ĐẠI HỌC QUỐC GIA HÀ NỘI
TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN

HÀ MỸ LINH

NGHIÊN CỨU CÁC PHƯƠNG PHÁP BIỂU DIỄN VÀ PHÁT TRIỂN NGỮ LIỆU, CÔNG CỤ CHO PHÂN TÍCH CÚ PHÁP VÀ NGỮ NGHĨA TIẾNG VIỆT

LUẬN ÁN TIẾN SĨ TOÁN TIN

HÀ NỘI – 2025

ĐẠI HỌC QUỐC GIA HÀ NỘI
TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN

HÀ MỸ LINH

NGHIÊN CỨU CÁC PHƯƠNG PHÁP BIỂU DIỄN VÀ PHÁT TRIỂN NGỮ LIỆU, CÔNG CỤ CHO PHÂN TÍCH CÚ PHÁP VÀ NGỮ NGHĨA TIẾNG VIỆT

Chuyên ngành: Cơ sở toán học cho tin học
Mã số: 9460117.02

LUẬN ÁN TIẾN SĨ TOÁN TIN

NGƯỜI HƯỚNG DẪN KHOA HỌC:
1. GS.TS. NGUYỄN LÊ MINH
2. TS. NGUYỄN THỊ MINH HUYỀN

HÀ NỘI – 2025

LỜI CAM ĐOAN

Tôi xin cam đoan các kết quả trình bày trong luận án là công trình nghiên cứu của bản thân nghiên cứu sinh trong thời gian học tập và nghiên cứu tại Trường Đại học Khoa học Tự nhiên, Đại học Quốc gia Hà Nội, dưới sự hướng dẫn của tập thể hướng dẫn khoa học. Các số liệu, kết quả trình bày trong luận án là hoàn toàn trung thực. Các kết quả sử dụng tham khảo đều đã được trích dẫn đầy đủ và theo đúng quy định.

Hà Nội, ngày tháng năm 2025

Nghiên cứu sinh

Hà Mỹ Linh

LỜI CẢM ƠN

Trong quá trình thực hiện luận án, tôi xin gửi lời cảm ơn sâu sắc nhất tới thầy cô hướng dẫn của mình là **GS.TS Nguyễn Lê Minh** và **TS Nguyễn Thị Minh Huyền**. Thầy cô luôn nhiệt tình chỉ dạy, định hướng, chia sẻ và động viên tôi rất nhiều. Tôi luôn cảm thấy trân trọng, biết ơn và ghi nhớ thời gian làm việc dưới sự hướng dẫn của thầy cô.

Tôi xin cảm ơn tới các thầy cô trong Khoa Toán - Cơ - Tin học, Phòng Đào tạo, đặc biệt là Bộ môn Tin học và Phòng Thí nghiệm Khoa học Dữ liệu - nơi tôi làm việc. Các thầy cô đã dạy dỗ cho tôi những kiến thức nền tảng trong nghiên cứu, các đồng nghiệp đã luôn động viên, chia sẻ và cùng nhau làm việc, góp ý để đạt được những mục tiêu nghiên cứu.

Tôi xin trân trọng cảm ơn **Quỹ học bổng đổi mới sáng tạo VinIF**. Nhờ có học bổng của quỹ, tôi đã yên tâm làm việc và nghiên cứu. Quỹ không chỉ hỗ trợ tôi về điều kiện vật chất mà còn là nguồn động lực to lớn để tôi thực hiện nhiệm vụ nghiên cứu của mình.

Cuối cùng, tôi xin bày tỏ lòng cảm ơn chân thành tới gia đình tôi đã luôn bên cạnh ủng hộ, yêu thương và động viên tôi trong suốt quá trình nghiên cứu. Gia đình là nơi đã giúp tôi vượt qua những giai đoạn khó khăn nhất để hoàn thành chặng đường nghiên cứu này.

Hà Nội, ngày tháng năm 2025

Nghiên cứu sinh

Hà Mỹ Linh

MỤC LỤC

DANH MỤC THUẬT NGỮ VÀ TỪ VIẾT TẮT

Từ viết tắt	**Tiếng Anh**	**Ý nghĩa**
AMR	Abstract Meaning Representation	Mô hình biểu diễn ngữ nghĩa trừu tượng
biLM	Bidirectional Language Model	Mô hình ngôn ngữ hai chiều
BERT	Bidirectional Encoder Representations from Transformers	Mô hình biểu diễn mã hoá hai chiều từ Transformers
CBOW	Continuous Bag of Words	Mô hình túi từ liên tục
DCS	Dependency based Compositional Semantics	Mô hình ngữ nghĩa thành phần dựa vào phụ thuộc
DRT	Discourse Representation Theory	Lý thuyết biểu diễn diễn ngôn
ELMo	Embedding from Language Model	Mô hình nhúng của ngôn ngữ
GMB	Groningen Meaning Bank	Kho ngữ liệu ngữ nghĩa Groningen
GloVe	Global Vectors for Word Representation	Mô hình biểu diễn véc tơ từ toàn cục
GPT	Generative Pretrained Transformer	Mô hình chuyển đổi được huấn luyện trước tạo sinh
LAS	Labeled Attachment Score	Độ đo đính kèm nhãn phụ thuộc
LIRICS	Linguistic InfRastructure for Interoperable ResourCes and Systems	Cơ sở ngôn ngữ cho các hệ thống và tài nguyên có thể tương tác
LTAG	Lexicalized Tree Adjoining Grammars	Văn phạm kết nối cây từ vựng hóa.
LLMs	Large Language Models	Mô hình ngôn ngữ lớn
LMF	Lexical Markup Framework	Khung đánh dấu từ vựng
NLP	Natural Language Processing	Xử lí ngôn ngữ tự nhiên
NLU	Natural Language Understanding	Hiểu ngôn ngữ tự nhiên
NSP	Next Sentence Prediction	Mô hình dự đoán câu tiếp theo
MLM	Masked Language Model	Mô hình ngôn ngữ có mặt nạ
POS	Part-of-Speech	Nhãn từ loại
SRL	Semantic Role Labeling	Gán nhãn vai nghĩa.
UCCA	Universal Conceptual Cognitive Annotation	Mô hình chú thích nhận thức khái niệm phổ quát
UAS	Unlabeled Attachment Score	Độ đo không đính kèm nhãn phụ thuộc
UD	Universal Dependency	Phụ thuộc phổ quát
U-POS	Universal Part-of-Speech	Nhãn từ loại phổ quát
VCL	Vietnamese Computational Lexicon	Từ điển tiếng Việt dùng cho máy tính

DANH MỤC HÌNH VẼ

DANH MỤC BẢNG

MỞ ĐẦU

Xử lí ngôn ngữ tự nhiên (*Natural Language Processing - NLP*) đã thu hút nhiều sự quan tâm của các nhóm nghiên cứu trên thế giới ngay từ khi máy tính điện tử ra đời. Mục tiêu của xử lí ngôn ngữ tự nhiên là nghiên cứu xây dựng các thuật toán và chương trình máy tính có khả năng xử lí, phân tích và tổng hợp được ngôn ngữ tự nhiên dưới dạng tiếng nói (*speech*) hoặc văn bản (*text*), nhằm nâng cao khả năng tương tác giữa máy tính với con người. Trong khuôn khổ luận án này, chúng ta sẽ bàn về các vấn đề phân tích ngữ pháp và ngữ nghĩa, hướng tới mục tiêu giúp máy tính hiểu ngôn ngữ tự nhiên.

Ngữ pháp là tập các quy tắc xác định cấu trúc câu dựa vào các từ, cụm từ và các chức năng, đảm bảo rằng thông tin được truyền đạt rõ ràng, mạch lạc. Ngữ nghĩa tập trung vào nghĩa của các từ và quan hệ giữa chúng trong văn cảnh cụ thể, từ đó xác định nghĩa của các đơn vị lớn hơn như cụm từ, câu hay văn bản. Việc hiểu nghĩa của một câu trong văn bản phụ thuộc vào cách xác định cấu trúc câu, kết quả của quá trình phân tích cú pháp. Phân tích ngữ nghĩa cần trả về một biểu diễn ngữ nghĩa một mặt cho phép trích xuất các thông tin về các thực thể, vai trò và quan hệ giữa các thực thể đó, cùng các thông tin ngữ cảnh như không gian hay thời gian, mặt khác cũng cho phép suy luận nhằm phát hiện ra các mối quan hệ khác.

Nhiều phương pháp đã được phát triển để giải quyết các bài toán phân tích cú pháp và ngữ nghĩa, từ các cách tiếp cận dựa trên luật, cho đến các kỹ thuật học máy, đặc biệt là các mô hình hiện đại sử dụng học sâu và gần đây nhất là sự phát triển của các mô hình ngôn ngữ lớn.

Phương pháp dựa vào luật thiết kế các luật để phân tích cú pháp và ngữ nghĩa, dựa trên các quy tắc ngôn ngữ học. Các phương pháp này có ưu điểm về tính minh bạch, dễ giải thích, nhất quán và hiệu quả với các ngữ cảnh hẹp với dữ liệu hạn chế. Tuy nhiên, chúng thường gặp phải hạn chế về khả năng mở rộng, thiếu tính linh hoạt khi xử lý ngữ nghĩa mơ hồ hoặc ngữ cảnh, và không thể tự học từ dữ liệu mới, do vậy khó áp dụng cho các cấu trúc phức tạp của ngôn ngữ. Trong khi đó, các kỹ thuật học máy và học sâu sử dụng các kho dữ liệu có gán nhãn để học các mẫu ngôn ngữ một cách tự động, mở ra khả năng xử lý ngữ cảnh và ngữ nghĩa hướng dữ liệu sát thực hơn. Gần đây, sự phát triển

của các mô hình máy biến đổi (*Transformer* [12]) như BERT [50] và các mô hình ngôn ngữ lớn (*Large Language Models - LLMs*) như GPT [97], Gemini [43] đã mang đến những bước đột phá quan trọng trong NLP. Những mô hình này được huấn luyện trên hàng tỷ từ, có khả năng nắm bắt ngữ cảnh và ngữ nghĩa từ các văn bản lớn, cho phép chúng tạo ra các câu trả lời tự nhiên và mạch lạc trong các tình huống giao tiếp đa dạng. Tuy nhiên, các mô hình này vẫn tồn tại những hạn chế đáng kể, bao gồm nhu cầu tài nguyên lớn trong quá trình huấn luyện và triển khai, cũng như khó khăn trong việc xử lý các ngữ cảnh phức tạp hoặc không rõ ràng [129, 69]. Ngoài ra, khả năng xử lý ngữ pháp và ngữ nghĩa của LLMs phụ thuộc rất nhiều vào cấu trúc và chất lượng dữ liệu huấn luyện. Bên cạnh đó, các mô hình ngôn ngữ lớn vẫn còn nhiều hạn chế trong khả năng suy luận (*reasoning*) và lập kế hoạch (*planning*) [47], cũng như phân tích các văn bản chuyên ngành như y khoa [25, 34] và pháp lý [56, 110].

Động lực nghiên cứu

Trong hầu hết các cách tiếp cận nhằm giải quyết bài toán phân tích cú pháp và biểu diễn ngữ nghĩa, việc xây dựng các tài nguyên từ vựng cũng như kho từ vựng hoặc kho ngữ liệu được chú giải ngôn ngữ đóng vai trò thiết yếu, có giá trị ứng dụng cao và tính ổn định lâu dài. Những tài nguyên này không chỉ là nền tảng quan trọng để huấn luyện và đánh giá hiệu quả của các mô hình xử lý ngôn ngữ tự nhiên, mà còn góp phần phục vụ các nghiên cứu mang tính lý thuyết trong lĩnh vực ngôn ngữ học.

Các tài nguyên từ vựng chứa các thông tin về hình thái từ, các khung cú pháp, ngữ nghĩa, các ràng buộc và mối quan hệ giữa các thành phần câu với từ vựng đó. Trong ngữ pháp và ngữ nghĩa, động từ đóng một vai trò quan trọng, thường truyền đạt ý chính của câu, và đòi hỏi có một biểu diễn chính xác, được xác định rõ ràng để có thể nắm bắt cả cấu trúc tham tố cũng như các ràng buộc về ngữ pháp, ngữ nghĩa của chúng [70]. Do vậy, ngữ liệu động từ đóng vai trò cốt yếu và phức tạp nhất trong kho từ vựng. Có thể kể đến một số kho từ vựng liên quan tới cú pháp và ngữ nghĩa nổi bật như mạng từ WordNet [86, 41] mô tả chi tiết các quan hệ ngữ nghĩa như đồng nghĩa/trái nghĩa, thượng danh/hạ danh của các từ; kho ngữ liệu khung vị từ FrameNet [13]; hay mạng động từ VerbNet [68] chứa các đặc tả cú pháp - ngữ nghĩa của các động từ được chia thành các lớp tương đương.

Thuật ngữ "ngân hàng cây" (*treebank*) chỉ các văn bản được chú giải thông tin từ loại, cú pháp chi tiết, tạo cơ sở cho các bài toán phân tích từ, cú pháp

và ngữ nghĩa. Nhờ có các chú giải này, hệ thống có thể học được mối quan hệ giữa các thành phần câu, chẳng hạn như chủ ngữ, vị ngữ và bổ ngữ, từ đó hỗ trợ phân tích ngữ nghĩa sâu hơn. Các nhóm nghiên cứu đã phát triển các ngân hàng cây cho ngôn ngữ của mình và phối hợp để hướng tới việc chuẩn hóa chúng. Một dự án nổi bật trong việc xây dựng các ngân hàng cây đa ngôn ngữ hướng chuẩn là Dự án Phụ thuộc Phổ quát (*Universal Dependencies - UD*), trong đó các nhóm nghiên cứu đã xây dựng hơn 200 ngân hàng cây cú pháp phụ thuộc cho hơn 150 ngôn ngữ trên toàn thế giới [90].

Về ngữ nghĩa, các mô hình biểu diễn và kho ngữ liệu có chú giải ngữ nghĩa cũng đã và đang được các nhóm nghiên cứu quan tâm và phát triển để có thể hình thức hóa nghĩa của từ, câu và đoạn văn. Các mô hình biểu diễn ngữ nghĩa giúp cho việc hiểu và diễn giải ngôn ngữ trong các ngữ cảnh khác nhau, giải quyết vấn đề nhập nhằng và mơ hồ ngữ nghĩa. Các kho ngữ liệu có chú giải ngữ nghĩa tiêu biểu gồm Propbank [67] với chú giải ở mức nông là vai nghĩa, AMR [14] với chú giải ở mức sâu theo mô hình biểu diễn ngữ nghĩa trừu tượng, hay các ngân hàng ngữ nghĩa khác như Groningen - GMB [58], UCCA [7], UMR [118], ...

Một yếu tố quan trọng được nhấn mạnh trong quá trình thiết kế và xây dựng tài nguyên ngôn ngữ là vấn đề chuẩn hóa. Việc này đảm bảo khả năng tương thích giữa các công cụ và các tập dữ liệu, cho phép tích hợp dữ liệu dễ dàng. Các mô hình chú giải chuẩn hóa cũng đóng vai trò quan trọng đối với tính nhất quán và chất lượng dữ liệu, tăng tính tin cậy trong việc chú giải ngữ liệu. Quy trình chuẩn hóa cũng giúp giảm chi phí nhờ khả năng tái sử dụng các tài nguyên hiện có, đẩy nhanh quá trình phát triển các kho ngữ liệu. Ngoài ra, nó còn hỗ trợ các ứng dụng đa ngôn ngữ và xuyên ngôn ngữ, thúc đẩy khả năng tích hợp nhiều tập ngữ liệu trong các ứng dụng công nghệ ngôn ngữ. Các tài nguyên được chuẩn hóa cũng sẽ dễ dàng được cập nhật và bảo trì, đảm bảo chúng luôn phù hợp khi ngôn ngữ và công nghệ phát triển. Nhiều nỗ lực chuẩn hóa đã được thực hiện, chẳng hạn như các tiêu chuẩn quốc tế về xây dựng và quản lý dữ liệu ngôn ngữ, như chuẩn TEI[1] và các dự án chuẩn khác nhau trong khuôn khổ ISO TC 37/SC 4[2] về chuẩn hóa quản lý tài nguyên ngôn ngữ.

Tính cấp thiết của đề tài

Đối với tiếng Việt, việc phát triển kho từ vựng và các kho ngữ liệu có chú giải ngữ pháp, ngữ nghĩa cũng đã được quan tâm từ nhiều năm trước. Tài nguyên từ

[1] `https://tei-c.org/release/doc/tei-p5-doc/en/html/CC.html`
[2] `https://www.iso.org/committee/297592.html`

vựng đầu tiên được xây dựng từ năm 2006 là Từ điển tiếng Việt cho máy tính [94] (*Vietnamese Computational Lexicon* – VCL), được xây dựng theo chuẩn LMF (*Lexical Markup Framework* [45]), gồm có 42,000 mục từ với các thông tin về hình thái học, cú pháp học và ngữ nghĩa học. Tiếp đến, kho từ vựng WordNet tiếng Việt [1] được xây dựng từ năm 2017 với hơn 78,000 tập đồng nghĩa (*synset*) và 80,413 mối quan hệ ngữ nghĩa. Ngân hàng cây cho tiếng Việt đầu tiên được xây dựng từ năm 2009 là Viettreebank [101], với thông tin về từ loại và cú pháp thành phần cho 10,165 câu tiếng Việt. Sau đó, dựa vào kho ngữ liệu này, các nhóm nghiên cứu đã xây dựng các ngân hàng cây cú pháp phụ thuộc với những tập nhãn phụ thuộc riêng của từng nhóm [92, 65, 30]. Đối với bài toán phân tích ngữ nghĩa, một tập nhãn vai nghĩa cùng kho ngữ liệu gồm 5,640 câu đã được xây dựng [2].

Các nghiên cứu về ngữ pháp và ngữ nghĩa tiếng Việt trong thời gian qua đã đạt được nhiều thành tựu đáng kể. Tuy nhiên, vẫn còn tồn tại những thách thức lớn cần được giải quyết. Mặc dù khối lượng văn bản tiếng Việt hiện diện trên Internet là không nhỏ, tiếng Việt vẫn được xếp vào nhóm ngôn ngữ nghèo tài nguyên, do thiếu hụt các tài liệu chuyên ngành cùng với các kho ngữ liệu có chú giải ngữ pháp và ngữ nghĩa một cách hệ thống. Do đó, việc xây dựng các kho ngữ liệu tiếng Việt có chú giải từ vựng, cú pháp và ngữ nghĩa ở mức độ sâu là một nhiệm vụ quan trọng, vừa cấp thiết vừa có ý nghĩa nền tảng đối với việc phát triển các ứng dụng NLP cho tiếng Việt. Đây cũng chính là một trong những mục tiêu trọng tâm mà luận án hướng tới.

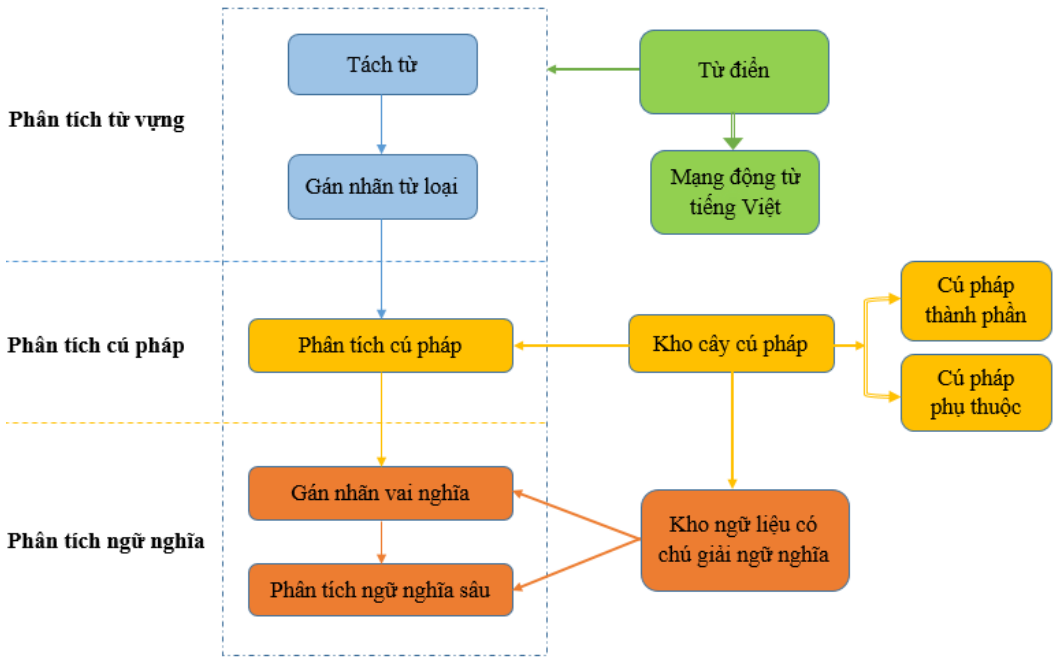
Cụ thể, về ngữ pháp, một trong những vấn đề đáng chú ý là sự thiếu thống nhất và chuẩn hóa trong việc xây dựng các bộ nhãn ngữ pháp, bao gồm cả nhãn thành phần và nhãn phụ thuộc. Hiện nay, các nhóm nghiên cứu thường tự phát triển bộ nhãn riêng mà không có sự phối hợp hoặc tuân theo một số chuẩn chung, dẫn đến khó khăn trong việc so sánh, đánh giá, và sử dụng chéo giữa các nghiên cứu. Bên cạnh đó, chưa có các tiêu chuẩn đánh giá rõ ràng để đo lường hiệu quả và mức độ phù hợp của các bộ nhãn này đối với tiếng Việt. Ngoài ra, nhiều nghiên cứu còn thiếu thông tin chi tiết về quy trình xây dựng bộ nhãn và gán nhãn dữ liệu, chưa phân tích kĩ các nhãn được sử dụng cũng như các trường hợp đặc trưng đối với tiếng Việt, làm giảm khả năng tái hiện và cải thiện kết quả. Về ngữ nghĩa, hiện tại, chưa có các mô hình biểu diễn ngữ nghĩa và kho ngữ liệu được gán nhãn ngữ nghĩa đầy đủ, toàn diện và được chuẩn hóa cho tiếng Việt. Dù đã có các kho ngữ liệu gán nhãn vai nghĩa, nhưng các kho này vẫn chưa được liên kết chặt chẽ với nhau hoặc với các tài nguyên ngôn ngữ khác, hiệu quả của các mô hình gán nhãn vai nghĩa cũng còn khá hạn chế.

