic Frames from Biomedical Corpora," pp. 2159-2166, 2008. |
| [11] | R. Caruana, "Multitask Learning," p. 41–75 , 1977. |
| [12] | Jacob Devlin, Ming-Wei Chang, Kenton Lee, Kristina Toutanova, "BERT: Pre-training of Deep Bidirectional Transformers for," 2018. |
| [13] | Yukun Zhu, Ryan Kiros, Richard Zemel, Ruslan Salakhutdinov, Raquel Urtasun, Antonio Torralba, Sanja Fidler, "Aligning Books and Movies: Towards Story-like Visual Explanations by Watching Movies and Reading Books," pp. 19-27, 2015. |
| [14] | Jinhyuk Lee, Wonjin Yoon, Sungdong Kim, Donghyeon Kim, Sunkyu Kim, Chan Ho So, and Jaewoo Kang, "BioBERT: a pre-trained biomedical language representation model for biomedical text mining," 2019. |
| [15] | Shexia He, Zuchao Li, Hai Zhao, Hongxiao Bai, Gongshen Liu, "Syntax for Semantic Role Labeling, To Be, Or Not To Be," 2018. |
| [16] | Diego Marcheggiani, Anton Frolov, Ivan Titov, "A Simple and Accurate Syntax-Agnostic Neural Model for Dependency-based Semantic Role Labelingc," 2017. |
| [17] | Qingrong Xia, Zhenghua Li, Min Zhang, Meishan Zhang, Guohong Fu, Rui Wang, Luo Si, "Syntax-Aware Neural Semantic Role Labeling," 2019. |
| [18] | Yaoyun Zhang, Min Jiang, Jingqi Wang, Hua Xu, "Semantic Role Labeling of Clinical Text: Comparing Syntactic Parsers," pp. 1285-1287, 2017. |
| [19] | Jiaxun Cai, Shexia He, Zuchao Li, Hai Zhao, "A Full End-to-End Semantic Role Labeler," *Proceedings of the 27th International Conference on Computational Linguistics,* p. 2753–2765, 2018. |
| [20] | A. Carnie, "Syntax: A Generative Introduction, 2th Edition," p. 51, 2006. |
| [21] | Wen-Chi Chou, Richard Tzong-Han Tsai, Ying-Shan Su, Wei Ku, Ting-Yi Sung, Wen-Lian Hsu, "A Semi-Automatic Method for Annotating a Biomedical Proposition Bank," p. 5–12, 2006. |
| [22] | Johnson, Christopher R, Charles J. Fillmore, Esther J. Wood, Josef Ruppenhofer, Margaret Urban, Miriam, "The FrameNet project: Tools for lexicon building," p. 86–90, 1998. |
| [23] | Karin Kipper, Hoa Trang Dang, Martha Palmer, "Class-based construction of a verb lexicon,," p. 691–696, 2000. |
| [24] | Paul Kingsbury, Martha Palmer, "From Treebank to PropBank," p. 38–43, 2002. |
| [25] | Introduction to WordNet: An On-line Lexical Database, Introduction to WordNet: An On-line Lexical Database, International Journal of Lexicography, 1990, p. 235–244. |
| [26] | Mitchell P. Marcus, Beatrice Santorini, Mary Ann Marcinkiewicz, Building a Large Annotated Corpus of English: The Penn Treebank, MIT Press, 1993, p. 313–330. |
| [27] | Martha Palmer, Paul Kingsbury, Paul Kingsbury, The Proposition Bank: An Annotated Corpus of Semantic Roles, Computational Linguistics, 2005, p. 71–106. |
| [28] | W.-L. Hsu, BioProp Version 1.0, Linguistic Data Consortium, 2009. |
| [29] | Nizamani, Sarwat; Memon, Nasrullah; Nizamani, Saad; Nizamani, Sehrish, "TDC: Typed Dependencies-Based Chunking Model," *Arabian Journal for Science and Engineering,* p. 3585–3595, 2017. |
| [30] | J. Park, "Selectively Connected Self-Attentions for Semantic Role Labeling," *Applied Sciences,* 2019. |
| [31] | C. Pollard, I. A. Sag, "Head-Driven Phrase Structure Grammar," 1994. |
| [32] | Liakata, Maria, Stephen G. Pulman, "From Trees To Predicate-Argument Structures," *International Conference on Computational Linguistics 20,* pp. 563-569, 2002. |
| [33] | X. Carreras, L. Màrquez, "Introduction to the CoNLL-2004 shared task: Semantic role labeling," *HLT-NAACL 2004 Workshop 8th Conf. Comput. Natural Lang. Learn.,* pp. 89-87, 2004. |
| [34] | X. Carreras, L. Màrquez, "Introduction To the CoNLL-2005 shared task: Semantic role labeling," *CoNLL (2005),* pp. 152-164, 2005. |
| [35] | Chi-San (Althon) Lin, Tony C. Smith, "Semantic role labeling via consensus in pattern-matching," *CoNLL 5,* pp. 185-188, 2005. |
| [36] | Trond Grenager, Christopher D. Manning, "Trond Grenager, Christopher D. Manning," *Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing,* pp. 1-8, 2006. |
| [37] | Tuangthong Wattarujeekrit, Nigel Collier , "Exploring Predicate-Argument Relations for Named Entity Recognition in the Molecular Biology Domain," *Exploring Predicate-Argument Relations for Named Entity Recognition in the Molecular Biology Domain,* pp. 267-280, 2005. |
| [38] | Ryu Iida, Mamoru Komachi, Kentaro Inui, Yuji Matsumoto, "Annotating a Japanese Text Corpus with Predicate-Argument and Coreference Relations," *the Linguistic Annotation Workshop,* pp. 132-139, 2007. |
| [39] | G. Stevens, "XARA: An XML- and rule-based semantic role labeler," *The Linguistic Annotation Workshop, Annual Meeting of the Association for Computational Linguistics,* 2007. |
| [40] | Minlie Huang, Xiaoyan Zhu, Yu Hao, Donald G. Payan, Kunbin Qu, Ming Li, "Discovering patterns to extract protein–protein interactions from full texts," *Bioinformatics, Volume 20, Issue 18, December 2004,* p. 3604–3612, 2004. |
| [41] | Daniel Gildea and Martha Palmer, "The Necessity of Parsing for Predicate Argument Recognition," *Proceedings of the 40th Meeting of the Association for Computational Linguistics (ACL 2002),* pp. 239-246, 2002. |
| [42] | Blaheta, Don and Eugene Charniak, "Assigning function tags to parsed text," *In Proceedings of the First Annual Meeting of the North American Chapter of the ACL (NAACL),* p. 234–240, 2000. |
| [43] | Daniel Gildea, Daniel Jurafsky, "Automatic labeling of semantic roles," *Computational Linguistics,* pp. 245-288, 2002. |
| [44] | Mihai Surdeanu, Sanda Harabagiu, John Williams, Paul Aarseth, "Using Predicate-Argument Structure for Information Extraction," *Annual Conference on the Association for Computational Linguistics 41,* p. 46–51, 2013. |
| [45] | Richard Tzong-Han Tsai , Wen-Chi Chou, Yu-Chun Lin, Wei Ku, Ying-Shan Su, Ting-Yi Sung, Wen-Lian Hsu, "BIOSMILE: Adapting Semantic Role Labeling for Biomedical Verbs: An Exponential Model Coupled with Automatically Generated Template Features," *BioNLP Conference,* 2006. |
| [46] | Laxmi Chaudhary, Buddha Singh , "Community detection using unsupervised machine learning techniques on COVID-19 dataset," *Social Network Analysis and Mining,* 2021. |