Mục tiêu nghiên cứu

Từ những nhận định trên, luận án đặt mục tiêu nghiên cứu phát triển ngữ liệu cùng các sơ đồ chú giải, bao gồm kho từ vựng cũng như các kho ngữ liệu có chú giải cú pháp, ngữ nghĩa, tuân theo các mô hình chuẩn hóa tài nguyên ngôn ngữ trên thế giới. Song song với việc xây dựng ngữ liệu, luận án cũng đánh giá, phát triển các công cụ phân tích cú pháp và ngữ nghĩa tiếng Việt, hỗ trợ qua lại công việc xây dựng ngữ liệu. Cụ thể, mục tiêu chi tiết của luận án tập trung vào:

- Xây dựng ngữ liệu: Thiết kế lược đồ chú giải ngữ pháp/ngữ nghĩa, xây dựng kho ngữ liệu được chú giải cú pháp (cú pháp thành phần, cú pháp phụ thuộc) và ngữ nghĩa. Các tài nguyên này được thiết kế dựa trên các mô hình chuẩn hóa quốc tế, bảo đảm tính nhất quán, khả năng mở rộng và tính tương thích với các hệ thống xử lý đa ngôn ngữ, đồng thời thiết kế và xây dựng mạng động từ tiếng Việt (viVerbNet).
- Phát triển công cụ: Nghiên cứu, phát triển, tinh chỉnh và đánh giá các mô hình phân tích cú pháp và ngữ nghĩa cho tiếng Việt, nhằm vừa hỗ trợ quá trình gán nhãn ngữ liệu, vừa tận dụng chính các ngữ liệu này để cải thiện hiệu suất của các mô hình phân tích ngữ pháp/ngữ nghĩa tiếng Việt.

Những mục tiêu này được mô tả cụ thể trong Hình 1.



Hình 1: Mục tiêu của luận án.

Phạm vi nghiên cứu

Để đạt được các mục tiêu trên, luận án sẽ giải quyết các bài toán sau:

- Phân tích cú pháp: Xây dựng tập nhãn cú pháp, kho ngữ liệu và phát triển các công cụ phân tích cú pháp thành phần, cú pháp phụ thuộc.
- Phân tích ngữ nghĩa câu: Xây dựng tập nhãn vai nghĩa, kho ngữ liệu, xây dựng mô hình biểu diễn ngữ nghĩa cho văn bản tiếng Việt, thử nghiệm một số mô hình phân tích ngữ nghĩa cho tiếng Việt.
- Phân tích ngữ nghĩa từ vựng: Nghiên cứu và thiết kế, xây dựng mạng động từ (viVerbnet) cho tiếng Việt.

Phương pháp nghiên cứu

Luận án áp dụng phương pháp nghiên cứu kết hợp giữa lý thuyết, thực nghiệm và định lượng nhằm đảm bảo tính toàn diện và khách quan trong quá trình xây dựng tài nguyên ngôn ngữ cũng như phát triển các mô hình phân tích cú pháp và ngữ nghĩa.

- *Phương pháp lý thuyết*: luận án khảo sát, phân tích tài liệu ngôn ngữ học và các sơ đồ chú giải sẵn có để xây dựng các sơ đồ chú giải và hệ thống hóa kho từ vựng, ngữ liệu cú pháp, ngữ nghĩa tiếng Việt, đảm bảo tính tương thích với các kho ngữ liệu cho các ngôn ngữ khác và phù hợp với đặc thù tiếng Việt.
- *Phương pháp thực nghiệm*: được triển khai qua việc xây dựng, thử nghiệm và đánh giá các mô hình phân tích cú pháp, ngữ nghĩa văn bản và mô hình phân cụm cho mạng động từ.
- *Phương pháp định lượng*: được sử dụng để phân tích thống kê dữ liệu, đo lường và đánh giá hiệu quả của các mô hình phân tích cú pháp, ngữ nghĩa và chất lượng kho ngữ liệu, đảm bảo tính tin cậy và khả năng ứng dụng trong các bài toán xử lý ngôn ngữ tự nhiên.

Đóng góp của luận án

Luận án đã có những đóng góp cơ bản về hai hướng chính:

- Xây dựng các lược đồ chú giải và kho ngữ liệu:

– Cú pháp phụ thuộc: Luận án đã xây dựng lại tập nhãn cú pháp phụ thuộc theo những cập nhật, sửa đổi dựa vào phiên bản 2.0 của Dự án cú pháp phụ thuộc phổ quát UD[3], xây dựng kho ngữ liệu gồm hơn 9,000 câu (trong đó 3,000 câu đã được tích hợp vào UD, vào tháng 11 năm 2022[4]). Kho ngữ liệu cú pháp phụ thuộc này sử dụng trong hội thảo về Xử lí ngôn ngữ tự nhiên và tiếng nói tiếng Việt (VLSP 2020[5]) ([P2], [P10]).

– Cú pháp thành phần: Dựa vào kho ngữ liệu cú pháp thành phần Viet-treebank đã có, luận án thực hiện việc rà soát, cập nhật và chuẩn hoá các nhãn cú pháp thành phần và tài liệu hướng dẫn gán nhãn để phù hợp với các nghiên cứu đối sánh đa ngữ. Kho ngữ liệu gồm hơn 9,000 câu đã được xây dựng lại và sử dụng trong hội thảo về Xử lí ngôn ngữ tự nhiên và tiếng nói tiếng Việt (VLSP 2022 và VLSP 2023[6]) ([P8]).

– Ngữ nghĩa nông (vai nghĩa): Luận án đã xây dựng kho ngữ liệu gán nhãn vai nghĩa cho tiếng Việt (gồm 2,570 câu) theo tiêu chuẩn vai nghĩa đa ngữ, kết hợp với dự án xây dựng Propbank 2.0[7] ([P5], [P7]).

– Ngữ nghĩa sâu: Luận án đã xây dựng mô hình và hướng dẫn gán nhãn ngữ nghĩa cho tiếng Việt dựa vào mô hình ngữ nghĩa trừu tượng của tiếng Anh (AMR) và các vai nghĩa LIRICS [98] - được thiết kế hướng chuẩn ISO. Kho ngữ liệu tiếng Việt gồm có 1,570 câu đã được xây dựng ([P1], [P4], [P6], [P9]).

– Mạng động từ tiếng Việt: Xây dựng lược đồ chú giải mạng động từ cho tiếng Việt trên cơ sở tham chiếu VerbNet tiếng Anh, với 5 thành phần chính: vai nghĩa, ràng buộc lựa chọn, khung cú pháp, ràng buộc cú pháp và vị từ ngữ nghĩa. Sau đó, mạng động từ tiếng Việt (viVerbNet) gồm 100 cụm động từ được phát triển và gán nhãn theo lược đồ đã đề xuất ([P3]).

• Về phương pháp và mô hình cho phân tích tiếng Việt:

– Luận án đã thử nghiệm, đánh giá và so sánh một số mô hình với các biểu diễn véc-tơ từ khác nhau để cải tiến hiệu quả của bài toán phân tích cú pháp phụ thuộc. Bên cạnh đó, thực hiện khảo sát các phương

[3] https://universaldependencies.org/
[4] https://github.com/UniversalDependencies/UD_Vietnamese-VTB/tree/master
[5] https://vlsp.org.vn/vlsp2020/eval/udp
[6] https://vlsp.org.vn/vlsp2023/eval/vcp
[7] https://universalpropositions.github.io/

pháp phân tích cú pháp thành phần, đưa ra một số thảo luận về kết quả của các phương pháp đã đạt được.

- Xây dựng công cụ chuyển đổi giữa cú pháp thành phần và cú pháp phụ thuộc, hỗ trợ quá trình gán nhãn dữ liệu.
- Phát triển và đánh giá các thuật toán phân cụm động từ tiếng Việt.
- Thử nghiệm các mô hình ngôn ngữ lớn để gán nhãn vai nghĩa và phân tích ngữ nghĩa cho văn bản tiếng Việt, đánh giá và phân tích kết quả đạt được.

Cấu trúc của luận án

Luận án được tổ chức như sau:

- Chương 1: Trình bày các kiến thức cơ sở. Trong đó, các khái niệm, các phương pháp phân tích cú pháp và ngữ nghĩa sẽ được mô tả chi tiết. Sau đó, chương này trình bày về phương pháp luận xây dựng kho ngữ liệu và các tài nguyên ngôn ngữ hiện có.

- Chương 2: Mô tả chi tiết về việc xây dựng tài nguyên và công cụ phân tích cú pháp tiếng Việt. Trong đó, trình bày cụ thể phương pháp xây dựng tập nhãn cú pháp thành phần, cú pháp phụ thuộc. Đồng thời, khảo sát và thử nghiệm phương pháp phân tích cú pháp mới và phân tích kết quả đạt được cho tiếng Việt. Cuối cùng, xây dựng thuật toán chuyển kho ngữ liệu cú pháp thành phần sang kho ngữ liệu gán nhãn phụ thuộc và ngược lại.

- Chương 3: Xây dựng tài nguyên và công cụ phân tích ngữ nghĩa tiếng Việt. Chương này trình bày về việc xây dựng kho ngữ liệu gán nhãn vai nghĩa cho tiếng Việt theo hướng tiếp cận liên ngữ, xây dựng mô hình biểu diễn ngữ nghĩa cho tiếng Việt, xây dựng công cụ hỗ trợ gán nhãn ngữ nghĩa, thực nghiệm với một số mô hình ngôn ngữ lớn sinh phân tích ngữ nghĩa cho tiếng Việt và đánh giá kết quả đạt được.

- Chương 4: Trình bày về việc xây dựng mạng động từ tiếng Việt (viVerbNet): mô tả phương pháp xây dựng các cụm động từ dựa vào các thuật toán phân cụm, xây dựng các thành phần của một cụm động từ như vai nghĩa, ràng buộc lựa chọn, khung cú pháp và ràng buộc cú pháp, vị từ ngữ nghĩa. Sau đó, luận án xây dựng công cụ hỗ trợ gán nhãn viVerbNet cho tiếng Việt.

- Phần kết luận: Tóm tắt một số kết quả đạt được và hướng phát triển trong tương lai.

Chương 1
KIẾN THỨC CƠ SỞ

Chương này sẽ trình bày các kiến thức cơ bản liên quan đến cú pháp và ngữ nghĩa bao gồm các khái niệm về cú pháp thành phần và cú pháp phụ thuộc, các loại thông tin ngữ nghĩa, các mô hình và phương pháp phân tích cú pháp, ngữ nghĩa. Sau đó, luận án sẽ trình bày chi tiết về các mô hình ngôn ngữ, bao gồm các kỹ thuật truyền thống tới hiện đại. Cuối cùng, chương này sẽ mô tả các vấn đề liên quan đến việc xây dựng kho ngữ liệu, một yếu tố quan trọng trong nghiên cứu và ứng dụng ngôn ngữ học tính toán, bao gồm các phương pháp thu thập, chú giải, và quản lý dữ liệu ngôn ngữ. Bên cạnh đó, luận án sẽ giới thiệu một số tài nguyên ngữ nghĩa hiện có, giúp hỗ trợ việc nghiên cứu và phát triển các ứng dụng xử lý ngôn ngữ tự nhiên.

1.1. Một số vấn đề cơ bản về cú pháp và ngữ nghĩa

Trong ngôn ngữ học, cú pháp và ngữ nghĩa là hai vấn đề có mối quan hệ chặt chẽ và bổ trợ lẫn nhau. Cú pháp thể hiện những quy tắc chi phối các từ kết hợp với nhau để tạo nên câu. Trong tiếng Việt, chúng ta thấy rằng các từ thường được sắp xếp theo một trật tự riêng. Theo đó, chúng ta không thể tùy tiện sắp xếp các từ theo ý muốn chủ quan của mình. Vì khi thay đổi trật tự từ sẽ làm thay đổi chức năng cú pháp và cách giải thích nghĩa của các từ đó, chẳng hạn: câu "Tôi không được ăn thịt" và "Tôi không ăn được thịt" đã thay đổi ý nghĩa của "được" động từ tình thái (được ăn) thành "được" phụ từ (ăn được), theo đó ý nghĩa thông báo của hai câu cũng khác nhau.

Tương tự, mỗi thành phần trong câu có thể đóng một vai trò ngữ nghĩa khác nhau, nếu thay đổi các thành phần câu thì ý nghĩa của câu sẽ thay đổi. Ví dụ: "Tôi nhìn em" và "Em nhìn tôi" thể hiện chủ thể (*agent*) và đối tượng (*goal*) của hành động "nhìn" đã thay đổi vị trí cho nhau.

Sự tương tác giữa cú pháp và ngữ nghĩa là thiết yếu trong việc diễn giải và phân tích câu, và điều này rất quan trọng trong nghiên cứu ngôn ngữ học ứng dụng. Phần tiếp theo của luận án sẽ trình bày các khái niệm cơ bản về cú pháp và ngữ nghĩa.

1.1.1. Cú pháp

Trong ngôn ngữ học, việc xây dựng câu liên quan đến việc kết hợp các từ thành các cụm từ, và các cụm từ này lại kết hợp với nhau để tạo thành câu. Khi các từ và cụm từ ghép lại, chúng sẽ có vai trò chính và phụ, trong đó một từ hoặc cụm từ đóng vai trò chính, còn những từ hoặc cụm từ khác hỗ trợ vai trò chính này. Cú pháp tập trung vào nghiên cứu cấu trúc của câu, tức là nghiên cứu cách thức các từ, cụm từ, mệnh đề (*clause*) kết hợp với nhau để tạo nên câu, hoặc nghiên cứu mối quan hệ lẫn nhau giữa các yếu tố có mặt trong câu.

Cú pháp có thể được phân chia thành hai loại chính: cú pháp phụ thuộc và cú pháp thành phần. Trong đó, cú pháp thành phần tập trung vào cấu tạo "ngữ", nghiên cứu các cụm từ trong câu. Trong mỗi cụm từ, sẽ có một từ mang ý nghĩa chính và là trung tâm của cụm từ đó. Ví dụ, trong cụm danh từ sẽ có một danh từ làm trung tâm. Các cụm từ cũng được bổ sung thêm một số thông tin về chức năng của các cụm từ như chủ ngữ, vị ngữ, bổ ngữ, trạng ngữ, ..., cũng như các ý niệm về hiểu biết, nhận thức như không gian, thời gian, điều kiện, nguyên nhân, lý do, kết quả, mục đích, tình thái, ... Cú pháp phụ thuộc quan tâm đến chức năng và mối quan hệ chính phụ giữa các từ trong câu. Mỗi loại cú pháp có một cách tiếp cận khác nhau trong việc mô tả cấu trúc của câu và cách các thành phần của câu kết nối với nhau. Phần tiếp theo của luận án sẽ trình bày chi tiết về hai loại cú pháp thành phần và cú pháp phụ thuộc.

1.1.1.1. Cú pháp thành phần

Phân tích cú pháp theo thành phần câu là phương pháp phân tích cấu trúc ngữ pháp của một câu dựa trên cấu trúc phân cấp của các thành phần câu xét về mặt chức năng cú pháp, do thực từ hoặc cụm từ đảm nhiệm, có các tên gọi như chủ ngữ, vị ngữ, định ngữ, bổ ngữ, trạng ngữ, giới ngữ, ... Mục tiêu chính của phương pháp này là xác định cách các từ trong câu kết hợp để tạo thành các cụm từ, cũng như cấu trúc phân cấp của các cụm từ trong câu đảm nhiệm vai trò chức năng gì. Phương pháp này được thể hiện qua các nội dung chủ yếu sau:

1. Cấu trúc cây cú pháp: vị trí và tôn ti của mỗi từ trong câu được biểu diễn thông qua sơ đồ các nút và nhánh, phản ánh mối quan hệ giữa chúng. Trong cấu trúc cây cú pháp, mỗi nhánh thể hiện mối quan hệ giữa các thành phần của câu, tương ứng với các cụm từ và được gắn nhãn để chỉ ra loại cụm từ

tương ứng (ví dụ: cụm danh từ, cụm động từ, ...). Mỗi từ trong câu cũng được gắn nhãn từ loại.

2. Tính chi phối: thể hiện mối quan hệ theo chiều dọc giữa các nút với nhau (quan hệ tôn ti), nút nằm trên kiểm soát nút nằm phía dưới. Theo đó, nếu A là nút mẹ của B thì A chi phối B.

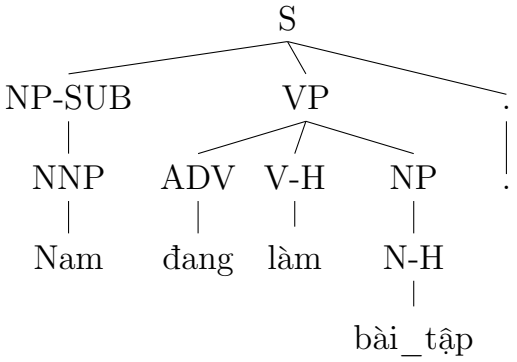
3. Các thành phần câu [4]: trong nghiên cứu tiếng Việt, hầu hết các nhà ngữ pháp học đều không đặt cho mình nhiệm vụ định nghĩa thành phần câu, chỉ tập trung miêu tả các thành phần câu cụ thể. Tuy nhiên, qua các giải pháp cụ thể đó, có thể thấy rằng quan niệm của từng người và vẫn còn những điểm chưa nhất trí trong giới nghiên cứu. Các tác giả Nguyễn Minh Thuyết và Nguyễn Văn Hiệp quan niệm rằng, thành phần câu là những từ tham gia nòng cốt câu (bắt buộc có mặt để đảm bảo tính trọn vẹn của câu) hoặc phụ thuộc trực tiếp vào nòng cốt câu. Những từ tham gia nòng cốt câu là thành phần chính của câu, gồm chủ ngữ, vị ngữ và các bổ ngữ bắt buộc của vị ngữ. Còn những từ ngữ phụ thuộc vào toàn bộ nòng cốt câu là thành phần phụ của câu. Trong số này, không có những thành tố chỉ có quan hệ với một từ trong câu, chúng chỉ là thành phần phụ của từ tổ. Ví dụ về các thành phần trong câu như:

    - Chủ ngữ: thường được tạo thành từ cụm danh từ (NP): "cái áo màu xanh", "một ngày" ..., đại từ: "tôi", "chúng ta" ..., cụm chủ-vị, ...
    - Vị ngữ: thường được tạo thành từ cụm động từ (VP) như "làm bài tập", "chơi đá bóng" ..., cụm tính từ (AP) như "xinh đẹp", "giàu tình cảm" ... hoặc thể từ, cụm chủ-vị, ...
    - Các thành phần phụ: như khởi ngữ, trạng ngữ, định ngữ. Ví dụ: cụm giới từ (PP): "ở nhà", "trên lớp học" hoặc các cụm từ khác như "còn gì", "như thế nào", ...

Thông thường, cú pháp thành phần của một câu được biểu diễn dưới dạng cây hoặc theo dạng đặt ngoặc. Cách biểu diễn này cho phép chúng ta có thể dễ dàng xây dựng và kết hợp các thành phần ngữ nghĩa dựa trên các thành phần cú pháp.

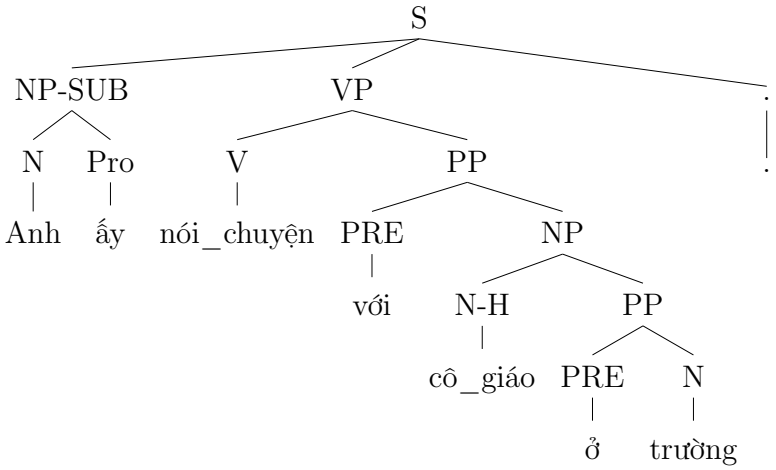
Ví dụ, với một câu tiếng Việt: `‘‘Nam đang làm bài_tập.’’` sẽ được phân tích cú pháp thành phần như trong Hình 1.1, và được biểu diễn dưới dạng đặt ngoặc là: `(S (NP-SUB (NNP Nam)) (VP (ADV đang) (V-H làm) (NP (N bài_tập)))`

(. .)). Trong đó, một số cụm từ xuất hiện trong câu như: Nam (cụm danh từ), đang làm bài tập (cụm động từ). Các nhãn NNP, N, ADV, V là các nhãn từ loại, NP, VP là các nhãn cụm từ và S là nhãn mệnh đề.