| [47] | Kashif Munir, Hai Zhao, Zuchao Li, "Neural Unsupervised Semantic Role Labeling," *ACM Transactions on Asian and Low-Resource Language Information Processing,* vol. 20, no. 6, pp. 1-16, 2021. |
| [48] | Ivan Titov, Alexandre Klementiev, "A Bayesian Approach to Unsupervised Semantic Role Induction," *Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics,* pp. 12-22, 2012. |
| [49] | Niko Schenk, Christian Chiarcos, "Unsupervised Learning of Prototypical Fillers for Implicit Semantic Role Labeling," *Proceedings of NAACL-HLT 2016,* pp. 1473-1479, 2016. |
| [50] | Giacomo Frisoni, Gianluca Moro, Giulio Carlassare, Antonella Carbonaro, "Unsupervised Event Graph Representation and Similarity Learning on Biomedical Literature," *Role and Challenges of Healthcare Cognitive Computing: From Extraction to Data Analysis Techniques,* 2022. |
| [51] | Prayag Tiwari, Sagar Uprety, Shahram Dehdashti, M. Shamim Hossain, "TermInformer: unsupervised term mining and analysis in biomedical literature," *Neural Computing and Applications,* 2020. |
| [52] | Mohamed Nadif, François Role, "Unsupervised and self-supervised deep learning approaches for biomedical text mining," *Briefings in Bioinformatics, ,* vol. 22, no. 2, p. 1592–1603, 2021. |
| [53] | Robert S. Swier, Suzanne Stevenson, "Exploiting a Verb Lexicon in Automatic Semantic Role Labelling," *EMNLP 05,* pp. 883-890, 2005. |
| [54] | Robert S. Swier, Suzanne Stevenson, "Unsupervised Semantic Role Labeling," *EMNLP 04,* pp. 95-102, 2004. |
| [55] | Alex Krizhevsky, Ilya Sutskever, Geoffrey E. Hinton, "ImageNet Classification with Deep Convolutional Neural Networks," *Advances in Neural Information Processing Systems 25 (NIPS 2012),* 2012. |
| [56] | Lance Ramshaw, Mitch Marcus, "Text Chunking using Transformation-Based Learning," *Third Workshop on Very Large Corpora,* 1995. |
| [57] | Peter Koomen, Vasin Punyakanok, Dan Roth, Wen-tau Yih, "Generalized inference with multiple semantic role labeling systems," *Proceedings of the 9th Conference on Computational Natural Language Learning, CONLL ’05,* p. 181–184, 2005. |
| [58] | Martha Palmer, Daniel Gildea, and Nianwen Xue, "Semantic Role Labeling," *Synthesis Lectures on Human Language Technology Series. Morgan and Claypool,* 2010. |
| [59] | Sameer Pradhan, Alessandro Moschitti, Nianwen Xue, Hwee Tou Ng, Anders Bjorkelund, Olga Uryupina, Yuchen Zhang, Zhi Zhong, "Towards robust linguistic analysis using ontonotes," *Proceedings of the Seventeenth Conference on Computational Natural Language Learning,* p. 143–152, 2013. |
| [60] | Ronan Collobert, Jason Weston, Leon Bottou, Michael Karlen, Koray Kavukcuoglu, Pavel Kuksa, "Natural language processing (almost) from scratch," *Journal of Marchine Learning Research,* p. 2493–2537, 2011. |
| [61] | Jie Zhou, Wei Xu, "End-to-end Learning of Semantic Role Labeling Using Recurrent Neural Networks," *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing,* p. 1127–1137, 2015. |
| [62] | Alex Graves, Marcus Liwicki, Santiago Fernandez, Roman Bertolami, Horst Bunke, Jurgen Schmidhuber, "A novel connectionist system for unconstrained handwriting recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence,* pp. 855-868. |