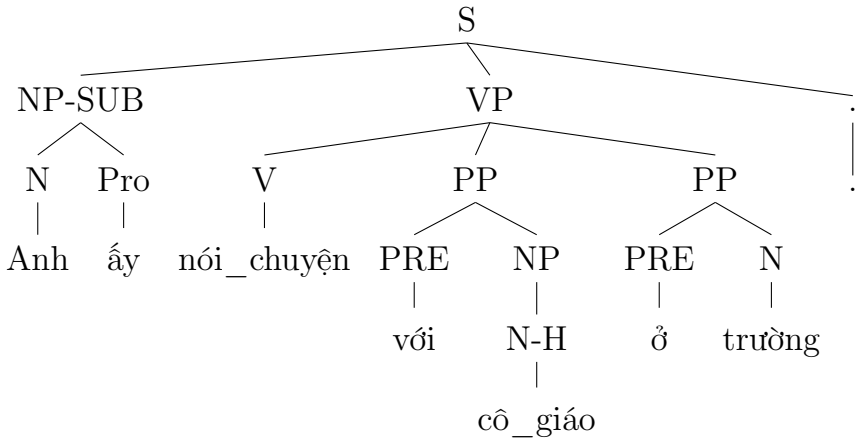
Hình 1.1: Cây cú pháp thành phần của câu: Nam đang làm bài_tập .

Trong tiếng Việt, thứ tự từ và các thành phần câu giúp người nghe, đọc hiểu được đâu là chủ ngữ, vị ngữ và các bổ ngữ, thành phần phụ trong câu. Khi thay đổi trật tự từ hoặc các cụm từ, ngữ nghĩa có thể bị thay đổi hoặc gây ra các nhập nhằng. Ví dụ với một câu tiếng Việt: "Anh ấy nói chuyện với cô giáo ở trường." có thể được phân tích cú pháp theo hai cách trong Hình 1.2 và 1.3.



Hình 1.2: Cách 1: cây cú pháp thành phần của câu: "Anh ấy nói_chuyện với cô_giáo ở trường.".

Câu này có thể được phân tích theo hai cách khác nhau do sự nhập nhằng cú pháp của cụm từ "ở trường", dẫn đến hai ngữ nghĩa khác nhau. Sự nhập nhằng này xuất phát từ việc không rõ "ở trường" bổ nghĩa cho "cô giáo" hay cho hành động "nói chuyện". Với hai cách phân tích cú pháp trên, có thể hiểu ngữ nghĩa của câu tương ứng như sau:



Hình 1.3: Cách 2: cây cú pháp thành phần của câu: "Anh ấy nói_chuyện với cô_giáo ở trường.".

- Cách 1: cụm giới từ "ở trường" bổ nghĩa cho động từ "nói chuyện", mô tả địa điểm diễn ra hành động nói chuyện. Tức là anh ấy đang ở trường đó, và nói chuyện với cô giáo.
- Cách 2: cụm giới từ "ở trường" bổ nghĩa cho danh từ cô giáo, mô tả vị trí hoặc địa điểm của cô giáo và tạo thành một cụm danh từ "cô giáo ở trường". Tức là anh ấy đang nói chuyện với một người, người này là cô giáo ở trường.

Ví dụ trên cho thấy sự liên quan chặt chẽ giữa cấu trúc cú pháp và ngữ nghĩa. Việc phân tích ngữ nghĩa văn bản do vậy thường dựa trên nền tảng phân tích cú pháp.

1.1.1.2. Cú pháp phụ thuộc

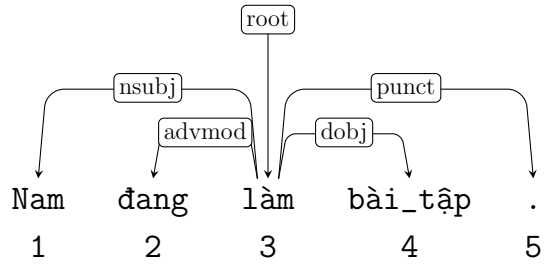
Cú pháp phụ thuộc là cấu trúc cú pháp chứa các từ nối với nhau bởi các quan hệ nhị phân không đối xứng, đó là quan hệ phụ thuộc, có thể đặt tên để phân biệt các loại quan hệ giữa hai từ [81]. Phân tích cú pháp phụ thuộc là phương pháp phân tích bằng cách tập trung vào mối quan hệ phụ thuộc giữa các từ trong câu, thay vì cấu trúc phân cấp của các thành phần câu. Phương pháp này chỉ ra rằng, trong một liên kết cú pháp giữa hai từ thì một từ có vai trò chi phối từ kia theo một mối quan hệ nhất định. Các nội dung chủ yếu của cú pháp phụ thuộc như sau:

1. Cấu trúc cây phụ thuộc: thể hiện mỗi từ trong câu đều đóng vai trò là một nút trong cây phân tích. Từ được chọn làm trung tâm của câu (thường là vị từ - có khả năng trực tiếp làm vị ngữ trong câu) gọi là *root* (gốc câu),

các từ khác trong câu có quan hệ phụ thuộc trực tiếp vào *root*, hoặc phụ thuộc vào các từ khác trong câu (phụ thuộc gián tiếp vào *root*). Mỗi cung quan hệ phụ thuộc đều mang theo nhãn để mô tả loại quan hệ giữa từ phụ thuộc và từ trung tâm.

2. Mối quan hệ phụ thuộc: thể hiện các mối quan hệ khác nhau như chủ ngữ, tân ngữ, định ngữ, bổ ngữ, trạng ngữ, phụ ngữ, ... Mỗi loại quan hệ biểu thị một chức năng ngữ pháp khác nhau của từ trong câu. Các nhãn phụ thuộc chủ yếu: `nsubj` (chủ ngữ danh từ), `obj` (bổ ngữ), `dobj` (bổ ngữ trực tiếp), `iobj` (bổ ngữ gián tiếp), `det` (ý nghĩa hạn định), `amod` (tính từ bổ nghĩa), `advmod` (phụ từ), ...

Ví dụ câu "`Nam đang làm bài_tập.`" sẽ được phân tích cú pháp phụ thuộc như trong Hình 1.4.



Hình 1.4: Cây cú pháp phụ thuộc của câu: Nam đang làm bài_tập.

Cấu trúc phụ thuộc được xác định bởi mối quan hệ giữa một từ trung tâm (*head*) và từ phụ thuộc (*dependent*) của nó. Cấu trúc phụ thuộc thích hợp với các ngôn ngữ có trật tự từ tự do, như tiếng Séc, hay Thổ Nhĩ Kì.

1.1.2. Ngữ nghĩa

Ngữ nghĩa là lĩnh vực nghiên cứu về cách con người hiểu và diễn giải ý nghĩa của từ ngữ và câu trong ngôn ngữ. Để hiểu rõ ngữ nghĩa của một câu, trước hết cần nắm được ý nghĩa của từng từ, đồng thời phân tích mối quan hệ giữa các từ khi chúng kết hợp thành cụm từ và sự liên kết giữa các cụm từ trong câu. Trong xử lý ngôn ngữ tự nhiên, việc biểu diễn và phân tích ngữ nghĩa chủ yếu tập trung vào hai khía cạnh chính là ngữ nghĩa từ vựng và ngữ nghĩa cú pháp [5].

Ngữ nghĩa từ vựng nghiên cứu và hiểu ý nghĩa của các từ trong ngôn ngữ, từ ý nghĩa cơ bản trong từ điển đến những sắc thái đa dạng trong các ngữ cảnh

khác nhau. Nó bao gồm việc xác định đồng nghĩa (cùng ý nghĩa) và trái nghĩa (nghĩa ngược lại) của từ, đa nghĩa (một từ có thể có nhiều nét nghĩa). Việc phân tích các thành phần cấu thành của từ, bao gồm cả tiền tố và hậu tố, cũng giúp làm rõ hơn về ý nghĩa và cách mà các từ được hình thành.

Ngữ nghĩa cú pháp nghiên cứu cách các từ kết hợp lại để tạo thành câu có ý nghĩa nào đó. Để hiểu rõ ngữ nghĩa của một câu, trước hết cần nắm được ý nghĩa của từng từ, đồng thời phân tích mối quan hệ giữa các từ khi chúng kết hợp thành cụm từ và sự liên kết giữa các cụm từ với nhau. Mục tiêu của ngữ nghĩa cú pháp là tìm hiểu những sự tình của thực tế được nói đến trong câu, từ đó giúp chúng ta hiểu được cách mà người nói và người nghe tương tác trong giao tiếp. Mỗi sự tình là một cấu trúc nghĩa bao gồm sự tình đó do vị từ biểu hiện và các tham tố bị vị từ chi phối, đó chính là các vai nghĩa. Nghĩa của vị từ trong câu quy định các kết trị (*valence*) của nó trong những bối cảnh giao tiếp và mục đích giao tiếp cụ thể.

Luận án sẽ trình bày về các thông tin ngữ nghĩa cần biểu diễn trong văn bản, các mô hình và ngôn ngữ biểu diễn ngữ nghĩa hiện có trong các phần tiếp theo.

1.1.2.1. Các thông tin ngữ nghĩa

Thông tin ngữ nghĩa của câu bao gồm nghĩa của các từ tạo nên nghĩa toàn thể mà câu đó truyền tải, được sinh ra từ một chỉnh thể cấu trúc nghĩa biểu hiện hoàn chỉnh. Cấu trúc nghĩa biểu hiện của câu chính là cấu trúc các vai nghĩa. Những vai nghĩa có tính chất bắt buộc bị chi phối bởi ý nghĩa từ vựng - ngữ pháp của vị từ trung tâm. Tức là, những vị từ có ý nghĩa từ vựng - ngữ pháp khác nhau sẽ quy định một bộ các vai nghĩa bắt buộc khác nhau. Nhận diện tường tận thông tin ngữ nghĩa của câu là cả một quá trình phức tạp. Tuy nhiên, nếu thông tin ngữ nghĩa của câu được nhận diện rõ ràng, nó sẽ góp phần quan trọng trong xử lí ngôn ngữ tự nhiên, giúp hệ thống không chỉ nhận diện từ ngữ mà còn có thể hiểu, suy luận và phản hồi phù hợp với ý nghĩa câu.

Ngữ nghĩa biểu thị mối quan hệ giữa các từ, cụm từ, kí hiệu, ... và ý nghĩa của chúng trong câu, đề cập đến ý nghĩa hoặc hàm ý được gắn liền với từng phần của ngôn ngữ. Nó tập trung vào việc hiểu ý nghĩa của các từ, câu, hoặc văn bản, bao gồm cả ý nghĩa tường minh và ngụ ý mà thông điệp đó chứa đựng. Vì thế, các mô hình biểu diễn ngữ nghĩa thường được thiết kế để có thể nắm bắt các thông tin này. Cụ thể, các thông tin ngữ nghĩa gồm có [8]:

1. Sự kiện: biểu diễn các hành động, quá trình hoặc trạng thái liên quan đến

các đối tượng. Chúng thường được biểu hiện qua động từ trong câu và có vai trò trung tâm trong việc xác định cấu trúc ngữ nghĩa của câu.

Ví dụ: `Hôm nay, Nam đến công ty lúc 9h sáng, vì anh ấy gặp sự cố ở giữa đường.`

Các sự kiện trong câu này sẽ là "`đến`", "`gặp`".

2. Tham tố: Mỗi vị từ đều đòi hỏi một số danh ngữ đi kèm theo nó trong câu, những danh ngữ này được gọi là tham tố. Số lượng các danh ngữ mà một vị từ đòi hỏi bắt buộc phải có là các kết trị của vị từ đó. Tham tố chính (bắt buộc) là các thành phần chính trong một câu, có vai trò thiết yếu trong việc xác định ý nghĩa của hành động hoặc sự kiện được mô tả. Những tham tố phụ (tùy chọn) thường là những tham tố bổ nghĩa, độc lập hoặc có quan hệ khác biệt như thời gian, địa điểm, cách thức, phương tiện và mức độ [13]. Trong ví dụ trên, các tham tố chính là: `Nam` (chủ thể của hành động "`đến`", "`gặp`"), `công ty` (đích đến), `sự cố` (bị thể). Các tham tố phụ gồm có: `hôm nay`, `lúc 9h sáng` (thông tin về thời gian), `giữa đường` (thông tin về địa điểm).

3. Vai nghĩa: *Lucien Tesnière*, nhà ngôn ngữ học người Pháp, là người đầu tiên đề cập đến việc nghiên cứu các vai nghĩa [113]. Theo ông, cấu trúc cú pháp của câu gồm một động từ làm trung tâm, xoay quanh là các diễn tố (*actant*) liên kết với động từ đó. Các diễn tố này là các đối tượng đảm nhận vai trò ngữ nghĩa (*semantic roles*) trong một hành động hoặc sự kiện nào đó. Khái niệm "diễn tố" được dùng để chỉ các vai nghĩa bắt buộc của một vị từ. Sau này các nhà ngôn ngữ học đã dùng khái niệm "tham tố" (*argument*) với nội hàm rộng hơn như đã đề cập ở trên, và từ đó "tham tố" được sử dụng phổ biến trong nhiều lí thuyết ngôn ngữ học hiện đại. Vai nghĩa giúp xác định cách mà các thực thể (đối tượng) liên quan đến hành động hoặc sự kiện trong câu. Một số hệ thống vai nghĩa phổ biến là FrameNet [13] và PropBank [67]. Các vai nghĩa của PropBank cũng đã được mở rộng và phát triển thành mô hình biểu diễn ngữ nghĩa trừu tượng AMR [14]. Ngoài ra, một hệ thống phân loại ngữ nghĩa từ vựng lớn và nổi tiếng của tiếng Anh đó là VerbNet [68]. Trong ví dụ trên, các vai nghĩa được định nghĩa gồm có: Agent - tác thể (`Nam, anh ấy`), Patient - bị thể (`sự cố`), Location - địa điểm (`giữa đường`), Time - thời gian (`hôm nay, lúc 9h sáng`).

4. Đồng sở chỉ [28]: Trong ngôn ngữ học, đồng sở chỉ (đồng tham chiếu - *co-reference*) xảy ra khi hai hay nhiều biểu thức cùng đề cập tới một người hoặc vật, tức là chúng có cùng đối tượng tham chiếu. Trong ví dụ trên, ‘‘Nam’’ của vế thứ nhất với ‘‘anh ấy’’ ở vế thứ hai là đồng sở chỉ.

   Việc xác định đồng sở chỉ thường không đơn giản. Chẳng hạn, trong câu "Bill nói rằng anh ấy sẽ đến", từ "anh ấy" có thể chỉ "Bill", nhưng cũng có thể không. Một số loại của đồng sở chỉ gồm có [60]:

   - Hồi chỉ (*anaphora*): đại từ hoặc biểu thức xuất hiện sau đối tượng mà nó đề cập đến. Ví dụ: "Lan rất chăm chỉ. Cô ấy luôn hoàn thành bài tập đúng hạn.", "Cô ấy" và "Lan" là hồi chỉ.
   - Khứ chỉ (*cataphora*): đại từ hoặc biểu thức xuất hiện trước đối tượng mà nó đề cập đến. Ví dụ: "Cô ấy rất chăm chỉ. Lan luôn hoàn thành bài tập đúng hạn.", "Cô ấy" và "Lan" là khứ chỉ.
   - Tiền đề phân tách (*split antecedents*): Đại từ tham chiếu tới nhiều hơn một tiền đề riêng biệt. Ví dụ: "Nam đến trường, và Mai ở nhà. Họ đều đang học bài.", "Họ" chỉ cả "Nam" và "Mai".
   - Cụm danh từ đồng sở chỉ (*coreferring noun phrases*): hai cụm danh từ khác nhau nhưng cùng chỉ một thực thể. Ví dụ: "Apple Inc. đã công bố sản phẩm mới. Tập đoàn công nghệ này kỳ vọng sẽ đạt doanh thu kỷ lục.", "Apple Inc." và "Tập đoàn công nghệ này" là hai cụm danh từ đồng sở chỉ.

   Việc xác định những biểu thức nào đồng tham chiếu là một phần quan trọng trong việc phân tích hay hiểu nghĩa của câu, và thường cần thông tin từ ngữ cảnh, các kiến thức thực tế (như các tên gọi chung được gán cho một loài nào đó, các giới tính ngữ pháp, ...).

5. Quan hệ thời gian: mô tả thông tin và quan hệ thời gian giữa các sự kiện, dự đoán thứ tự tương đối của các sự kiện theo thời gian. Một mô hình biểu diễn thời gian được phát triển cho nhiều ngôn ngữ là TimeML [51]. Trong ví dụ trên, có hai thực thể thời gian được mô tả là:
   `<TIMEX3 tid=‘‘t1’’ type=‘‘DATE’’ value=‘‘2024-09-27’’>Hôm nay</TIMEX3>`
   `<TIMEX3 tid=‘‘t2’’ type=‘‘TIME’’ value=‘‘09:00’’>9h sáng</TIMEX3>`

6. Quan hệ không gian: xác định và phân loại các yếu tố không gian và mối quan hệ như các địa điểm, đường đi, hướng và các chuyển động cũng như

các cấu hình của chúng. Việc biểu diễn các mối quan hệ không gian đóng vai trò quan trọng trong các lý thuyết nhận thức về ngữ nghĩa, các hệ thống thông tin địa lí hoặc điều hướng robot, là nội dung chính của các cuộc thi SpaceEval [1]. Trong ví dụ trên, thông tin không gian được thể hiện như sau:

```
<LOCATION id=‘‘loc1’’>giữa đường</LOCATION>
```

7. Quan hệ diễn ngôn: Quan hệ diễn ngôn là cách thức các phần của văn bản kết nối với nhau và tạo nên một cấu trúc logic, mối liên kết ý nghĩa giữa các câu, đoạn văn, hay các phần khác của văn bản để hình thành một ý nghĩa hoàn chỉnh. Các quan hệ diễn ngôn bao gồm các mối quan hệ như thời gian, so sánh, tương quan nguyên nhân - kết quả, giải thích, ngoại lệ, và nhiều loại quan hệ khác nhau để giúp văn bản trở nên logic, liên kết và dễ hiểu hơn. Trong ví dụ trên, một số loại quan hệ diễn ngôn được xác định là: `Hôm nay, lúc 9h sáng` (thông tin thời gian), `anh ấy gặp sự cố ở giữa đường` (thông tin về lí do).

Có thể thấy rằng, mỗi thành phần trên đều đóng vai trò nhất định trong việc hiểu và xử lý ngôn ngữ. Các mô hình biểu diễn ngữ nghĩa thường cố gắng kết hợp các thành phần này để tạo ra một hệ thống đủ mạnh có khả năng hiểu và tạo sinh ngôn ngữ tự nhiên hoặc giải quyết các vấn đề liên quan đến ngôn ngữ.

1.1.2.2. Các mô hình và ngôn ngữ biểu diễn ngữ nghĩa

Các mô hình biểu diễn ngữ nghĩa đã trải qua nhiều giai đoạn phát triển. Ban đầu, các hệ hình thức logic được phát triển để nắm bắt ý nghĩa thông qua các quy tắc và biểu thức logic. Các thông tin ngữ nghĩa có thể mô tả được bằng hệ hình thức logic gồm có các sự kiện trong câu, các thực thể tham gia vào sự kiện, mối quan hệ giữa các sự kiện và thực thể, các thông tin lượng hoá, một số loại phủ định và thông tin thời gian. Sau đó, để mở rộng khả năng biểu diễn và chi tiết hoá hơn các thành phần ngữ nghĩa, các nghiên cứu tiếp theo tập trung vào việc biểu diễn ngữ nghĩa dưới dạng đồ thị. Cách tiếp cận này cho phép thể hiện các sự kiện, thực thể, khái niệm, và mối quan hệ ngữ nghĩa giữa chúng một cách trực quan, rõ ràng và linh hoạt hơn.

Các mô hình và ngôn ngữ biểu diễn ngữ nghĩa sẽ được phân loại thành các dạng như sau [61]:

[1] https://alt.qcri.org/semeval2015/task8/#

- Các hệ hình thức dựa vào logic (*Logic-based formalisms*): Một cách biểu diễn ngôn ngữ phổ biến và đơn giản nhất là sử dụng mệnh đề logic bậc nhất (*First-order logic - FOL*). Đây là một hệ thống logic có khả năng diễn đạt ý nghĩa của các tuyên bố sử dụng các nguyên lý lượng tử (*quantifier*) như "tất cả" và "một số" cho các biến và hàm. Các thông tin ngữ nghĩa được biểu diễn gồm có các đối tượng cụ thể như số, hay sự kiện cụ thể.

  Ví dụ, mệnh đề "`Tất cả các số nguyên tố lớn hơn 2 đều là số lẻ`" có thể diễn đạt sử dụng FOL như sau: $\forall(x).prime(x) \wedge more(x, 2) \rightarrow odd(x))$.

  Tuy nhiên, mệnh đề logic bậc nhất có một số hạn chế, chẳng hạn như không thể diễn đạt các thao tác với tập hợp, ví dụ như "`Đếm số lượng số nguyên tố nhỏ hơn 10`". Điều này đòi hỏi các biểu diễn logic phức tạp hơn, như mở rộng bằng cách sử dụng tính toán lambda [15] (*lambda calculus - LC*). Ví dụ trên sẽ được biểu diễn bằng biểu thức: $count(\lambda x.prime(x) \wedge less(x, 10))$. Trong đó, λx biểu thị tập hợp tất cả các x thoả mãn điều kiện đã cho. Sau đó, tác giả *Percy Liang* [76] đã phát triển mô hình biểu diễn ngữ nghĩa ($\lambda - DCS$) để biểu thị ý nghĩa một cách linh hoạt hơn và phong phú hơn dựa vào tính toán lambda và cú pháp phụ thuộc.

- Các hệ hình thức dựa vào đồ thị (*Graph-based formalims*): Một cách khác để biểu diễn ngữ nghĩa là sử dụng các mô hình/hệ hình thức dựa vào đồ thị. Mô hình biểu diễn ý nghĩa của một câu, một đoạn, ... được biểu diễn dưới dạng đồ thị có gán nhãn, trong đó các nút thường biểu thị thực thể/sự kiện và các cạnh biểu thị mối quan hệ ngữ nghĩa giữa các nút. Biểu diễn bằng đồ thị sẽ mang lại một số lợi ích hơn so với các biểu diễn khác như:

  – Dễ đọc và dễ hiểu hơn đối với con người.

  – Có xu hướng trừu tượng hoá khỏi cấu trúc cú pháp và có thể không được liên kết với các từ trong câu, ví dụ như mô hình biểu diễn ngữ nghĩa trừu tượng AMR [14].

  – Có rất nhiều các tài liệu và thuật toán đồ thị để nghiên cứu, sử dụng hoặc học tập.

  Ví dụ về các hình thức dựa trên đồ thị: mô hình biểu diễn ngữ nghĩa trừu tượng (*Abstract Meaning Representation - AMR*) [14], mô hình biểu diễn ngữ nghĩa chú giải nhận thức khái niệm phổ quát (*Universal Conceptual Cognitive Annotation - UCCA* [7]), mô hình biểu diễn ngữ nghĩa phân rã dựa

vào phụ thuộc phổ quát (*Universal Decompositional Semantics on Universal Dependencies - UDS*) [122], ...

- Các ngôn ngữ lập trình (*Program Languages - PLs*): Gần đây, đã có nỗ lực chuyển đổi trực tiếp câu truy vấn ngôn ngữ tự nhiên sang các ngôn ngữ lập trình (PLs) mức cao, đa mục đích như Python, Java, SQL, Bash [109]. Chuyển đổi sang các PL mức cao có những ưu điểm so với việc chuyển đổi thành các hình thức logic có cấu trúc vì một số ưu điểm như có cấu trúc tương đối đơn giản và dễ hiểu, được sử dụng rộng rãi trong cộng đồng phát triển phần mềm, phù hợp với lĩnh vực nghiên cứu mới nổi về học máy tự động [82].

Đặc biệt, một điều quan trọng cần xem xét trong biểu diễn và phân tích ngữ nghĩa là thông tin ngữ nghĩa phổ quát. Mặc dù các ngôn ngữ khác nhau về hình thức, nhưng biểu diễn ngữ nghĩa sẽ giống nhau nếu cùng mô tả một nội dung nào đó. Việc phát triển những mô hình biểu diễn và phân tích ngữ nghĩa đa ngôn ngữ rất quan trọng và là nền tảng trong các ứng dụng sử dụng xử lí ngôn ngữ tự nhiên. Phần tiếp theo, luận án sẽ trình bày về các phương pháp phân tích cú pháp và ngữ nghĩa đã và đang được phát triển.

1.2. Các phương pháp phân tích cú pháp và ngữ nghĩa

Các phương pháp phân tích cú pháp và ngữ nghĩa đã và đang được phát triển và thu hút nhiều sự quan tâm của các nhóm nghiên cứu. Trong phần này, luận án sẽ trình bày về các cách tiếp cận để giải quyết bài toán phân tích cú pháp và ngữ nghĩa. Sau đó, trình bày về các mô hình ngôn ngữ và biểu diễn văn bản.

1.2.1. Phát biểu bài toán

Một bài toán phân tích cú pháp, ngữ nghĩa giới hạn trong câu có thể được phát biểu hình thức như sau:

- Đầu vào:
  - Câu đầu vào là một chuỗi n từ: $x = w_1, w_2, \ldots, w_n$. Thông thường, câu x sẽ được trải qua một số bước tiền xử lý như tách từ và gán nhãn từ loại. Trong đó mỗi w_i chứa thông tin từ và từ loại.
- Đầu ra: Thông tin cú pháp, ngữ nghĩa của câu x theo mô hình hoặc định dạng cụ thể.

Độ đo đánh giá

Để đánh giá chất lượng một hệ thống phân tích cú pháp, người ta thường sử dụng các tiêu chí độ chính xác, độ phủ và độ đo F_1 như sau:

$$\text{Precision } (P) = \frac{\text{Số thành phần phân tích đúng}}{\text{Tổng số thành phần trên cây phân tích}}$$

$$\text{Recall } (R) = \frac{\text{Số thành phần phân tích đúng}}{\text{Tổng số thành phần trên cây đúng (gold)}}$$

$$F_1 = \frac{2 \times R \times P}{R + P}$$

Trong đó, mỗi chỉ số này sẽ được thay đổi theo từng bài toán cụ thể. Ví dụ đối với các hệ thống phân tích cú pháp phụ thuộc thường sử dụng hai chỉ số *UAS* (*Unlabeled Attachment Score*): độ chính xác trên từ trung tâm, chưa có nhãn phụ thuộc; *LAS* (*Labeled Attachment Score*): là độ chính xác tính cả trên từ trung tâm và nhãn phụ thuộc tương ứng.

Đối với bài toán phân tích ngữ nghĩa, việc đánh giá chất lượng của mô hình sinh biểu diễn ngữ nghĩa thường sử dụng độ đo Smatch[2] cũng được tính theo công thức trên. Tuy nhiên, một quan hệ ngữ nghĩa có thể được biểu diễn ở dạng logic mệnh đề quan hệ (biến, giá trị) hoặc quan hệ (biến, biến). Điểm Smatch được tính bằng tất cả số bộ ba có thể đối sánh tối đa trong tất cả các biến ánh xạ có thể có.

1.2.2. Các phương pháp phân tích cú pháp - ngữ nghĩa

Trong xử lí ngôn ngữ tự nhiên, việc phân loại các phương pháp phân tích cú pháp và ngữ nghĩa có thể dựa vào nhiều tiêu chuẩn. Thông thường, dựa vào công nghệ sử dụng, có thể phân biệt thành hai hướng chính: Các phương pháp truyền thống (dựa vào luật, dựa vào thống kê, các phương pháp kết hợp) và các phương pháp sử dụng mạng nơ-ron [55].

Đối với bài toán phân tích cú pháp thành phần, một số phương pháp truyền thống nổi bật đã được phát triển như thuật toán CYK (*Cocke-Younger-Kasami* [63]), thuật toán Earley [52], thuật toán Chart Parsing [84], thuật toán Shift-Reduce [53], thuật toán dựa vào bước chuyển [54].

Đối với phân tích cú pháp phụ thuộc, có hai phương pháp phân tích cú pháp cơ bản. Thứ nhất là thuật toán phân tích cú pháp phụ thuộc dựa vào đồ thị được *Eisner* (1996), *McDonald* cùng cộng sự (2005) phát triển (công cụ MSTParser[3]),

[2] `https://amr.isi.edu/smatch-13.pdf`

[3] http://sourceforge.net/projects/mstparser/

thực hiện phân tích cú pháp phụ thuộc thông qua tham số hóa mô hình phụ thuộc dựa vào các đồ thị con và huấn luyện các tham số trên toàn bộ các đồ thị. Sử dụng suy luận toàn cục trong hệ thống để tìm những đồ thị có trọng số cao nhất trong số các cách thiết lập tất cả các đồ thị. Thứ hai là mô hình phân tích cú pháp phụ thuộc dựa trên các bước chuyển được các nhóm *Yamada* và cộng sự (2003), *Nivre* cùng cộng sự (2004) phát triển (công cụ MaltParser[4]). Thuật toán này thực hiện phân tích cú pháp phụ thuộc thông qua các bước chuyển từ trạng thái phân tích này tới trạng thái phân tích khác. Các tham số trong mô hình thường được huấn luyện sử dụng kĩ thuật phân lớp chuẩn để dự đoán bước chuyển tiếp theo từ một tập hợp các bước chuyển trước đó.

Các hệ thống phân tích ngữ nghĩa truyền thống ban đầu chủ yếu được xây dựng dựa trên luật ví dụ như hệ thống SAVVY [59]. Đây là một hệ thống dựa trên luật, thực hiện phương pháp khớp mẫu. Mặc dù thiết kế khá đơn giản nhưng bị hạn chế bởi tính "nông" của phương pháp khớp mẫu này, nghĩa là hệ thống chỉ có thể xử lý các mẫu cụ thể của đầu vào, chỉ áp dụng cho những lĩnh vực đặc thù, khó có thể tổng quát hoá các mẫu đã định nghĩa trước. Một phương pháp khác sử dụng các hệ thống dựa trên cú pháp như hệ thống LUNAR [123]. Hệ thống cho phép người dùng đặt câu hỏi phức tạp bằng ngôn ngữ tự nhiên và sau đó chuyển đổi câu hỏi đó thành truy vấn cơ sở dữ liệu, giúp các nhà khoa học truy xuất thông tin liên quan từ một cơ sở dữ liệu chứa các phân tích hoá học và dữ liệu địa chất.

Đối với hướng thứ hai, các mô hình dựa vào mạng nơ ron cũng đã trải qua nhiều giai đoạn phát triển từ các mạng nơ ron cơ bản cho đến các mô hình học sâu phức tạp như các mạng Transformer. Trong giai đoạn đầu, các mạng nơ ron hồi tiếp (*Recurrent Neural Network - RNN*) và các biến thể như LSTM (*Long Short-Term Memory*) và GRU (*Gated Recurrent Unit*) được sử dụng rộng rãi. Những mô hình này giúp xử lý dữ liệu tuần tự, đặc biệt phù hợp với các chuỗi ngôn ngữ tự nhiên. Trong phân tích cú pháp-ngữ nghĩa, RNN và LSTM có thể mã hóa câu đầu vào thành các biểu diễn ngữ nghĩa và sinh ra các cấu trúc cú pháp hoặc ngữ nghĩa tương ứng. Kiến trúc Seq2Seq và LSTM đã trở thành nền tảng để xây dựng các mô hình phân tích cú pháp và ngữ nghĩa có hiệu suất cao như [112], [64], [132].

Sau đó, các hệ thống phân tích cú pháp, ngữ nghĩa đã kết hợp cơ chế chú ý (*attention*) [12] vào kiến trúc Seq2Seq. Cơ chế này cho phép mô hình tập trung

[4]http://www.maltparser.org/

vào các phần quan trọng của chuỗi đầu vào trong khi sinh ra các phần tử của chuỗi đầu ra, giúp cải thiện hiệu suất trong các bài toán phân tích ngữ nghĩa phức tạp. Một số mô hình phân tích cú pháp và ngữ nghĩa sử dụng cơ chế chú ý đạt hiệu quả cao như [38], [111].

Giai đoạn tiếp theo phải kể đến các mô hình ngôn ngữ huấn luyện trước (*Pretrained Language Models*) như BERT [50], RoBERTa [77], T5 [26], GPT [97] đã mang lại những cải tiến đáng kể cho bài toán phân tích cú pháp - ngữ nghĩa. Các mô hình này được huấn luyện trên lượng dữ liệu khổng lồ và có thể được tinh chỉnh (*fine-tuning*) cho các nhiệm vụ cụ thể. Với khả năng hiểu ngữ cảnh từ cả hai chiều, các mô hình này giúp xác định mối quan hệ giữa các từ trong câu một cách chính xác, nâng cao độ chính xác trong việc phân tích cấu trúc ngữ pháp và ngữ nghĩa của câu.

Tóm lại, các phương pháp phân tích cú pháp - ngữ nghĩa đã và đang được phát triển bởi nhiều nhóm nghiên cứu, đa dạng kĩ thuật như các phương pháp dựa vào luật, thống kê và mạng nơ-ron. Các phương pháp này không chỉ nâng cao hiệu suất phân tích cú pháp - ngữ nghĩa mà còn mở ra nhiều hướng nghiên cứu mới. Phần tiếp, luận án sẽ trình bày sâu hơn về các mô hình ngôn ngữ và cách chúng cải thiện khả năng hiểu ngữ nghĩa trong các tác vụ ngôn ngữ tự nhiên.

1.2.3. Mô hình ngôn ngữ và biểu diễn văn bản

Mô hình ngôn ngữ (*Language Model - LM*) là các mô hình tính toán xác suất của một chuỗi từ hoặc dự đoán từ tiếp theo dựa trên ngữ cảnh của các từ trước đó. Mô hình ngôn ngữ huấn luyện trước (*Pretrained Language Models*) là các mô hình được huấn luyện trên khối lượng lớn dữ liệu văn bản trước khi được tinh chỉnh (*fine-tune*) cho các nhiệm vụ cụ thể. Ý tưởng chính là thay vì huấn luyện mô hình từ đầu cho mỗi nhiệm vụ riêng lẻ, ta có thể sử dụng một mô hình đã được huấn luyện trên các nhiệm vụ chung và sau đó tinh chỉnh nó cho nhiệm vụ đặc thù của bài toán. Phần tiếp theo sẽ trình bày về một số mô hình ngôn ngữ cơ bản, được huấn luyện trước phổ biến như: n-grams, Word2vec, FastText, GloVe, BERT và các mô hình ngôn ngữ lớn như Llama, Gemini, GPT.

1.2.3.1. Mô hình n-gram

Mô hình n-gram [17] là một trong những mô hình ngôn ngữ cơ bản nhất, sử dụng chuỗi các từ hoặc ký tự liền kề để dự đoán từ tiếp theo hoặc tính toán xác

suất của một chuỗi từ. Đây là phương pháp phổ biến trong NLP truyền thống, đặc biệt là trong các bài toán như dự đoán từ, phân tích cú pháp, hoặc dịch máy.

Mô hình n-gram hoạt động dựa vào một chuỗi liên tiếp gồm n từ hoặc ký tự trong văn bản. Với $n = 1$: là mô hình unigram, đơn vị chỉ là một từ, $n = 2$: là mô hình bigram, đơn vị là hai từ liên tiếp, $n = 3$: là mô hình trigram, đơn vị là ba từ liên tiếp.

Mô hình n-gram dựa trên giả định rằng xác suất của một từ chỉ phụ thuộc vào (n-1) từ trước đó:

$$P(w_1, w_2, \ldots, w_n) \approx \prod_{i=1}^{n} P(w_i | w_{i-(n-1),\ldots,w_{i-1}})$$

Trong đó, $P(w_i | w_{i-(n-1),\ldots,w_{i-1}})$ là xác suất của từ w_i dựa trên các từ trước đó trong chuỗi n từ. Điều này có nghĩa là mô hình n-gram chỉ dựa trên ngữ cảnh của $n-1$ từ trước đó để dự đoán từ tiếp theo. Xác suất của một n-gram được tính bằng công thức:

$$P(w_i | w_{i-(n-1)}, \ldots, w_{i-1}) = \frac{Count(w_{i-(n-1)}, \ldots, w_i)}{Count(w_{i-(n-1)}, \ldots, w_{i-1})}$$

Mô hình n-gram được sử dụng trong các tác vụ như dự đoán văn bản, nhận dạng giọng nói, ... Đây là mô hình đơn giản, dễ triển khai và có hiệu quả với dữ liệu lớn.

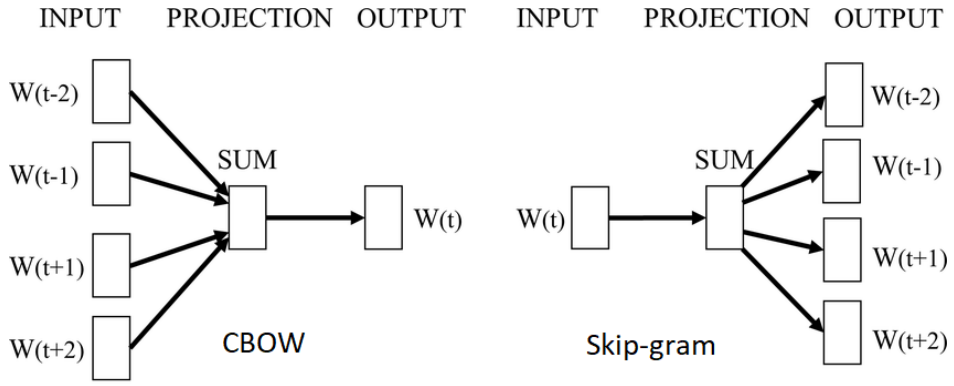
1.2.3.2. Mô hình Word2vec

Mô hình Word2vec được phát triển bởi *Tomas Mikolov* và cộng sự vào năm 2013 [87]. Đây là một trong những phương pháp phổ biến nhất để tạo ra biểu diễn véc tơ từ trong NLP. Nó sinh ra các véc tơ từ có số chiều thấp trong quá trình dự đoán các từ xung quanh mỗi từ. Word2vec học từ dữ liệu văn bản lớn thông qua việc tối ưu hóa một hàm mất mát (*loss function*), thường là hàm *cross-entropy*, để dự đoán từ hoặc ngữ cảnh xung quanh. Khi huấn luyện xong, các véc tơ từ được học sẽ chứa thông tin về ngữ nghĩa và mối quan hệ giữa các từ trong không gian nhiều chiều. Đặc điểm của phương pháp này là nhanh và có thể dễ dàng kết hợp một câu trong một văn bản mới hoặc thêm vào từ vựng. Word2vec cung cấp hai phương pháp chính để học biểu diễn véc tơ từ đó là:

- Skip-gram: có mục tiêu học cách dự đoán các từ ngữ cảnh (*context words*) xung quanh một từ trung tâm (*target word*).

- CBOW (*Continuous Bag of Words*): thực hiện ngược lại so với Skip-gram: dự đoán từ trung tâm từ các từ ngữ cảnh xung quanh

Cụ thể, kiến trúc của hai mô phương pháp được mô tả trong Hình 1.5.



Hình 1.5: Kiến trúc của mô hình CBOW và Skip-gram.

Với Word2vec tiếng Việt, một mô hình nổi tiếng thường được sử dụng trong các bài toán NLP là mô hình huấn luyện sẵn của tác giả Vũ Xuân Sơn và cộng sự [119].

1.2.3.3. Mô hình FastText

Một trong những cải tiến của mô hình Word2vec là FastText [16] - được phát triển bởi Facebook AI Research (FAIR)[5]. Với mục đích cải thiện hiệu quả và tốc độ của Word2vec, FastText được mở rộng bằng cách xem xét các từ con (*subword*) thay vì chỉ học trên các từ đầy đủ, giúp mô hình có thể xử lý tốt hơn các từ chưa xuất hiện trong tập dữ liệu huấn luyện.

FastText hoạt động dựa trên một ý tưởng chính: mỗi từ không chỉ là một đơn vị độc lập mà còn có thể được chia thành các n-gram ký tự (*subword*). Khi gặp các từ mới (*OOV - out-of-vocabulary*), FastText có thể tạo ra một biểu diễn bằng cách tổng hợp các n-grams, điều này đặc biệt hữu ích trong các ngôn ngữ có nhiều hình thái biến đổi hoặc khi đối mặt với các từ hiếm. FastText cũng cung cấp hai cơ chế hoạt động như Word2vec đó là dự đoán các từ xung quanh từ cho trước trong một câu và dự đoán một từ dựa vào ngữ cảnh của các từ xung quanh.

FastText cung cấp các mô hình đã được huấn luyện trước (*pre-trained models*) cho 157 ngôn ngữ khác nhau (trong đó có tiếng Việt). Các mô hình này được

[5]`https://github.com/facebookresearch/fastText`

xây dựng từ dữ liệu lấy từ Wikipedia, giúp nó hỗ trợ một loạt ngôn ngữ phong phú, bao gồm cả những ngôn ngữ ít phổ biến hoặc có ít tài nguyên.

1.2.3.4. Mô hình GloVe

GloVe (*Global Vectors for Word Representation*) là một mô hình học biểu diễn từ được phát triển bởi Stanford University [100]. Mô hình GloVe được thiết kế để học các véc tơ biểu diễn từ dựa trên mối quan hệ ngữ nghĩa giữa các từ trong một tập dữ liệu văn bản lớn. GloVe là một mô hình dựa trên ma trận đồng xuất hiện từ (*word co-occurrence matrix*), và nó khác biệt với các mô hình như Word2vec ở cách tiếp cận tổng thể trong việc học biểu diễn ngữ nghĩa của từ.