| [63] | Sanket Vaibhav Mehta, Jay Yoon Lee, Jaime Carbonell, "Towards Semi-Supervised Learning for Deep Semantic Role Labeling," *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing,* p. 4958–4963, 2018. |
| [64] | Luheng He, Kenton Lee, Mike Lewis, Luke Zettlemoyer, "Deep Semantic Role Labeling: What Works and What’s Next," *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics,* vol. 1, p. 473–483, 2017. |
| [65] | Peng Shi and Jimmy Lin, "Simple BERT Models for Relation Extraction and Semantic Role Labeling," *arXiv:1904.05255 [cs.CL],* 2019. |
| [66] | Rico Sennrich, Barry Haddow, và Alexandra Birch, "Neural machine translation of rare word," *Proceedings of the 54th Annual Meeting of the Association for Computational,* vol. 1: Long Papers, pp. 1715-1725, 2016. |
| [67] | Yuhao Zhang, Victor Zhong, Danqi Chen, Gabor Angeli, and Christopher D. Manning, "Positionaware attention and supervised data improve slot filling," *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing,,* pp. 35-45, 2017. |
| [68] | Yuhao Zhang, Peng Qi, and Christopher D. Manning, "Graph convolution over pruned dependency trees improves relation extraction," *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing,,* p. 2205–2215, 2018. |
| [69] | Felix Wu, Tianyi Zhang, Amauri Holanda de Souza Jr, Christopher Fifty, Tao Yu, and Kilian Q. Weinberger, "Simplifying graph convolutional networks," *arXiv:1902.07153,* 2019. |
| [70] | Zuchao Li, Shexia He, Hai Zhao, Yiqing Zhang, Zhuosheng Zhang, Xi Zhou, and Xiang Zhou, "Dependency or span, end-to-end uniform semantic role labeling," *Proceedings of the 33rd AAAI Conference on Artificial Intelligence.,* 2019. |
| [71] | Hiroki Ouchi, Hiroyuki Shindo, and Yuji Matsumoto, "A span selection model for semantic role labeling.," *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing,* p. 1630–1642, 2018. |
| [72] | Hiroki Ouchi, Hiroyuki Shindo, Yuji Matsumoto, "A Span Selection Model for Semantic Role Labeling," *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing,* p. 1630–1642, 2018. |
| [73] | Luheng He, Kenton Lee, Omer Levy, Luke Zettlemoyer, "Jointly Predicting Predicates and Arguments in Neural Semantic Role Labeling," *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Short Papers),* p. 364–369, 2018. |
| [74] | Swabha Swayamdipta, Sam Thomson, Kenton Lee, Luke Zettlemoyerr Chris Dyer, Noah A. Smithr, "Syntactic Scaffolds for Semantic Structures," *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing,* p. 3772–3782, 2018. |
| [75] | Diego Marcheggiani, Anton Frolov, Ivan Titov, "A Simple and Accurate Syntax-Agnostic Neural Model for Dependency-based Semantic Role Labeling," *Proceedings of the 21st Conference on Computational Natural Language Learning (CoNLL 2017),* p. 411–420, 2017. |
| [76] | Diego Marcheggiani, Ivan Titov, "Encoding Sentences with Graph Convolutional Networks for Semantic Role Labeling," *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing,* p. 1506–1515, 2017. |
| [77] | Jiaxun Cai, Shexia He, Zuchao Li, Hai Zhao, "A Full End-to-End Semantic Role Labeler, Syntax-agnostic Over Syntax-aware?," *Proceedings of the 27th International Conference on Computational Linguistics,* p. 2753–2765, 2018. |
| [78] | Yu Zhang, Qingrong Xia, Shilin Zhou, Yong Jiang, Guohong Fu, Min Zhang, "Semantic Role Labeling as Dependency Parsing: Exploring Latent Tree Structures inside Arguments," *Proceedings of the 29th International Conference on Computational Linguistics,* p. 4212–4227, 2022. |
| [79] | J. Eisner, "Bilexical Grammars and their Cubic-Time Parsing Algorithms," pp. 29-61, 2000. |
| [80] | Yu Zhang, Zhenghua Li, Min Zhang, "Effi- cient second-order TreeCRF for neural dependency parsing," *Proceedings of ACL,* p. 3295–3305, 2020. |
| [81] | Yufei Wang, Mark Johnson, Stephen Wan, Yifang Sun, Wei Wang, "How to best use Syntax in Semantic Role Labelling," *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics,* p. 5338–5343, 2019. |
| [82] | Carlos Gómez-Rodríguez, David Vilares, "Constituent parsing as sequence labeling," *Proceedings of the 2018 Conference on Empirical Meth- ods in Natural Language Processing,* pp. 1314-1324, 2018. |
| [83] | Nianwen Xue, Martha Palmer, "Calibrating features for semantic role labeling," *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing,* pp. 88-94, 2004. |
| [84] | Diego Marcheggiani, Ivan Titov, "Graph Convolutions over Constituent Trees for Syntax-Aware Semantic Role Labeling," *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing,* p. 3915–3928, 2020. |
| [85] | Zhixing Tan, Mingxuan Wang, Jun Xie, Yidong Chen, Xiaodong Shi, "Deep Semantic Role Labeling with Self-Attention," *arXiv:1712.01586v1,* 2018. |
| [86] | Fariz Ikhwantri, Samuel Louvan, Kemal Kurniawan, Bagas Abisena, Valdi Rachman, Alfan Farizki Wicaksono, Rahmad Mahendra, "Multi-Task Active Learning for Neural Semantic Role Labeling on Low Resource Conversational Corpus," *Proceedings of the Workshop on Deep Learning Approaches for Low-Resource NLP,* p. 43–50, 2018. |
| [87] | D. D. Hung, "VIBLO," 21 6 2021. [Online]. Available: https://viblo.asia/p/tim-hieu-ve-co-che-attention-924lJjbmlPM. [Accessed 2 2023]. |
| [88] | A. Navon, "github.io," 4 12 2019. [Online]. Available: https://bit.ly/3JQR90Z. [Accessed 2 2023]. |
| [89] | N. V. Anh, "VIBLO," 1 5 2020. [Online]. Available: http://bit.ly/3mQvD3c. [Accessed 2 2023]. |
| [90] | V. John, "ResearchGate," 12 2016. [Online]. Available: https://bit.ly/40bzPbZ. [Accessed 2 2023]. |
| [91] | Matthew Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, Luke, "Deep contextualized word representations," *NAACL,* 2018a. |
| [92] | Alec Radford, Karthik Narasimhan, Tim Salimans, Ilya Sutskever, "Improving language understanding with unsupervised learning," *Technical report, OpenAI,* 2018. |
| [93] | Y. Bengio, P. Frasconi, P. Simard, "The problem of learning long-term dependencies in recurrent networks," *IEEE Transactions on Neural Networks archive,* vol. 5, no. 2, pp. 157-166, 1994. |