Thống kê số lần xuất hiện của từ trong kho ngữ liệu là thông tin chính cung cấp cho quá trình học véc tơ biểu diễn từ không giám sát. Một số ký hiệu được sử dụng trong mô hình:

- X là ma trận biểu diễn số lần xuất hiện đồng thời của 2 từ. Với X_{ij} là số lần xuất hiện của từ thứ j trong ngữ cảnh của từ thứ i. Ngữ cảnh của một từ là chuỗi từ kết hợp với nó hoặc bao xung quanh nó, đủ để làm cho nó được cụ thể hoá và hoàn toàn xác định về nghĩa.
- $X_i = \sum_k X_{ik}$ là số lần xuất hiện của từ bất kỳ trong ngữ cảnh từ thứ i.
- $P_{ij} = P(j|i) = \frac{X_{ij}}{X_i}$ là xác suất để từ thứ j xuất hiện trong ngữ cảnh từ thứ i.

Tỉ lệ P_{ik}/P_{jk} phụ thuộc vào 3 từ i, j, k. Mô hình được đưa ra như sau:

$$F(w_i, w_j, \tilde{w}_k) = P_{ik}/P_{jk} \tag{1.1}$$

Với $w \in \mathbb{R}^d$ và $\tilde{w} \in \mathbb{R}^d$ là những véc tơ từ biểu diễn trong không gian. F biểu diễn thông tin của tỉ lệ P_{ik}/P_{jk} trong không gian véc tơ. Vì không gian véc tơ có cấu trúc tuyến tính nên công thức tính F (1.1) có thể viết lại như sau:

$$F(w_i - w_j, \tilde{w}_k) = P_{ik}/P_{jk} \tag{1.2}$$

Tham số của F trong công thức 1.2 là các véc tơ còn kết quả vế bên phải là một giá trị. Trong khi F là một hàm chức năng phức tạp nào đó, F có thể là mạng nơron. Để tránh mất cấu trúc tuyến tính, thì:

$$F((w_i - w_j)^T \tilde{w}_k) = P_{ik}/P_{jk} \tag{1.3}$$

1.2.3.5. Mô hình BERT

Mô hình BERT (*Bidirectional Encoder Representations from Transformers)*: BERT [35] là một mô hình huấn luyện sẵn (*pre-trained model*), học ra các véc tơ đại diện theo ngữ cảnh hai chiều của từ (từ trái qua phải và từ phải qua trái), được sử dụng để giải quyết các bài toán khác trong lĩnh vực NLP.

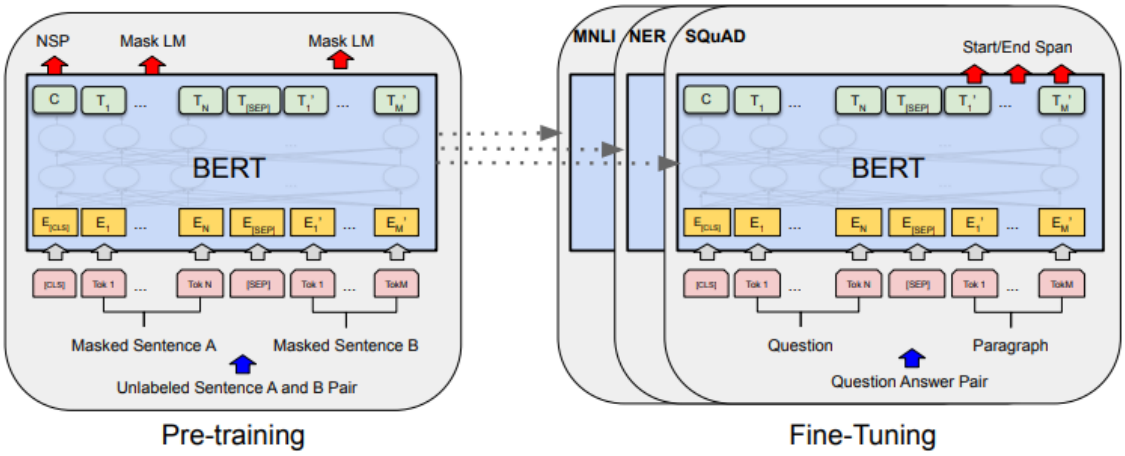
Một trong những đặc điểm nổi bật của BERT là có kiến trúc gồm một bộ mã hoá transformer hai chiều, để có thể tận dụng việc phát hiện các phụ thuộc với khoảng cách xa (*long-distance dependencies*). BERT là một mô hình khổng lồ, với hàng triệu tham số. Đối với mô hình BERT, có thể tiền huấn luyện với hai tác vụ chính. Tác vụ thứ nhất là "Mô hình ngôn ngữ có mặt nạ" (*Masked Language Model*): Để mô hình học ngữ cảnh hai chiều, trong quá trình huấn luyện, BERT che "mặt nạ" (*mask*) ngẫu nhiên 15% các từ trong câu và yêu cầu mô hình đoán từ đó dựa trên các từ còn lại. Điều này giúp mô hình học cách hiểu ngữ cảnh của từ trong cả hai chiều trái và phải. Tác vụ thứ hai là "Mô hình dự đoán câu tiếp theo" (*Next Sentence Prediction*): BERT được huấn luyện để dự đoán xem hai câu có liên quan đến nhau hay không. Điều này giúp mô hình nắm bắt được các mối quan hệ ngữ nghĩa giữa các câu, hữu ích trong các tác vụ như hỏi đáp và suy diễn văn bản.

Google cung cấp các mô hình BERT đã được tiền huấn luyện sẵn với hai phiên bản chính: BERT base (12 lớp Encoder, 110 triệu tham số) và BERT large (24 lớp Encoder, 340 triệu tham số).

Sau giai đoạn tiền huấn luyện, BERT có thể được tinh chỉnh (*fine-tune*) trên các tác vụ cụ thể bằng cách thêm một vài lớp phía trên mô hình đã được huấn luyện trước, như mô tả trong Hình 1.6.

BERT đã thu được kết quả tối ưu cho 11 nhiệm vụ xử lý ngôn ngữ tự nhiên, bao gồm việc cải tiến kết quả của nhiệm vụ GLUE benchmark lên 80.4% (cải tiến thêm 7.6%) và SQuAD v.1.1 với điểm F_1 trên tập kiểm thử đạt 93.2% (cải tiến thêm 1.5%), tốt hơn con người 2%.

Đối với tiếng Việt, PhoBERT được phát triển vào năm 2020 [31], là một mô hình được huấn luyện sẵn cho tiếng Việt. Có hai phiên bản của PhoBERT là PhoBERT base (với 12 khối transformer) và PhoBERT large (với 24 khối transformer). PhoBERT được huấn luyện trên khoảng 20GB dữ liệu, sử dụng VNCoreNLP để tách từ cho dữ liệu đầu vào.



Hình 1.6: Tiến trình huấn luyện trước và tinh chỉnh của mô hình BERT [35].

1.2.3.6. Các mô hình ngôn ngữ lớn

Mô hình ngôn ngữ lớn (*Large language models - LLMs*) là các mô hình học sâu (*deep learning*) có hàng tỷ hoặc thậm chí hàng trăm tỷ tham số, được huấn luyện trên tập dữ liệu văn bản lớn. Những mô hình này được thiết kế để dự đoán từ tiếp theo trong một chuỗi từ hoặc để sinh văn bản mới dựa trên ngữ cảnh đã cho. Các mô hình ngôn ngữ lớn có thể thực hiện nhiều nhiệm vụ khác nhau trong NLP như: sinh văn bản, trả lời câu hỏi, dịch máy, tóm tắt văn bản, phân tích cảm xúc, ... Một số mô hình ngôn ngữ lớn nổi tiếng và được sử dụng trong nhiều tác vụ NLP như:

- GPT (*Generative Pre-trained Transformer*[6]): Là mô hình ngôn ngữ lớn được phát triển bởi OpenAI nổi bật với kích thước và khả năng xử lý ngôn ngữ tự nhiên vượt trội. Với 175 tỷ tham số, GPT-3 là một trong những mô hình đầu tiên của OpenAI được công nhận rộng rãi trong lĩnh vực trí tuệ nhân tạo nhờ khả năng sinh văn bản phức tạp, tự nhiên, và hiệu quả trong nhiều ngữ cảnh khác nhau. Tiếp nối thành công của GPT-3, OpenAI đã phát triển phiên bản tiếp theo, GPT-4, một mô hình đa phương thức lớn hơn, có thể xử lý đầu vào văn bản và hình ảnh. Khả năng này cho phép GPT-4 không chỉ tạo phản hồi dựa trên văn bản mà còn phân tích và trả lời các câu hỏi liên quan đến hình ảnh, mở rộng ứng dụng của nó trong các lĩnh vực như nhận diện hình ảnh, tạo mô tả hình ảnh, và hỗ trợ trong các tác vụ kết hợp giữa văn bản và hình ảnh.

[6] `https://openai.com/chatgpt/`

- Llama[7]: Là mô hình ngôn ngữ lớn được phát triển bởi Meta AI nhằm đáp ứng các nhu cầu trong xử lý ngôn ngữ tự nhiên, trí tuệ nhân tạo, chú trọng vào các tác vụ liên quan đến việc hiểu, tạo văn bản. Với mục tiêu cạnh tranh với các mô hình ngôn ngữ lớn nổi tiếng như GPT của OpenAI và T5 của Google, Llama tập trung vào việc tối ưu hóa kích thước và hiệu năng, đồng thời mở rộng khả năng hiểu biết và sinh văn bản trong nhiều ngữ cảnh khác nhau. Meta AI đã phát triển một số phiên bản của Llama, trong đó Llama 2 và Llama 3 (8 tỷ tham số và 80 tỷ tham số). Llama 3 được huấn luyện dựa trên hơn 15 nghìn tỷ token từ nguồn dữ liệu đa ngôn ngữ và đa dạng như sách, báo, ... Llama 3 không chỉ mang lại hiệu năng cao mà còn mở ra nhiều ứng dụng rộng rãi, từ trợ lý ảo, dịch thuật đến sáng tạo nội dung và tự động hóa quy trình làm việc.

- Gemini[8] [43]: Một hệ mô hình ngôn ngữ lớn đa phương thức phát hành bởi Google DeepMind vào cuối năm 2023, đóng vai trò là một mô hình ngôn ngữ thay thế cho LaMDA và PaLM 2, nhằm cạnh tranh trực tiếp với GPT-4 và Claude.

Các mô hình ngôn ngữ lớn sở hữu nhiều ưu điểm nổi bật trong việc xử lý ngôn ngữ tự nhiên, cho phép học từ khối lượng dữ liệu khổng lồ và tinh chỉnh cho từng tác vụ cụ thể. Chúng được áp dụng rộng rãi trong các lĩnh vực như trợ lý ảo, dịch thuật, phân tích cảm xúc và tổng hợp văn bản. Các mô hình ngôn ngữ lớn cũng được áp dụng trong luận án để sinh các phân tích ngữ nghĩa cho văn bản tiếng Việt, đóng vai trò quan trọng trong nội dung của Chương 3.

1.3. Một số vấn đề cơ bản về xây dựng ngữ liệu

Kho ngữ liệu (*corpus*) là một tập hợp lớn các văn bản đã được cấu trúc hóa, được dùng như một cơ sở để nghiên cứu ngôn ngữ. Giá trị và chất lượng của kho ngữ liệu phần lớn phụ thuộc vào cách tiếp cận và phương pháp luận của khung lý thuyết áp dụng. Kho ngữ liệu thường dùng cho phân tích thống kê và kiểm thử giả thuyết, cũng như trắc nghiệm và kiểm tra sự xuất hiện của các quy luật ngôn ngữ trong một miền ngữ liệu cụ thể.

Việc xây dựng kho ngữ liệu (*corpus*) bắt đầu từ giữa thế kỷ 20, với các dự án tiên phong như Brown Corpus - 1961 [42], đánh dấu bước ngoặt trong việc thu

[7] `https://llama.meta.com/`
[8] `https://deepmind.google/technologies/gemini/`

thập và xử lý dữ liệu ngôn ngữ. Ban đầu, các kho ngữ liệu thường nhỏ và chỉ bao gồm văn bản viết. Đến những năm 1990, với sự phát triển của máy tính và internet, quy mô và độ phong phú của ngữ liệu tăng đáng kể, bao gồm cả văn bản nói và viết từ nhiều ngôn ngữ khác nhau. Những dự án như British National Corpus[9] (BNC) và American National Corpus (ANC) [106] đã góp phần chuẩn hóa việc xây dựng kho ngữ liệu lớn, đa dạng, mở đường cho các nghiên cứu ngôn ngữ học hiện đại và các ứng dụng trí tuệ nhân tạo. Để xây dựng các kho ngữ liệu chuẩn và có tính ứng dụng cao, việc tuân thủ một số quy trình xây dựng và tiêu chuẩn là vô cùng quan trọng.

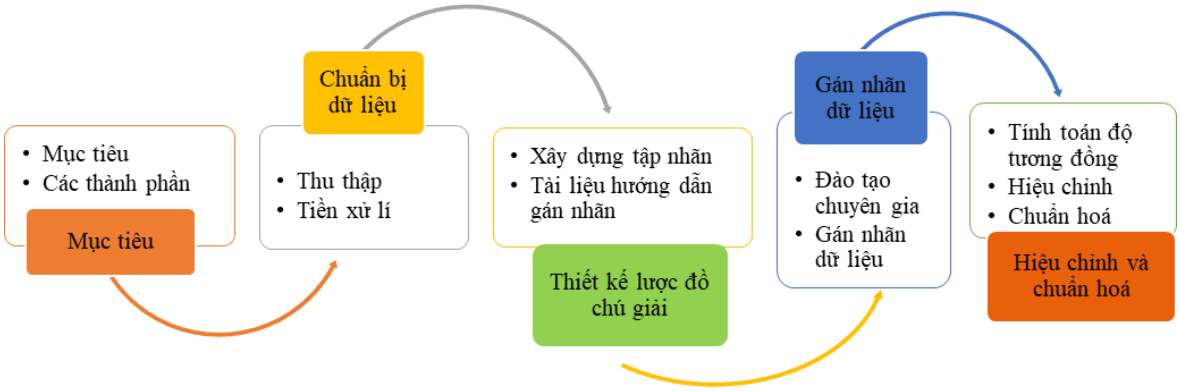
1.3.1. Phương pháp luận

Các kho ngữ liệu đóng vai trò quan trọng trong việc huấn luyện, phát triển và kiểm thử các mô hình học máy và học sâu. Tuy nhiên, nhiều nghiên cứu đã cho thấy các tập dữ liệu này có thể chứa những yếu tố không phù hợp hoặc thiên lệch, dẫn đến sự thiếu chính xác trong cách các hệ thống xử lý và phản hồi đối với một số nhóm người hoặc ngữ cảnh cụ thể. Các mô hình có thể phản ánh, thậm chí khuếch đại các thiên lệch về giới tính, chủng tộc và tôn giáo vốn tồn tại trong dữ liệu huấn luyện. Trong lĩnh vực xử lý ngôn ngữ tự nhiên, một số nhóm nghiên cứu đã phát triển các quy trình chú giải và mô tả dữ liệu chuẩn hóa, chẳng hạn như xây dựng các tài liệu mô tả dữ liệu [39] hoặc các bảng dữ liệu [114]. Các quy trình chú giải và tài liệu mô tả dữ liệu giúp chuẩn hóa việc xây dựng các kho ngữ liệu, đảm bảo tính chính xác, minh bạch và giảm thiểu các rủi ro và thiên lệch, góp phần nâng cao chất lượng và độ tin cậy của dữ liệu.

Quy trình chú giải cho một kho ngữ liệu là quá trình gắn nhãn và bổ sung thông tin cho các đơn vị ngôn ngữ (như từ, cụm từ, câu, đoạn văn) trong một tập dữ liệu ngôn ngữ. Một quy trình chú giải dữ liệu chuẩn hoá bao gồm các bước chi tiết trong Hình 1.7.

Cụ thể, quy trình gán nhãn dữ liệu bao gồm nhiều bước để đảm bảo dữ liệu được gán nhãn chính xác và nhất quán. Đầu tiên, cần xác định mục đích của việc gán nhãn, tức là xác định những thông tin cần mô tả trong dữ liệu là gì. Các dữ liệu thô được thu thập và tiền xử lý. Tiếp theo, cần thiết kế một lược đồ chú giải chi tiết. Ở bước này, những tiêu chuẩn về gán nhãn dữ liệu được khảo sát và tài liệu hướng dẫn gán nhãn sẽ và xây dựng để phù hợp với mục tiêu của kho ngữ liệu và các tiêu chuẩn này. Sau đó, những chuyên gia gán nhãn sẽ được

[9] http://www.natcorp.ox.ac.uk/



Hình 1.7: Quy trình gán nhãn dữ liệu chuẩn.

đào tạo và dữ liệu sẽ được phân bổ cho nhiều người gán nhãn độc lập để tránh sai sót hoặc thiên lệch. Cuối cùng, việc đánh giá chất lượng gán nhãn được thực hiện bằng cách tính toán các độ đo Kappa, độ đo F_1, độ chính xác [11] nhằm kiểm tra độ đồng thuận giữa những người gán nhãn. Nếu có sự chênh lệch lớn, các nhãn cần được rà soát và chuẩn hoá lại để đảm bảo độ chính xác và tin cậy của dữ liệu. Đây chính là bước cuối cùng, hiệu chỉnh và chuẩn hoá dữ liệu.

Khi đã hoàn thành gán nhãn dữ liệu theo đúng quy trình chuẩn, các đặc điểm của dữ liệu, chi tiết về quy trình chú giải và các tiêu chuẩn chuẩn hoá dữ liệu sẽ được mô tả chi tiết trong tài liệu mô tả dữ liệu hoặc bảng dữ liệu. Tài liệu này thường gồm các thông tin sau:

- Cơ sở lí luận: trong tài liệu mô tả dữ liệu, cần mô tả rõ mục tiêu và phạm vi của việc xây dựng kho ngữ liệu chẳng hạn như phục vụ cho các tác vụ phân tích cú pháp, dịch máy, hoặc trích xuất thông tin, . . . Phạm vi bao gồm việc xác định loại ngữ liệu cần thu thập, ví dụ văn bản hay từ vựng. Thông tin về người tạo ra tập dữ liệu (công ty, cơ quan, tổ chức), nhà tài trợ hoặc các mối liên kết tài trợ cũng cần được trình bày cụ thể.

- Các thành phần của kho ngữ liệu: trong tài liệu, cần mô tả rõ loại dữ liệu, các thông số cụ thể của dữ liệu như: biến thể ngôn ngữ, các đặc điểm của dữ liệu, số lượng của mỗi loại dữ liệu, dữ liệu có độc lập hay liên kết với các tài nguyên khác hay không, dữ liệu đại diện cho những mẫu cụ thể nào.

Đồng thời, các thông tin chi tiết về diễn giả, người gán nhãn, thời gian thu thập và gán nhãn cũng cần được mô tả cụ thể.

- Phương pháp thu thập dữ liệu: một phương pháp phổ biến là thu thập các văn bản từ sách, báo, trang web, và kho ngữ liệu khác đã có. Đây là cách nhanh chóng và tiện lợi để xây dựng kho ngữ liệu văn bản, nhưng cần đảm bảo bản quyền và tính đại diện của ngữ liệu. Khi xây dựng, cần mô tả rõ phương pháp thu thập dữ liệu: ai đã tham gia vào quá trình thu thập dữ liệu, dữ liệu thu thập trong khoảng thời gian nào, có quy trình đánh giá nào không, các chiến lược lấy mẫu dữ liệu là gì.

- Tiền xử lí dữ liệu, làm sạch dữ liệu: các kho ngữ liệu cần trải qua các bước tiền xử lí nhằm đảm bảo chất lượng, tính đồng nhất như loại bỏ lỗi chính tả, xử lí các dữ liệu nhạy cảm, tách từ, gán nhãn từ loại, chuyển đổi về các định dạng thống nhất, ... Quá trình này giúp cải thiện độ chính xác và tính hiệu quả của các mô hình học máy và ngôn ngữ. Vì thế, các thông tin này cần được mô tả rõ trong tài liệu mô tả dữ liệu. Bên cạnh đó, các phần mềm và công cụ được sử dụng cho việc tiền xử lý, làm sạch và gán nhãn dữ liệu cũng cần được mô tả chi tiết để đảm bảo tính minh bạch và khả năng tái sử dụng trong các dự án khác.

- Thiết kế lược đồ chú giải và gán nhãn dữ liệu: đối với từng mục tiêu cụ thể, cần phải việc xây dựng cấu trúc kho ngữ liệu và các tài liệu hướng dẫn gán nhãn là rất quan trọng. Đây là giai đoạn đòi hỏi nhiều thời gian và công sức của các chuyên gia để đảm bảo tính nhất quán và độ chính xác. Sau khi chuẩn bị, quá trình gán nhãn có thể được thực hiện bằng nhiều phương pháp khác nhau: thủ công, tự động, bán tự động, hoặc thông qua các nền tảng trực tuyến (*crowdsoursing*). Đối với quá trình gán nhãn dữ liệu thủ công, sau khi kết thúc gán nhãn, cần mô tả rõ độ đồng thuận giữa những người gán nhãn. Một số độ đồng thuận được sử dụng để đánh giá quá trình gán nhãn như:

  - Độ đo Kappa (Cohen's Kappa hoặc Fleiss' Kappa) cho hai người gán nhãn:

$$k = \frac{P_0 - P_e}{1 - P_e}$$

Trong đó: P_0 là tỷ lệ đồng thuận quan sát được – phần trăm các trường hợp mà cả hai người gán nhãn đồng ý với nhau. P_e là tỷ lệ đồng thuận ngẫu nhiên kỳ vọng – xác suất mà hai người gán nhãn đồng ý một cách ngẫu nhiên, tính dựa trên tần suất gán nhãn.

– Độ đo F_1:

$$F_1 = \frac{2 * P * R}{(P + R)}$$

Trong đó, tuỳ vào các bài toán khác nhau mà độ chính xác (P) và độ truy hồi (R) sẽ được định nghĩa cụ thể.

Cả độ đo F_1 và Kappa đều là các chỉ số phổ biến dùng để đánh giá mức độ đồng thuận giữa những người gán nhãn hoặc giữa hệ thống tự động và người gán nhãn. Tuy nhiên, luận án chọn sử dụng độ đo F_1, được tuỳ chỉnh cho các bài toán phân tích cú pháp và ngữ nghĩa, vì nó không chỉ phản ánh độ đồng thuận mà còn cân nhắc đến sự chính xác và đầy đủ của quá trình gán nhãn. Điều này giúp đưa ra đánh giá chi tiết và chính xác hơn về hiệu quả của hệ thống hoặc quá trình gán nhãn, đảm bảo sự cân bằng giữa độ chính xác và độ đầy đủ trong các ứng dụng thực tế.

Ngoài ra, các công cụ gán nhãn dữ liệu được sử dụng trong nghiên cứu cũng cần được mô tả rõ ràng để đảm bảo tính minh bạch và khả năng tái tạo của nghiên cứu. Việc mô tả chi tiết các công cụ này giúp người đọc hiểu được quy trình gán nhãn, cũng như các tiêu chí và thuật toán mà công cụ áp dụng để thực hiện nhiệm vụ gán nhãn.

- Hiệu chỉnh và chuẩn hóa: dữ liệu cần được kiểm tra để đảm bảo chất lượng, loại bỏ lỗi và tính không nhất quán. Các chuẩn quốc tế như TEI[10] hoặc ISO[11] có thể được sử dụng để đảm bảo tính đồng nhất và khả năng tương tác giữa các kho ngữ liệu khác nhau.

- Sử dụng, cập nhật, bảo trì, chia sẻ và công bố dữ liệu: Kho ngữ liệu cần được cập nhật thường xuyên với dữ liệu mới để giữ cho nó phù hợp và chính xác. Bảo trì dữ liệu là cần thiết để tránh lỗi và lỗi cập nhật, đảm bảo kho ngữ liệu luôn sẵn sàng cho người dùng. Cuối cùng, chia sẻ kho ngữ liệu với

[10] `https://tei-c.org/release/doc/tei-p5-doc/en/html/CC.html`
[11] `https://www.iso.org/committee/297592.html`

cộng đồng nghiên cứu và phát triển, nếu phù hợp, và công bố tài liệu liên quan để người dùng có thể hiểu và sử dụng dữ liệu một cách hiệu quả.

Tương tự như vậy, đối với việc xây dựng các mô hình, thẻ mô hình (*model cards*) được Metchel và cộng sự nghiên cứu và phát triển từ năm 2018 [80] với mục đích đưa ra những tiêu chuẩn về mô hình như các mô tả chi tiết về mô hình, mục đích sử dụng, các nhân tố phát triển, độ đo sử dụng, dữ liệu huấn luyện và đánh giá, các thảo luận và phân tích định lượng. Các thành phần trong một thẻ mô hình gồm có:

- Chi tiết về mô hình: gồm có người hoặc tổ chức phát triển mô hình, ngày giờ, các phiên bản, loại mô hình, giấy phép phát hành của mô hình là gì.
- Mục đích sử dụng: các mô hình cần được mô tả chi tiết xem phát triển cho mục đích nào, người sử dụng chính là gì.
- Các nhân tố: các nhân tố trong mô hình gồm có nhóm (những người liên quan như sử dụng hay tác động tới mô hình), các thiết bị và môi trường (mô tả môi trường và các thiết bị mà mô hình được triển khai và cài đặt).
- Độ đo: các thước đo hiệu suất của mô hình, ngưỡng quyết định là gì.
- Dữ liệu huấn luyện và kiểm thử: bộ dữ liệu nào được sử dụng để huấn luyện và kiểm thử hiệu suất của mô hình, có sử dụng các bước tiền xử lí nào không.
- Phân tích định lượng: các phân tích định lượng cần được thực hiện và đánh giá theo các độ đo đã chọn: mô hình hoạt động thế nào với các tham số, môi trường, thiết bị khác nhau. Sự thay đổi về các dữ liệu sẽ ảnh hưởng thế nào tới hiệu suất của mô hình.
- Các vấn đề về đạo đức: các vấn đề về đạo đức là một trong số những thách thức trong quá trình xây dựng mô hình và dữ liệu. Mô hình có sử dụng dữ liệu nhạy cảm nào không, những rủi ro và tác hại nào có thể xảy ra khi sử dụng mô hình.

Việc xây dựng kho ngữ liệu trong NLP là một quá trình thiết yếu, quá trình chú giải dữ liệu có phương pháp luận chặt chẽ từ việc xác định mục tiêu, thu thập và xử lý dữ liệu đến gán nhãn và xây dựng cấu trúc kho ngữ liệu rất quan trọng để đảm bảo chất lượng và khả năng sử dụng. Các thông tin chi tiết về mô

hình, bao gồm thông số, độ đo hiệu suất và các vấn đề đạo đức, cần được mô tả rõ ràng. Tất cả các kho ngữ liệu và mô hình trong luận án sẽ được phát triển dựa trên các quy trình và tài liệu chuẩn hóa này, tạo nền tảng vững chắc cho các nghiên cứu và ứng dụng thực tiễn trong tương lai.

1.3.2. Chuẩn hoá biểu diễn tài nguyên ngôn ngữ

Chuẩn hóa biểu diễn tài nguyên ngôn ngữ là quá trình tạo ra một hệ thống đồng nhất để mô tả và lưu trữ thông tin, giúp đảm bảo rằng các tài nguyên như từ điển, kho ngữ liệu và mô hình cú pháp, ngữ nghĩa được hiểu và sử dụng một cách nhất quán. Quá trình chuẩn hoá rất quan trọng vì nó giúp giảm thiểu sự mơ hồ trong việc diễn đạt ý nghĩa và cấu trúc ngữ pháp, từ đó nâng cao độ chính xác của các hệ thống xử lý ngôn ngữ tự nhiên. Bên cạnh đó, chuẩn hóa còn hỗ trợ việc chia sẻ tài nguyên giữa các nhóm nghiên cứu, tạo điều kiện thuận lợi cho việc phát triển các công nghệ mới và các nghiên cứu đa ngôn ngữ.

ISO.TC 37/SC 4[12] được thành lập vào năm 2002 nhằm phát triển các tiêu chuẩn quốc tế cho việc quản lý các nguồn tài liệu ngôn ngữ, với mục đích cung cấp tiêu chuẩn cho việc chú giải và biểu diễn dữ liệu ngôn ngữ cơ bản. Nhóm làm việc này tập trung vào việc chuẩn hóa các khía cạnh khác nhau của quản lý nguồn tài liệu ngôn ngữ, từ mô tả cơ bản đến các ứng dụng cụ thể trong biểu diễn ngữ nghĩa. Các nhóm tiêu chuẩn cơ bản được mô tả gồm có:

- ***WG 1***: Các mô tả và cơ chế cơ bản là nhóm tiên phong trong việc phát triển các tiêu chuẩn cơ bản cho quản lý nguồn tài liệu ngôn ngữ. Hai tài liệu chính do nhóm này xây dựng là khung cơ bản về chú giải ngôn ngữ (*Linguistic annotation framework - LAF*) và biểu diễn cấu trúc đặc trưng. LAF đề xuất ba điểm chính: sử dụng chú giải bổ sung (*standoff*) thay vì chú giải trong dòng (hoặc ngay tại chỗ - *inline*), mô hình dữ liệu hai tầng phân biệt rõ ràng giữa cấu trúc tham chiếu và nội dung, và biểu diễn các cấu trúc nội dung trong các cấu trúc đặc trưng. Cấu trúc đặc trưng, sử dụng XML, cung cấp chi tiết về cách biểu diễn các cấu trúc đặc trưng và thiết lập nền tảng cho các tiêu chuẩn khác.

- ***WG 2***: Gán nhãn ngữ nghĩa tập trung vào việc định nghĩa các thông tin trong quá trình gán nhãn ngữ nghĩa. Bao gồm: trích xuất thông tin về thời gian trong văn bản, chú giải đối thoại, hỗ trợ việc ghi chép các hành động

[12] `https://www.iso.org/committee/297592.html`

giao tiếp trong cuộc đối thoại, đề xuất sơ đồ chú giải cho các vai trò ngữ nghĩa, làm rõ mối quan hệ giữa động từ và các tham tố trong câu, mô tả các nguyên tắc của chú giải ngữ nghĩa, phác thảo chiến lược SemAF để phát triển các lược đồ chú giải. Ngoài ra, gán nhãn ngữ nghĩa còn tập trung vào thông tin không gian, các quan hệ diễn ngôn, và các phần mở rộng về thông tin định lượng và số lượng như QML và QuantML.

- ***WG 3***: Biểu diễn văn bản đa ngôn ngữ tập trung vào việc phát triển các tiêu chuẩn cho việc quản lý, trao đổi và tích hợp dữ liệu văn bản bằng nhiều ngôn ngữ. Nhóm làm việc này đảm bảo nội dung đa ngôn ngữ được xử lý nhất quán và có khả năng tương tác, giải quyết các vấn đề như mã hóa văn bản và chú giải.

- ***WG 4***: Phát triển các mô hình cho việc biểu diễn và lưu trữ dữ liệu từ vựng. Các tài liệu chính bao gồm: mô tả mô hình cốt lõi của khung đánh dấu từ vựng (*Lexical Markup Framework - LMF*), mô hình từ điển máy đọc được (*Machine-Readable Dictionary - MRD*), phần mở rộng từ nguyên. Các phần khác của tiêu chuẩn bao gồm tuần tự hóa TEI và tuần tự hóa trao đổi cơ sở từ vựng.

- ***WG 5***: Luồng công việc của việc quản lý nguồn ngôn ngữ phát triển các tiêu chuẩn để quản lý toàn bộ vòng đời của tài nguyên ngôn ngữ, từ việc tạo ra đến lưu trữ và truy xuất. Các tiêu chuẩn này hỗ trợ việc quản lý và tích hợp hiệu quả tài nguyên ngôn ngữ trong các hệ thống tính toán.

- ***WG 6***: Chú giải ngôn ngữ tập trung vào việc tạo ra các tiêu chuẩn cho chú giải dữ liệu ngôn ngữ, bao gồm việc chú giải các hiện tượng ngôn ngữ để hỗ trợ các tác vụ xử lý ngôn ngữ khác nhau. Nhóm làm việc này phát triển các khuôn khổ để chú giải các khía cạnh hình thái, ngữ nghĩa và thực dụng của dữ liệu ngôn ngữ, đảm bảo khả năng tương tác và tái sử dụng các tài nguyên ngôn ngữ.

Tất cả các nhóm làm việc này đều đóng góp vào việc xây dựng một hệ thống tiêu chuẩn hóa toàn diện cho việc quản lý và xử lý tài nguyên ngôn ngữ, từ các khái niệm cơ bản đến các ứng dụng cụ thể trong ngữ nghĩa và dữ liệu đa ngôn ngữ.

1.4. Các tài nguyên ngôn ngữ

Các tài nguyên ngôn ngữ là các kho dữ liệu chứa thông tin về từ vựng, ngữ pháp, ngữ nghĩa - được sử dụng trong các hệ thống ngôn ngữ dựa vào luật, thống kê, các mô hình học máy, học sâu để nâng cao hiệu quả của các tác vụ NLP. Các tài nguyên từ vựng cung cấp thông tin về nghĩa của các từ, các mối quan hệ như đồng nghĩa, trái nghĩa và cấu trúc phân cấp. Tài nguyên cú pháp và ngữ nghĩa được xây dựng để mô tả các thông tin về cấu trúc câu, quy tắc ngữ pháp, vai trò ngữ nghĩa của các thành phần trong câu. Các tài nguyên từ vựng (Wordnet, VerbNet, FrameNet, VCL) và các kho văn bản có chú giải (kho ngữ liệu cú pháp thành phần, cú pháp phụ thuộc, các kho ngữ liệu có chú giải ngữ nghĩa) đã được xây dựng cho nhiều ngôn ngữ và có mối liên hệ chặt chẽ với nhau. Cụ thể, dự án VerbNet đã mô tả 6,791 nghĩa của các động từ, phân thành 329 lớp cha, 272 lớp con, được liên kết tới 5,649 động từ trong PropBank, 4,186 khung trong FrameNet và 4,898 liên kết nhóm trong SynSemClass Lexicon[13].

Các phần tiếp theo sẽ trình bày chi tiết về các tài nguyên từ vựng và các kho văn bản có chú giải ngữ pháp, ngữ nghĩa.

1.4.1. Tài nguyên từ vựng

Các tài nguyên từ vựng (*lexical resources*) là các kho dữ liệu chứa thông tin về từ ngữ của một ngôn ngữ, bao gồm các thuộc tính ngữ pháp, ngữ nghĩa, ngữ dụng của từ. Những tài nguyên này được sử dụng rộng rãi trong các ứng dụng NLP, ngôn ngữ học, và các hệ thống thông tin ngôn ngữ để giúp máy tính hiểu và xử lý ngôn ngữ của con người. Trong phần này sẽ trình bày về các mạng từ có thông tin về ngữ nghĩa và cú pháp như WordNet, VerbNet, FrameNet và các từ điển (VCL).

1.4.1.1. WordNet

WordNet là một cơ sở dữ liệu từ vựng rất lớn, gồm các danh từ, động từ, tính từ, trạng từ, ... được nhóm thành tập hợp các lớp đồng nghĩa về nhận thức (*synsets*), mỗi lớp thể hiện một khái niệm riêng biệt. Các tập hợp được liên kết với nhau bằng các quan hệ khái niệm - ngữ nghĩa và từ vựng. Các quan hệ ngữ nghĩa gồm có: đồng nghĩa, trái nghĩa, thượng - hạ vị, quan hệ bộ phận - toàn bộ, các quan hệ nhân quả. WordNet là một trong những cơ sở dữ liệu hữu ích

[13] `https://uvi.colorado.edu/`

cho các ứng dụng của ngôn ngữ học tính toán và xử lý ngôn ngữ tự nhiên.

WordNet có nhiều phiên bản cho các ngôn ngữ khác nhau, phổ biến nhất là WordNet tiếng Anh [86] [41]. Tính đến phiên bản 3.0, WordNet tiếng Anh có khoảng 117,000 danh từ, 11,400 động từ, 22,000 tính từ và 4,600 trạng từ. Hiện tại, WordNet đã được phát triển cho các ngôn ngữ khác như tiếng Pháp [103] (109,447 từ), tiếng Trung Quốc [120] (61,536 từ) và tiếng Việt [96] (9,615 từ).

1.4.1.2. VerbNet

VerbNet [68] là nguồn tài nguyên động từ, trong đó các động từ được xếp thành các lớp khác nhau dựa vào thuộc tính ngữ pháp và ngữ nghĩa của chính các động từ đó.

VerbNet được phát triển cho nhiều ngôn ngữ khác nhau. Tiêu biểu nhất là VerbNet tiếng Anh gồm hơn 5,800 động từ, được chia thành 270 nhóm, theo cách phân loại động từ của Beth Levin [72]. Dựa vào cách phân loại của Levin, các tác giả đã tổ chức các động từ theo thứ bậc để đảm bảo rằng tất cả các thành viên đều có các thuộc tính ngữ nghĩa và cú pháp chung. Mỗi lớp trong hệ thống phân cấp đều có các đặc trưng có thể mở rộng bởi tập các động từ và được mô tả bằng các khung cú pháp, các vị từ ngữ nghĩa và một danh sách các tham tố động từ điển hình.

Mỗi lớp động từ trong VerbNet được mô tả một cách đầy đủ bởi các thông tin như: các vai nghĩa chính (*thematic roles*), các lựa chọn được hạn chế cho các tham tố *(selectional restrictions on the arguments)*, khung (*frame*) mô tả chức năng cú pháp, ngữ nghĩa, các ví dụ đi kèm.

Class Hit-18.1			
Roles and Restrictions: Agent[+int_control] Patient[+concrete] Instrument[+concrete]			
Members: bang, bash, hit, kick, ...			
Frames:			
Name	Example	Syntax	Semantics
Basic Transitive	Paula hit the ball	Agent V Patient	cause(Agent, E)manner(during(E), directed motion, Agent) !contact(during(E), Agent, Patient) manner(end(E),forceful, Agent) contact(end(E), Agent, Patient)

Hình 1.8: Ví dụ về lớp động từ Hit-18.1 trong VerbNet.

Ngoài ra, VerbNet cũng đã được phát triển cho các ngôn ngữ khác như tiếng Pháp[14] [33], tiếng Ả Rập[15], tiếng Tây Ban Nha[16], ... Một loạt các nhiệm vụ ngôn ngữ tự nhiên bao gồm dịch máy, tạo ngôn ngữ [37], phân loại tài liệu [70], phân tích và gán nhãn vai nghĩa [108, 44], xác định và xây dựng các lớp nghĩa của từ vựng [18, 62], xây dựng các đồ thị khái niệm [49], ... đã được phát triển từ việc sử dụng các lớp động từ trong VerbNet.

1.4.1.3. FrameNet

FrameNet[17] [13] là một dự án xây dựng một cơ sở dữ liệu từ vựng về tiếng Anh mà cả người và máy đều có thể đọc được, dựa trên các ví dụ chú giải về cách các từ được sử dụng trong các văn bản thực tế. FrameNet được coi là một cuốn từ điển gồm hơn 13,000 từ vựng, hầu hết đều có các ví dụ chú giải thể hiện ý nghĩa và cách sử dụng. Đối với nhà nghiên cứu về NLP, hơn 200,000 câu được chú giải thủ công được liên kết với hơn 1,200 khung ngữ nghĩa. Cơ sở dữ liệu giống FrameNet đã được xây dựng cho một số ngôn ngữ và một dự án mới đang làm việc để gióng hàng FrameNet giữa các ngôn ngữ.

Một đơn vị từ vựng (*lexical unit - LU*) là một cặp gồm một từ và ý nghĩa của nó. Thông thường, mỗi ý nghĩa của một từ đa nghĩa sẽ thuộc một khung ngữ nghĩa cụ thể, một cấu trúc khái niệm giống tập lệnh mô tả một loại tình huống, đối tượng hoặc sự kiện cùng với những người tham gia và các dụng cụ đi kèm.

Ví dụ, khung **Apply_heat** mô tả một tình huống thông thường liên quan đến *COOK*, *FOOD* và *HEATING_INSTRUMENT*, và nó sẽ thường được gợi ý bằng các từ như *bake, blanch, boil, broil, brown, simmer, steam.* FrameNet định nghĩa đây là các yếu tố của khung (*FEs*) và các từ gợi lên khung là các đơn vị từ vựng (LUs).

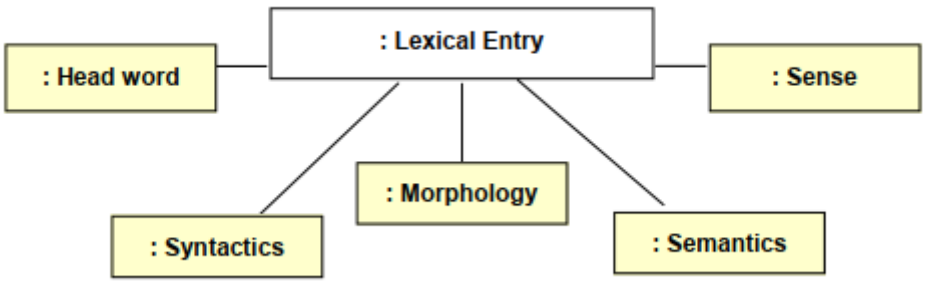
FrameNet được phát triển cho các ngôn ngữ khác như tiếng Pháp [20], tiếng Trung [128], ... FrameNet tập trung vào các tình huống và mối quan hệ ngữ nghĩa giữa các từ trong ngữ cảnh của một sự kiện hoặc hành động cụ thể, là một tài nguyên hữu ích cho các tác vụ NLP.

[14] `https://verbenet.inria.fr/`
[15] `https://github.com/JaouadMousser/Arabic-Verbnet`
[16] `https://clic.ub.edu/corpus/en/ancoranet`
[17] https://framenet.icsi.berkeley.edu/fndrupal/about

1.4.1.4. VCL

Từ điển dùng cho máy tính (*machine-readable dictionary - MRD*) là một loại từ điển được thiết kế để máy tính có thể đọc, xử lý và sử dụng cho các tác vụ xử lý ngôn ngữ tự nhiên. Khác với các từ điển truyền thống dùng cho con người, MRD được cấu trúc và lưu trữ ở định dạng mà máy tính có thể hiểu và truy xuất dễ dàng. Với các ngôn ngữ khác nhau, đều có những phiên bản từ điển dùng cho máy tính khác nhau được phát triển.