# **Phụ lục A: THÔNG SỐ CẤU HÌNH CHO MÔ HÌNH HỌC ĐA TÁC VỤ**

1. **Data transformation**

Cấu trúc của hàm chuyển đổi

* **coNLL\_srl\_to\_tsv** (dataDir, readFile, wrtDir, transParamDict, isTrainFile=False)

Hàm này có thể được sử dụng để thực hiện các phép biến đổi. Hàm được xác định để lấy dữ liệu thô ở định dạng nhất định ở dạng BIO, thực hiện các bước chuyển đổi đã xác định và tạo tệp tsv tương ứng.

Trong đó:

* dataDir (str) – Đường dẫn đến thư mục chứa các tệp dữ liệu BIO cần đọc.
* readFile (str) – Đây là tệp hiện đang được đọc và chuyển đổi bởi hàm.
* wrtDir (str) – Đường dẫn đến thư mục lưu các tệp tsv đã chuyển đổi.
* transParamDict (dict, mặc định là None) – Từ điển các tham số chức năng cụ thể. Không cần thiết cho chức năng chuyển đổi này.

Ngoài ra, chúng ta có thể làm điều đó bằng cách viết hàm biến đổi của riêng mình trong trường hợp cần chuyển đổi một số dữ liệu định dạng tùy chỉnh sang định dạng tsv tiêu chuẩn. Tuy nhiên, cần phải tuân theo một số nguyên tắc nhất định: Hàm phải lấy các đối số đầu vào giống như các hàm Transform như trên. Có thể thêm bất kỳ tham số cụ thể nào của hàm bổ sung vào đối số transParamDict.

1. **Transform File:**

Sau khi đã có hàm để chuyển đổi dữ liệu sang dạng tsv, lúc này chúng ta có thể dễ dàng sử dụng các hàm chuyển đổi hoặc hàm chuyển đổi tự viết bằng cách xác định tệp biến đổi định dạng YAML.

Sau đây là một ví dụ cho tệp chuyển đổi:

sample\_transform:

transform\_func: scoNLL\_srl\_to\_tsv

read\_file\_names:

- srl\_train.txt

- srl\_dev.txt

- srl\_test.txt

read\_dir: snips\_data

save\_dir: demo\_transform

Tham số trong Transform file:

* transform\_func (required): Tên của hàm biến đổi sẽ sử dụng.
* transform\_params (optional): Từ điển các tham số cụ thể của hàm sẽ có trong tham số transParamDict của hàm.
* read\_file\_names (required): Danh sách các tệp dữ liệu thô để chuyển đổi. Tệp đầu tiên sẽ được coi là tệp huấn luyện và sẽ được sử dụng để tạo tệp bản đồ nhãn khi được yêu cầu.
* read\_dir (required): Thư mục chứa các tập tin đầu vào.
* save\_dir (required): Thư mục lưu các tệp bản đồ tsv/nhãn đã chuyển đổi.

1. **Hàm mất mát:**

Hàm CrossEntropyLoss: Hàm này được được định nghĩa trong pytorch. Hàm mất mát này nên được sử dụng cho các nhiệm vụ phân loại câu đơn (single sentence) hoặc cặp câu (pair sentence).

Hàm SRLLoss: Bài toán mà khóa luận quan tâm là tác vụ SRL, do đó phải phần một hàm mất mát cho các nhiệm vụ liên quan tới SRL (sequence labeling tasks). Do đó, Hàm SRLLoss này là một phiên bản sửa đổi của mất mát CrossEntropy cho các nhiệm vụ ghi nhãn SRL/trình tự. Mất mát này bỏ qua các giá trị 'O' bổ sung thông qua attention masks.

1. **Metrics đánh giá:**

Để đánh giá mô hình trên tập Dev và tập Test trong suốt quá trình huấn luyện, Khóa luận sử dụng một số metrics đánh giá sau:

* seqeval\_f1\_score (yTrue, yPred)
* seqeval\_precision (yTrue, yPred)
* seqeval\_recall (yTrue, yPred)

Ba hàm này là các hàm chuyên biệt để tính điểm F1, Precision, Recall cho các tác vụ ghi nhãn SRL hay các tác vụ có dạng sequence labelling tasks được lấy từ thư viện seqeval.

Parameters trong các hàm: Ví dụ đối với tác vụ SRL

yTrue (list of list) – [[‘O’, ‘O’, ‘O’, ‘B-A1’, ‘I-A1’, ‘I-A1’, ‘O’], [‘B-A1’, ‘I-A1’, ‘O’]]

yPred (list of list) – [[‘O’, ‘O’, ‘B-A1’, ‘I-A1’, ‘B-A1, ‘O’, ‘O’], [‘B-A1’, ‘I-A1’, ‘O’]]

1. **Task file**

Task file có cấu trúc như sau:

TaskA/(B, C):

model\_type: BERT

config\_name: bert-base-uncased

dropout\_prob: 0.05/(0.3, 0.05)

label\_map\_or\_file:

-label1/(seq\_f1)

-label2/(seq\_precision)

-label3/(seq\_recall)

metrics:

- accuracy

loss\_type: CrossEntropyLoss/(NERLoss, CrossEntropyLoss)

task\_type: SingleSenClassification/(NER, SentencePairClassification)

file\_names:

- taskA\_train.tsv/(taskB\_train.tsv, taskC\_train.tsv)

- taskA\_dev.tsv/(taskB\_dev.tsv, taskC\_dev.tsv)

- taskA\_test.tsv/(taskB\_test.tsv, taskC\_test.tsv)

Trong đó:

* task\_type (required): Định dạng của tác vụ như được mô tả trong Task type
* file\_names (required): Danh sách tên tệp tsv dữ liệu cần thiết cho tác vụ. Tệp đầu tiên được coi là tệp để huấn luyện (train), tệp thứ hai là tệp dev và tệp thứ ba là tệp test.
* model\_type (required): Loại shared-encoder dùng chung sẽ sử dụng. Model\_type cho tất cả các tác vụ được đề cập trong tệp phải giống nhau. Chúng ta có thể tham khảo Model type để chọn loại model.
* config\_name (optional): Cấu hình của mô hình bộ mã hóa. Chúng ta có thể tham khảo Loại mô hình để chọn cấu hình loại mô hình. Trong trường hợp không có tham số này, cấu hình mặc định sẽ được sử dụng.
* class\_num (bắt buộc/tùy chọn): Số lớp hiện có để phân loại. Tham số này là tùy chọn nếu nhãn\_map\_or\_file được cung cấp, nếu không thì bắt buộc.
* label\_map\_or\_file (bắt buộc/tùy chọn):
* Trong trường hợp nhãn là chuỗi, đây là danh sách các nhãn duy nhất.
* Chúng ta cũng có thể cung cấp tệp bản đồ từ điển kết xuất joblib như {‘nhãn1’:0, ‘nhãn2’:1, ...}.
* Nếu chúng ta đang sử dụng Chuyển đổi dữ liệu để tạo các tệp dữ liệu, thì đường dẫn đến tệp label\_map được tạo cùng với các tệp đã chuyển đổi sẽ được cung cấp tại đây.
* loss\_type (required): Hàm mất mát cho việc huấn luyện được xác định trong Losses.
* dropout\_prob (optional): Thông số drop out
* metrics (optional): Danh sách metrics sẽ sử dụng trong quá trình đánh giá tác vụ
* loss\_weight (tùy chọn): Giá trị loss weight (từ 0 đến 1) cho từng tác vụ.

1. **Training**

Một số thông số cần thiết lập khi huấn luyện mô hình:

* data\_dir (required): - Đường dẫn đến thư mục lưu trữ dữ liệu đã chuẩn bị. (ví dụ: bert\_base\_uncased\_prepared\_data)
* task\_file (required): - Đường dẫn đến tệp tác vụ để đào tạo.
* out\_dir (required): - Đường dẫn lưu các điểm kiểm tra mô hình đa tác vụ.
* epochs (required): - Số epochs để huấn luyện.
* train\_batch\_size (optional, default 8):- Kích thước batch size để huấn luyện.
* eval\_batch\_size (optional, default 32): - Kích thước batch size để đánh giá.
* grad\_accumulation\_steps (optional, default 1): - Số lô cần tích lũy trước khi cập nhật.
* log\_per\_updates (optional, default 10): - Số lần cập nhật sau đó để ghi nhật ký bị mất.
* silent (optional, default True): - Đặt thành False để nhật ký cũng được hiển thị trên đầu ra của thiết bị đầu cuối.
* max\_seq\_len (optional, default 128): - Độ dài chuỗi tối đa được sử dụng trong quá trình chuẩn bị dữ liệu.
* save\_per\_updates (optional, default 0): - Số bước cập nhật sau điểm kiểm tra mô hình sẽ được lưu. Mô hình luôn được lưu vào cuối mỗi epochs.
* load\_saved\_model (optional, default None): - Đường dẫn đến mô hình đã lưu trong trường hợp tải.
* resume\_train (optional, default False): - Đặt thành True để tiếp tục đào tạo từ mô hình đã lưu. Đào tạo sẽ tiếp tục từ bước lưu mô hình đã tải.