Đối với tiếng Việt, tài nguyên từ vựng lớn nhất được kể đến là Từ điển tiếng Việt dùng cho máy tính (*Vietnamese Computational Lexicon* – VCL) [94], được xây dựng theo chuẩn LMF (*Lexical Markup Framework [45]*). Mục tiêu của việc xây dựng từ điển VCL là cung cấp được cho các hệ thống xử lí ngôn ngữ tự nhiên các thông tin ngôn ngữ ở nhiều tầng bậc khác nhau như hình thái, ngữ pháp, ngữ nghĩa, tốt hơn nữa là có thể phục vụ cả các hệ thống xử lí đơn ngữ và đa ngữ. Các thông tin mô tả của một từ vựng trong VCL được thể hiện trên 3 khía cạnh: hình thái học, cú pháp học và ngữ nghĩa học như mô tả trong Hình 1.9.



Hình 1.9: Cấu trúc tổng quát của một mục từ trong VCL.

Hiện tại, VCL chứa gần 42,000 mục từ, được tổ chức thành cơ sở dữ liệu, cho phép cập nhật và thay đổi khi cần thiết.

1.4.2. Các kho văn bản có chú giải ngữ pháp, ngữ nghĩa

Các kho văn bản có chú giải (*Annotated Corpora*) là tập hợp các văn bản, câu, hoặc từ ngữ được gán thêm thông tin chú giải nhằm cung cấp ngữ cảnh, cú pháp, ngữ nghĩa hoặc thông tin khác liên quan đến xử lý ngôn ngữ tự nhiên. Chú giải có thể bao gồm từ loại, cấu trúc cú pháp (như trong kho văn bản Penn Treebank [88], Chinese Treebank [125]), Universal Dependency [89], ...), phân

tích ngữ nghĩa (PropBank [67], AMR [14]), phân tích cảm xúc ([40]), thực thể có tên (NER [75]), và nhiều loại khác. Các phần sau sẽ trình bày về một số kho văn bản có chú giải các thông tin như cú pháp thành phần, cú pháp phụ thuộc và các thông tin ngữ nghĩa.

1.4.2.1. Kho ngữ liệu cú pháp thành phần

Kho ngữ liệu cú pháp thành phần (*Constituency Parsing Corpus*) thường được gọi là Treebank, với cấu trúc gồm có 3 phần chính: Gán nhãn từ loại, phân tích cú pháp thành phần và chú giải phát âm.

Ví dụ về một câu được phân tích cú pháp và gán nhãn từ loại trong Treebank tiếng Anh như sau:

```
((S (NP Martin Maritetta Corp.) was (VP given (NP a $29.9 million
Air Force contract (PP for (NP low-altitude navigation and targeting
equipment))))).)
```

Trong đó:

- S (câu), NP (cụm danh từ), VP (cụm động từ), PP (cụm giới từ) là các nhãn cú pháp thành phần của câu.

Penn Treebank [88] là kho ngữ liệu cú pháp thành phần đầu tiên được xây dựng cho tiếng Anh, bao gồm việc gán nhãn từ loại và việc phân tích cú pháp thành phần theo dạng đặt ngoặc cho các văn bản (tạp chí Wall Street và kho ngữ liệu Brown). Kho cú pháp thành phần này bao gồm hơn 7 triệu từ của các văn bản được gán nhãn từ loại, 3 triệu từ được phân tích cú pháp, hơn 2 triệu từ được phân tích cú pháp đối với cấu trúc vị ngữ và hơn 1.6 triệu trong văn bản nói được phiên âm sang các ngôn ngữ khác. Sau đó, một loạt các treebank khác được phát triển dựa vào Penn Treebank như Chinese Treebank [125] (780,000 từ), French Treebank [9] (21,550 câu - 664,500 tokens), VietTreebank [101] (10,165 câu).

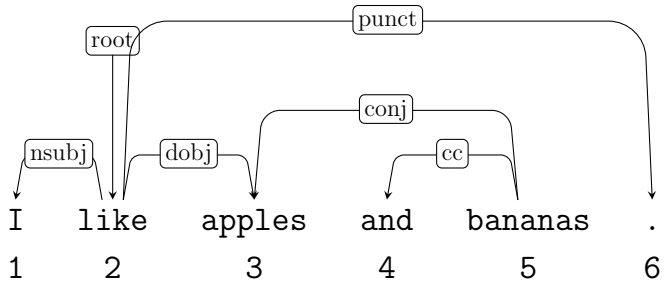
1.4.2.2. Kho ngữ liệu phân tích cú pháp phụ thuộc đa ngôn ngữ

Các kho ngữ liệu được gán nhãn quan hệ phụ thuộc đa ngôn ngữ (*Universal Dependency - UD*[18]) được xây dựng bởi nhóm nghiên cứu của trường Đại học Stanford. Đây là một dự án được phát triển dựa vào chú giải treebank cho đa ngôn ngữ, với mục tiêu tạo điều kiện thuận lợi cho sự phát triển phân tích cú pháp đa ngôn ngữ, học chéo giữa các ngôn ngữ. Mục tiêu chung của việc phát

[18]https://universaldependencies.org/

triển một bộ nhãn phụ thuộc đa ngôn ngữ là để có thể cung cấp một kho ngữ liệu chung về các nhãn, các hướng dẫn tạo điều kiện thuận lợi cho việc xây dựng những công trình tương tự đối với các ngôn ngữ khác, cho phép mở rộng đối với một ngôn ngữ mới khi cần thiết.

Ví dụ: Đồ thị phụ thuộc của câu tiếng Anh: `I like apples and bananas.` được thể hiện như trong Hình 1.10.



Hình 1.10: Đồ thị phụ thuộc của câu: I like apples and bananas.

Mỗi nhãn quan hệ trong câu trên đều thể hiện một ý nghĩa nhất định. Ví dụ: quan hệ phụ thuộc *dobj(like-2, apples-3)* có nghĩa là: *apples* là tân ngữ trực tiếp của *like*.

Tập nhãn đa ngôn ngữ đã được phát triển cho nhiều ngôn ngữ khác nhau như tiếng Anh, tiếng Pháp, tiếng Đức, tiếng Trung Quốc, ... và có nhiều nhóm nghiên cứu đóng góp kho ngữ liệu cú pháp phụ thuộc vào dự án. Một vài thông tin về các kho ngữ liệu cú pháp phụ thuộc được mô tả trong Bảng 1.1.

Bảng 1.1: Một vài kho ngữ liệu cú pháp phụ thuộc trong dự án UD.

STT.	**Ngôn ngữ**	**Số lượng treebank**	**Số từ**	**Số câu**
1	Arabic	3	1,042,096	28,402
2	Chinese	6	287,287	12,549
3	Czech	5	2,227,231	127,507
4	English	9	762,012	45,815
5	French	8	1,208,515	48,297
6	Vietnamese	1	43,783	3,000

Đối với tiếng Việt, một trong những nghiên cứu nổi bật nhất về việc xây dựng kho ngữ liệu cú pháp phụ thuộc tiếng Việt là của nhóm nghiên cứu [91]. Các tác giả đã nghiên cứu về tập nhãn quan hệ phụ thuộc đa ngôn ngữ, cùng với Viettreebank, và xây dựng một bộ nhãn cú pháp phụ thuộc cho tiếng Việt. Sau đó, khi dự án UD được phát triển từ năm 2014 [89], nhóm nghiên cứu tiếp

tục phát triển tập nhãn để tích hợp vào dự án này, phù hợp với môi trường đa ngôn ngữ. Bộ nhãn quan hệ phụ thuộc tiếng Việt gồm có 48 nhãn. Hiện tại, kho ngữ liệu cú pháp phụ thuộc tiếng Việt gồm có gần 10,000 câu được chuyển từ Viettreebank sang, trong đó có 3,000 câu đã được tích hợp vào dự án Universal Dependencies để phục vụ cho các bài toán xử lí liên ngữ.

1.4.2.3. Kho ngữ liệu có gán nhãn vai nghĩa

PropBank [67] là một kho ngữ liệu được xây dựng nhằm bổ sung thông tin ngữ nghĩa cho các động từ trong Treebank. PropBank chú thích các thành phần tham gia hành động (các tham tố) và gán cho chúng vai trò ngữ nghĩa cụ thể, như người thực hiện hành động, đối tượng bị tác động, công cụ, thời gian, ... Mỗi nghĩa khác nhau của một động từ được phân biệt bằng một mã gọi là Roleset (hay còn gọi là FrameSet ID), giúp làm rõ các cách sử dụng khác nhau của cùng một vị từ. Đặc biệt, trong những trường hợp phù hợp, các vai nghĩa trong PropBank còn được ánh xạ sang hệ thống vai nghĩa của VerbNet [78]. Trong khi PropBank sử dụng các tham số đánh số (Arg0, Arg1,...) theo từng vị từ cụ thể, thì VerbNet lại dùng các vai nghĩa có thể dùng chung giữa các vị từ tương tự.

Cụ thể:

- PropBank chú giải ngữ nghĩa cho khoảng 40,000 câu trong tập dữ liệu Penn Treebank.
- Tổng số Frameset có thể lên đến hàng ngàn, vì mỗi động từ có thể có nhiều cách sử dụng khác nhau và mỗi cách sử dụng lại có thể có một Frameset riêng biệt.
- Các vai nghĩa trong PropBank gồm có: Arg0 (tác thể), Arg1 (bị thể), Arg2 (công cụ, người hưởng lợi, thuộc tính), Arg3 (điểm bắt đầu, người hưởng lợi, thuộc tính), Arg4 (điểm kết thúc), và các vai nghĩa bổ sung ArgM (TMP - thời gian, LOC - địa điểm, MNR - cách thức, ...).

Đối với tiếng Việt, tác giả [2] cùng nhóm nghiên cứu đã đề xuất xây dựng một tập nhãn vai nghĩa dựa vào tập nhãn PropBank tiếng Anh. Mỗi câu trong kho ngữ liệu Propbank tiếng Việt có thể có vị ngữ do động từ, tính từ, số từ, danh từ hoặc các giới từ làm trung tâm. Với mỗi thành phần trung tâm vị ngữ, thực hiện gán nhãn vai nghĩa cho các thành phần chính (tham tố - Arg) quanh

trung tâm đó, và cũng gán nhãn vai nghĩa cho các thành phần phụ (*modifier* – ArgM) của nó. Mỗi vai nghĩa này có thể được chia thành các vai nghĩa mịn hơn đối với các trường hợp phân biệt tường minh giữa các vai nghĩa với nhau. Sau đó, nhóm nghiên cứu đã đề xuất một thuật toán chuyển tự động từ kho ngữ liệu thành phần Viettreebank sang kho ngữ liệu được gán nhãn vai nghĩa thô gồm hơn 10,000 câu. Kết quả là kho ngữ liệu có gán nhãn vai nghĩa cho tiếng Việt đã được xây dựng gồm 5,460 câu tiếng Việt, với 7,525 nhãn vai nghĩa.

Một vài ví dụ về các vai nghĩa được xây dựng trong Propbank tiếng Việt:

- Có lẽ **con cừu**$_{\text{Arg0}}$ đã ăn mất đoá hoa chăng?
- **Cá voi**$_{\text{Arg0-Carrier}}$ là$_{\text{rel}}$ **loài động vật có vú**$_{\text{Arg1-Attribute}}$

Tuy nhiên, kho ngữ liệu gán nhãn vai nghĩa này không thực sự giống như Propbank tiếng Anh, vì nó không được liên kết tới bất cứ tài nguyên ngôn ngữ nào khác như FrameNet, WordNet hay VerbNet, do tiếng Việt chưa có các tài nguyên ngôn ngữ này. Tuy nhiên, đây cũng là một trong những bước nghiên cứu nền tảng để có thể phát triển tiếp bài toán này, cũng như áp dụng vào một số vấn đề khác của NLP.

1.4.2.4. Kho ngữ liệu gán nhãn ngữ nghĩa trừu tượng AMR

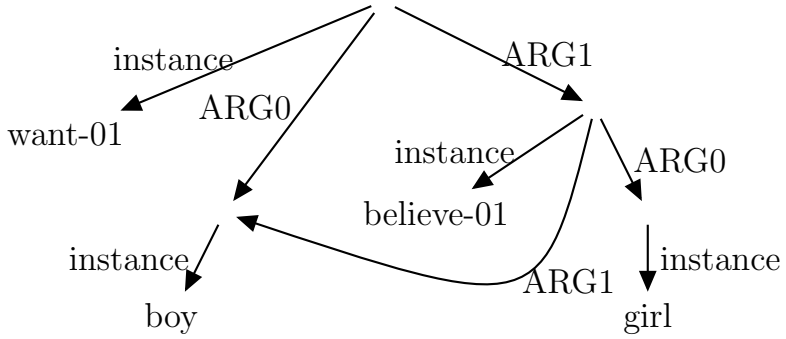
Nhóm tác giả *Laura Banarescu* cùng các cộng sự [14] đã đưa ra mô hình biểu diễn ngữ nghĩa trừu tượng (*Abstract Meaning Representation - AMR*) với mục đích xây dựng kho dữ liệu gán nhãn ngữ nghĩa logic (*sembank*) cho tiếng Anh. AMR đưa ra cách biểu diễn ngữ nghĩa cho một câu với cấu trúc đơn giản nhưng thống nhất và giữ đầy đủ thông tin ngữ nghĩa. AMR nắm bắt thông tin mô tả "ai làm gì cho ai" (*"who is doing what to whom"*) trong câu. Mỗi câu sẽ được biểu diễn bằng một đồ thị có gốc, có hướng và không có chu trình (*rooted, directed, acyclic graph*) với các nhãn trên các cung biểu diễn các quan hệ (*relation*) và các nút lá biểu diễn các khái niệm (*concept*). Các thông tin ngữ nghĩa trong kho dữ liệu AMR được nắm bắt thông qua các sự kiện, khái niệm, được mô tả như các vị từ cùng với các tham tố của nó.

AMR của một câu được xây dựng theo các nguyên tắc sau:

- Là đồ thị có gốc, được gán nhãn dễ đọc, dễ hiểu và các lập trình viên dễ duyệt đồ thị này.

- AMR mở rộng việc sử dụng khung PropBank, hướng tới sự trừu tượng hóa từ những đặc tính cú pháp riêng. Các câu có cùng ngữ nghĩa miêu tả sẽ có cùng biểu diễn AMR.
- AMR là bất khả lượng về việc lấy ra ý nghĩa từ một chuỗi hoặc ngược lại. Trong việc chuyển câu sang AMR, sẽ không có một chuỗi các quy tắc cụ thể được đưa ra để áp dụng.
- Mô hình AMR khi được thiết kế là riêng cho tiếng Anh, nó không quan tâm đến tính liên ngữ.

Ví dụ về một đồ thị trong kho dữ liệu AMR được thể hiện trong Hình 1.11.



Hình 1.11: Ví dụ về một câu được phân tích AMR

Trong ví dụ trên, các thông tin ngữ nghĩa được biểu diễn như sau: trong câu này, có một sự kiện là sự mong muốn (*wanting event*) của 2 tham tố là ARG0 và ARG1, trong đó ARG0 (người có mong muốn) là một chàng trai (*boy*), và ARG1 (điều được mong muốn) là sự kiện tin tưởng (*believing event*). Tương tự như sự kiện mong muốn, sự kiện tin tưởng cũng có 2 tham tố ARG0 và ARG1, trong đó ARG0 là người có niềm tin (*believer*) có thể hiện là một cô gái (*girl*), và ARG1 là điều được tin tưởng có thể hiện là chàng trai (*boy*) đã nói đến ở trên. Ở đây, từ chàng trai (*boy*) đóng 2 vai trò: thứ nhất là ARG0 của (*want-01*) và ARG1 của (*believe-01*).

Kho ngữ liệu ngữ nghĩa trừu tượng AMR hiện có nhiều phiên bản khác nhau. AMR cho tiếng Anh có ba phiên bản chính: AMR 1.0 (39,260 câu), AMR 2.0 (59,255 câu) và AMR 3.0 lớn nhất gồm 76,572 câu từ các nguồn đa dạng như tin tức, tiểu thuyết và bài phát biểu chính trị. Các ngôn ngữ khác như tiếng Tây Ban Nha [121], tiếng Hàn [24], tiếng Trung [73], ... cũng đã xây dựng các kho AMR với số lượng từ 1,000 đến 5,000 câu.

AMR được sử dụng trong các lĩnh vực như dịch máy, hỏi đáp và phân tích ngữ nghĩa, và đang tiếp tục phát triển để hỗ trợ nhiều ngôn ngữ hơn. Tuy nhiên, việc tự động tạo ra AMR từ văn bản vẫn đang là một thách thức lớn đối với các hệ thống NLP.

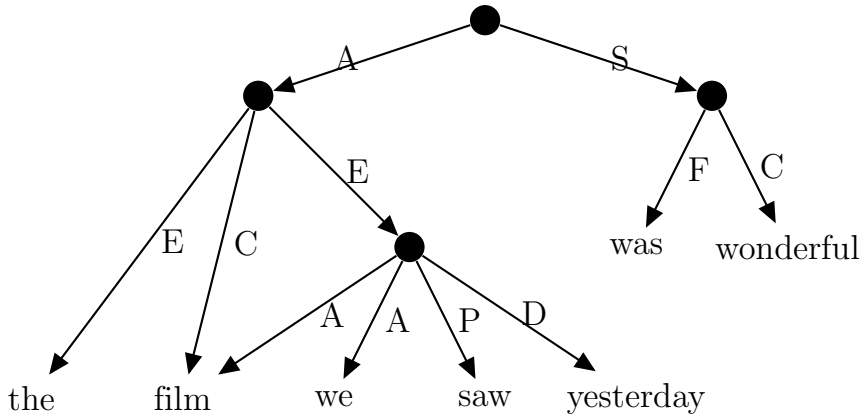
1.4.2.5. Kho ngữ liệu ngữ nghĩa UCCA

Mô hình biểu diễn ngữ nghĩa UCCA (*Universal Conceptual Cognitive Annotation [7]*) là một cách biểu diễn ngữ nghĩa nhằm chú giải sự khác biệt ngữ nghĩa và hướng tới mục đích trừu tượng hóa cấu trúc cú pháp cụ thể. UCCA bao gồm một tập hợp phân biệt ngữ nghĩa phong phú và cung cấp tất cả các thông tin ngữ nghĩa cần thiết cho một đoạn văn. UCCA có chứa một số ngữ nghĩa nhất định (*Scenes*) trong đó bao gồm: các quan hệ, cấu trúc tham tố của các động từ, danh từ, tính từ đi kèm. UCCA được xây dựng dựa vào lý thuyết ngôn ngữ cơ bản (*Basic linguistic theory - BLT*), một biểu diễn UCCA được xây dựng dựa vào các nguyên tắc cơ bản sau:

- UCCA sử dụng đồ thị có hướng không có chu trình để biểu diễn cho cấu trúc ngữ nghĩa.
- Nguyên tử đơn vị ngữ nghĩa là các nút lá của đồ thị DAG và được gọi là các terminals. Trong lớp nền tảng, terminals là các từ và các từ multi-words, mặc dù định nghĩa này có thể được mở rộng để bao gồm các hình thái tùy ý. Các nút của đồ thị được gọi là các đơn vị. Một đơn vị có thể là: 1) terminal hoặc 2) một yếu tố được coi như là một thực thể duy nhất hoặc nhận thức quan trọng.

Ví dụ: "`The film we saw yesterday was wonderful.`" được mô tả chi tiết trong Hình 1.12.

UCCA có cấu trúc nhiều tầng, các tầng thêm vào có thể điều chỉnh các quan hệ đã tồn tại hoặc chú giải một số các thành phần khác trong những ngữ cảnh đã có. Kho dữ liệu có chú giải UCCA gồm có 56,980 tokens, trong 148 đoạn văn từ các bài báo từ Wikipedia tiếng Anh, có công cụ hỗ trợ chú giải UCCA. Các chú giải được thực hiện trên một đoạn, trung bình từ 300-400 tokens, được sửa thủ công trước khi đưa vào lưu trữ. Mỗi đoạn chứa trung bình 450 đơn vị và 42.2 ngữ cảnh. Quá trình gán nhãn có 4 nhà chú giải có trình độ ngôn ngữ khác nhau, được đào tạo từ 30-40 giờ trước khi tiến hành gán nhãn.



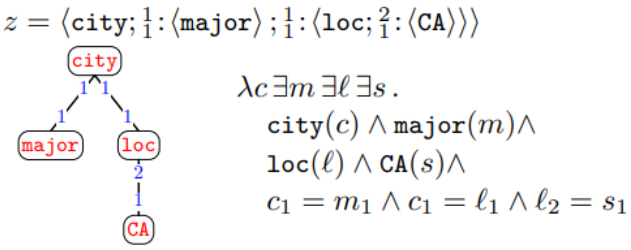
Hình 1.12: Ví dụ về một câu được phân tích UCCA

1.4.2.6. Kho ngữ liệu ngữ nghĩa dựa vào cú pháp phụ thuộc

Mô hình biểu diễn ngữ nghĩa dựa vào cú pháp phụ thuộc DCS (*Dependency based Compositional Semantics [76, 99]*) là một trong những cách biểu diễn ngữ nghĩa dựa vào cú pháp phụ thuộc, được nhóm tác giả *Percy Liang* và cộng sự phát triển vào năm 2013. Với mục tiêu xây dựng một hệ thống trả lời các câu hỏi từ ngôn ngữ tự nhiên bằng cách trình bày ngữ nghĩa của nó dưới dạng một hình thức logic và tính toán các câu trả lời từ một cơ sở dữ liệu có cấu trúc của các sự kiện. Các tác giả đã xây dựng một công cụ phân tích ngữ nghĩa từ các cặp câu hỏi và câu trả lời, trong đó, hình thức logic được mô hình hóa như một biến tiềm ẩn. Các tác giả thấy rằng, các biểu diễn cú pháp hiện có không giải quyết được các hiện tượng ngôn ngữ sâu như: định lượng, phủ định hay so sánh nhất.

Mô hình này có một câu hỏi là x, được ánh xạ tới một dạng logic ẩn z, sau đó sẽ được đánh giá có quan hệ với một từ w (cơ sở dữ liệu thực tế), và tạo ra một câu trả lời y. Các tác giả biểu diễn các dạng logic z như một cây được gán nhãn, tự động tạo ra từ cặp (x, y). Các tác giả mong muốn rằng chỉ tạo ra dạng logic tiềm ẩn z và tham số θ với chỉ cặp (x, y) được đưa ra, như thế sẽ tốt hơn rất nhiều so với việc tạo ra cặp (x, z). Vì thế các tác giả đã đề xuất một biểu diễn ngữ nghĩa mới, được gọi là ngữ nghĩa thành phần dựa vào cú pháp phụ thuộc (DCS). Ví dụ:

Các tác giả đã thực nghiệm trên hai bộ dữ liệu chuẩn là GEO và JOBS. Tập dữ liệu GEO gồm có 880 câu hỏi về địa lý và tập JOBS gồm có 640 câu hỏi về công việc.



Hình 1.13: Biểu diễn ngữ nghĩa dạng đồ thị DCS

1.4.2.7. Kho ngữ liệu ngữ nghĩa Groningen

Kho dữ liệu ngữ nghĩa Groningen (GMB) [58] bao gồm các văn bản tiếng Anh như các bài báo, tạp chí, ... với các biểu diễn cú pháp và biểu diễn ngữ nghĩa tương ứng. GMB được phát triển bởi nhóm nghiên cứu của trường Đại học Groningen, có phiên bản đa ngôn ngữ. Trong GMB, các hiện tượng được tích hợp với nhau, thay vì biểu thị từng hiện tượng đơn lẻ. Chính vì thế, sử dụng kho dữ liệu GMB có thể giải thích sự phụ thuộc giữa các hiện tượng ngôn ngữ, bao gồm các ngữ cảnh của các từ, vai trò chủ đề, phạm vi định lượng, các khía cạnh khác nhau. GMB sẽ chú giải ngữ nghĩa cho toàn văn bản chứ không phải là từng câu riêng biệt.

GMB được xây dựng bằng cách sử dụng bootstrapping và các công cụ NLP tiên tiến nhất (bao gồm các công cụ C&C và Boxer) để tạo ra một xấp xỉ hợp lí cho các chú thích chuẩn. Thông tin về các công cụ được sử dụng trong GMB:

- Elephant là một công cụ thống kê cho các từ và phân đoạn các câu được sử dụng trong GMB.
- C&C cung cấp một chú giải tự động của GMB bao gồm: gán nhãn từ loại, gán nhãn thực thể có tên, và phân tích cú pháp.
- Morpha được sử dụng để phân tích hình thái từ.
- Boxer cung cấp một biểu diễn ngữ nghĩa diễn ngôn (DRS) trên mỗi cây cú pháp và một bản demo trực tuyến của C&C và Boxer.

Từ khi phát hành tới giờ, các chú giải trong GMB được sửa và tinh chỉnh bởi các nhà nghiên cứu theo hai cách chính: Các chuyên gia trực tiếp chỉnh sửa trong GMB thông qua các trình duyệt và những người không phải là chuyên gia sẽ tham gia một trò chơi có tên là *Wordrobe* để làm giàu kho dữ liệu này. Lý thuyết được sử dụng cho các chú thích ngữ nghĩa trong GMB dựa vào lý thuyết

biểu diễn diễn ngôn (*Discourse Representation Theory – DRT*) và VerbNet cho các vai nghĩa, các biến thể của việc phân loại thực thể có tên, WordNet cũng được sử dụng cho các ngữ cảnh và phân đoạn DRT cho các quan hệ hùng biện. Các thành phần chính trong GMB bao gồm: Penn Treebank (gán nhãn từ loại), các lớp Animacy, các loại thực thể có tên, các ngữ cảnh WordNet, các vai nghĩa VerbNet, các phân hoạch cú pháp, các loại ngữ nghĩa trong DRT và các quan hệ hùng biện SDRT.

Phiên bản đầu tiên của GMB gồm hơn 1,000 văn bản với 4,239 câu và 82,752 từ[19]. Phiên bản cuối cùng gồm 10,000 văn bản với hơn 1 triệu từ loại.

1.5. Kết luận chương 1

Chương này đã tập trung làm rõ những khía cạnh nền tảng trong lĩnh vực xử lý ngôn ngữ tự nhiên là cú pháp và ngữ nghĩa. Trước hết, các định nghĩa cùng phương pháp phân tích cú pháp (thành phần và phụ thuộc) – đã được trình bày như là những kỹ thuật cốt lõi nhằm xác định cấu trúc của câu và các mối quan hệ giữa các thành tố trong câu. Việc nắm vững các phương pháp phân tích này không chỉ giúp máy tính hiểu rõ hơn về cấu trúc ngôn ngữ mà còn đóng vai trò quan trọng trong nhiều tác vụ nâng cao như dịch máy, trích xuất thông tin hay tổng hợp văn bản.

Tiếp theo, chương cũng đề cập đến một số vấn đề cơ bản trong xây dựng ngữ liệu – một bước không thể thiếu trong quá trình phát triển các hệ thống xử lý ngôn ngữ. Việc thiết kế và thu thập ngữ liệu đảm bảo tính đa dạng, chất lượng và khả năng phản ánh đặc trưng ngôn ngữ một cách chính xác là điều kiện tiên quyết để các mô hình học máy, học sâu có thể học được những đặc điểm hữu ích từ dữ liệu.

Cuối cùng, chương đã giới thiệu các loại tài nguyên ngôn ngữ quan trọng, bao gồm tài nguyên từ vựng và các kho văn bản. Những tài nguyên này không chỉ hỗ trợ trong việc huấn luyện và đánh giá mô hình mà còn là cơ sở để phân tích ngôn ngữ một cách hiệu quả và hệ thống hơn.

Các nội dung trong chương là cơ sở lý thuyết quan trọng cho những nghiên cứu ở các chương sau, hướng tới việc xây dựng các kho ngữ liệu và những hệ thống xử lý cú pháp, ngữ nghĩa ngày càng chính xác và hiệu quả hơn cho tiếng Việt.

[19] `http://gmb.let.rug.nl/`

Chương 2
XÂY DỰNG TÀI NGUYÊN VÀ CÔNG CỤ CHÚ GIẢI NGỮ PHÁP TIẾNG VIỆT

Mục tiêu xuyên suốt của luận án là xây dựng các kho ngữ liệu và công cụ phân tích cú pháp, ngữ nghĩa chia sẻ cho cộng đồng nghiên cứu xử lí ngôn ngữ tiếng Việt. Chương này tập trung vào vấn đề phân tích cú pháp tiếng Việt là mô hình cú pháp phụ thuộc và cú pháp thành phần. Phần 2.1 mô tả về việc xây dựng kho ngữ liệu cú pháp phụ thuộc cho tiếng Việt theo quy trình chuẩn hoá, thử nghiệm một số mô hình phân tích cú pháp phụ thuộc tiên tiến nhất, đánh giá kết quả đạt được và đưa ra một số thảo luận. Phần 2.2 trình bày về việc cập nhật kho văn bản có gán nhãn cú pháp thành phần Viettreebank đã có sẵn, nhằm đáp ứng tiêu chí chuẩn hoá tài nguyên ngôn ngữ. Các mô hình phân tích cú pháp thành phần cũng được khảo sát trong mục này. Tiếp theo, phần 2.3 mô tả việc xây dựng thuật toán chuyển đổi giữa hai kho ngữ liệu.

Kết quả của chương này được công bố trong các công trình [P2, P6, P8, P10] trong "Danh mục công trình công bố" của luận án.

2.1. Kho ngữ liệu phân tích cú pháp phụ thuộc cho tiếng Việt

Đối với bài toán phân tích cú pháp phụ thuộc, một số kho ngữ liệu gán nhãn phụ thuộc tiếng Việt đã được các nhóm nghiên cứu xây dựng và phát triển như:

1. Năm 2008, tác giả Nguyễn Lê Minh cùng cộng sự [3] đã xây dựng 450 câu gán nhãn cú pháp phụ thuộc. Tuy nhiên, nhóm tác giả không trình bày chi tiết về tập nhãn cú pháp phụ thuộc đã xây dựng và sử dụng.

2. Năm 2014, tác giả Nguyễn Quốc Đạt cùng cộng sự [29] cũng sử dụng kho ngữ liệu gán nhãn thành phần để phát triển thuật toán chuyển đổi và xây dựng kho ngữ liệu gán nhãn phụ thuộc, với tập nhãn phụ thuộc tổng quát gồm 33 nhãn.

3. Năm 2017, tác giả Nguyễn Kiêm Hiếu [65] đã xây dựng kho ngữ liệu gán nhãn phụ thuộc thủ công với khoảng 6,900 câu tiếng Việt, sử dụng 27 nhãn

phụ thuộc được xây dựng dựa trên nhãn phụ thuộc của Stanford xây dựng [83].

4. Năm 2013-2020, nhóm tác giả Nguyễn Thị Lương [91, 92] đã xây dựng kho ngữ liệu cú pháp phụ thuộc tiếng Việt, dựa trên bộ nhãn quan hệ phụ thuộc của Đại học Stanford [83]. Các tác giả đã xây dựng các thuật toán chuyển đổi từ kho ngữ liệu cú pháp thành phần Viettreebank [101] và gán nhãn cú pháp phụ thuộc cho dữ liệu đã chuyển đổi. Khi dự án cú pháp phụ thuộc phổ quát UD ra đời vào năm 2014 với mục tiêu phát triển các kho ngữ liệu cú pháp phụ thuộc đa ngôn ngữ [89], nhóm nghiên cứu đã tiếp tục mở rộng và điều chỉnh tập nhãn để phù hợp với những đối sánh phụ thuộc đa ngữ. Đến năm 2018, kho ngữ liệu cú pháp phụ thuộc tiếng Việt có 10,165 câu, trong đó 3,000 câu (với độ dài < 25) đã được tích hợp vào dự án UD.

Có thể thấy rằng, việc xây dựng kho ngữ liệu cú pháp phụ thuộc đã thu hút sự quan tâm và phát triển từ nhiều nhóm nghiên cứu. Tuy nhiên, một vấn đề nổi bật là các bộ nhãn phụ thuộc hiện vẫn chưa được thống nhất, khi mỗi nhóm sử dụng một bộ nhãn riêng và thiếu các tiêu chuẩn đánh giá rõ ràng về tính hiệu quả, độ phù hợp và ánh xạ đa ngôn ngữ. Bên cạnh đó, các nghiên cứu thường chưa cung cấp chi tiết về quá trình xây dựng hệ thống nhãn, quy trình gán nhãn dữ liệu, cũng như cách đánh giá chất lượng của kho ngữ liệu.

Với mục tiêu xây dựng tài nguyên cú pháp phụ thuộc chuẩn hoá, theo đúng quy trình gán nhãn, luận án thực hiện cập nhật và xây dựng lại tập nhãn cú pháp phụ thuộc tiếng Việt dựa vào các công trình của tác giả Nguyễn Thị Lương và cộng sự. Những cập nhật này được thực hiện theo những thay đổi của UD phiên bản 2.11 [90] trong cách tách từ, gán nhãn từ loại, xác định các quan hệ phụ thuộc. Tập nhãn phụ thuộc cũng có nhiều thay đổi như một số nhãn được thay thế hoặc loại bỏ, một số nhãn mới được thêm vào để có thể nắm bắt quan hệ phụ thuộc chung cho đa ngôn ngữ. Sau đó, luận án thực hiện thu thập thêm các dữ liệu trên nhiều miền khác nhau, xây dựng thuật toán chuyển từ cú pháp thành phần sang cú pháp phụ thuộc và gán nhãn thủ công kho ngữ liệu đã chuyển đổi. Về công cụ phân tích cú pháp phụ thuộc, một số thuật toán đã được thử nghiệm, đánh giá và thảo luận kết quả đạt được.

2.1.1. Xây dựng tập nhãn cú pháp phụ thuộc tiếng Việt

Luận án thực hiện xây dựng và cập nhật lại tập nhãn phụ thuộc này dựa vào phiên bản UD phiên bản 2.11 [90]. Tập nhãn hiện tại của tiếng Việt gồm 84 nhãn (trong đó có 40 nhãn chính và 44 nhãn con). Việc bổ sung các nhãn mới được thực hiện theo các nguyên tắc và sự phân loại của UD, chủ yếu là thêm vào các nhãn con (*sub-type*) nhằm phản ánh chính xác hơn các đặc trưng ngữ pháp và cú pháp riêng biệt của tiếng Việt, như các yếu tố biểu thị thời gian, danh từ hoá động từ, hệ từ "là" trong câu hoặc các tổ hợp từ. Các nhãn phụ thuộc tiếng Việt được phân loại theo các nhóm dựa trên phương pháp phân loại của dự án phụ thuộc phổ quát UD, với tiêu chí dựa trên vai trò ngữ pháp, mối quan hệ cú pháp, và tính liên ngữ, đồng thời đảm bảo tính thống nhất nhưng vẫn linh hoạt để phù hợp với đặc thù của từng ngôn ngữ. Cụ thể, các nhóm nhãn phụ thuộc tiếng Việt gồm có:

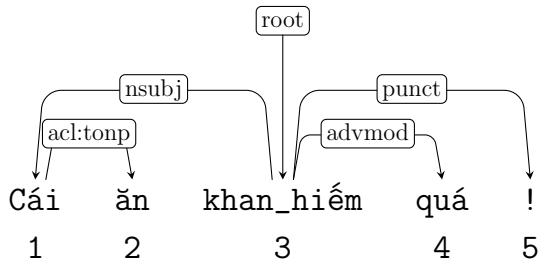
- Những phụ thuộc cốt lõi của vị từ: chủ ngữ danh từ (`nsubj`), chủ ngữ mệnh đề (`csubj`), tân ngữ trực tiếp (`dobj`), tân ngữ gián tiếp (`iobj`), ...
- Những phụ thuộc không cốt lõi của vị từ: bổ ngữ danh từ (`nmod`), trạng ngữ mệnh đề (`advcl`), phụ từ bổ nghĩa (`advmod`), phụ từ phủ định (`neg`).
- Những phụ thuộc mệnh đề đặc biệt: thành phần xưng hô (`vocative`), trợ động từ (`aux`), hệ từ (`cop`), ...
- Những phụ thuộc danh từ: định ngữ số lượng (`nummod`), định ngữ là mệnh đề (`acl`), ...
- Những phụ thuộc về các từ không thể phân tích và các nhóm từ ghép: từ ghép (`compound`), thành ngữ (`mwe`), ...
- Những phụ thuộc về sự liên hợp: quan hệ liên hợp (`conj`), liên từ đẳng lập (`cc`), dấu câu (`punct`).
- Những phụ thuộc về sở hữu, các giới từ, hoặc các trường hợp đặc biệt được đánh dấu: giới từ trước danh ngữ (`case`).
- Những phụ thuộc về các thành phần tham gia: quan hệ liệt kê (`list`), thành phần đẳng lập (`parataxis`), ...
- Và những phụ thuộc khác: gốc (`root`), ...

Một số nhãn phụ thuộc cụ thể đã được thêm vào để biểu thị các trường hợp đặc biệt của tiếng Việt.

Tiếng Việt có nhiều đặc điểm ngữ pháp riêng biệt cần được thể hiện rõ trong các hệ thống phân tích ngữ pháp. Để xử lý chính xác các cấu trúc đặc thù này, một số nhãn phụ thuộc đã được thêm vào. Những nhãn này giúp biểu thị các hiện tượng cụ thể như danh từ hóa động từ, chủ ngữ là động từ hoặc tính từ, danh từ chỉ loại, phân biệt cụ thể các bổ ngữ, và tổ hợp từ.

1. *acl:tonp*: Danh từ hoá động từ.

   Hiện tượng danh từ hoá là việc dùng một từ không phải là danh từ (như động từ, tính từ, trạng từ) làm thành một danh từ hoặc trung tâm ngữ cho cụm danh từ. Hiện tượng này xảy ra trong nhiều ngôn ngữ. Trong tiếng Anh, việc danh từ hoá được thực hiện bằng cách thêm các hậu tố như *-tion*, *-ment*, *-ity*, ... vào động từ hoặc các gốc từ để tạo thành danh từ. Ngoài ra, cũng có một số trường hợp sử dụng dạng động từ nguyên thể hoặc V-ing làm danh từ. Khi đó, các quan hệ danh từ hoá được sử dụng như `mark` hoặc `csubj` (nếu động từ nguyên thể làm chủ ngữ của câu). Đối với tiếng Trung, tương tự như tiếng Việt, hiện tượng danh từ hoá động từ xảy ra theo cách khác, do đặc điểm ngôn ngữ không biến hình. Tiếng Việt và tiếng Trung sẽ thêm các từ chỉ loại hoặc từ ngữ pháp trước các động từ hoặc có thể dùng trực tiếp động từ như danh từ. Với trường hợp đầu tiên, trong tiếng Việt, động từ được danh từ hóa bằng một danh từ chỉ loại đứng trước nó như "cái", "sự", "việc" ... Khi đó, sẽ sử dụng quan hệ `acl:tonp` để biểu thị trường hợp đặc biệt này, ví dụ như trong Hình 2.1.



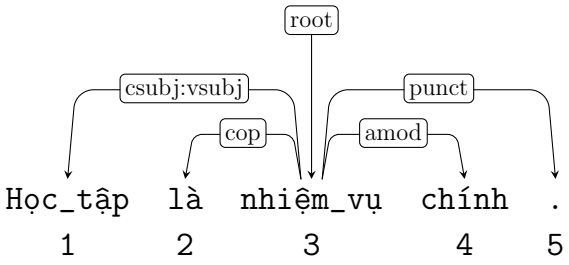
Hình 2.1: Ví dụ về nhãn "acl:tonp".

2. *csubj:vsubj*: Động từ là chủ ngữ của một câu.

   Trong tiếng Anh, khi động từ được danh từ hoá bằng cách thêm vào các

hậu tố, thì chủ ngữ khi đó sẽ là danh từ, sử dụng nhãn `nsubj`. Khi động từ nguyên thể đóng vai trò là chủ ngữ trong câu, sẽ được sử dụng nhãn `csubj`.

Đối với tiếng Việt, ngoài danh từ thì động từ và tính từ cũng đóng vai trò làm chủ ngữ trong câu. Khi động từ làm chủ ngữ của câu, thường sẽ được danh từ hoá hoặc để nguyên thể, khi đó nhãn `csubj:vsubj` được thêm vào để thể hiện điều này, ví dụ như trong Hình 2.2.

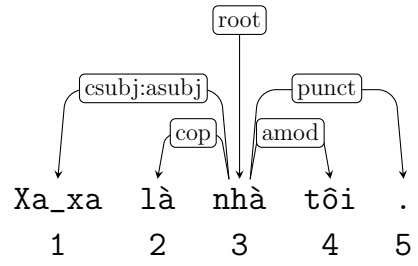


Hình 2.2: Ví dụ về nhãn "csubj:vsubj".

3. *csubj:asubj*: Tính từ là chủ ngữ của một câu.

   Trong tiếng Anh, tính từ hiếm khi đóng vai trò là một chủ ngữ độc lập, nhưng có thể xuất hiện như một danh từ hoá, bằng cách thêm các mạo từ (*the*, *this*, *these*, ...) vào trước. Khi đó, quan hệ chủ ngữ sẽ được sử dụng tương tự như cụm danh từ, là nhãn `nsubj`.

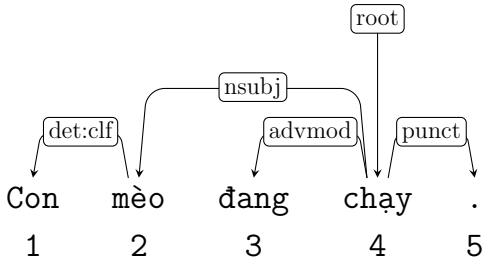
   Đối với tiếng Việt, giới từ có thể là chủ ngữ của câu. Vì thế, nhãn con `csubj:asubj` để thể hiện điều này, ví dụ như trong Hình 2.3.



Hình 2.3: Ví dụ về nhãn "csubj:asubj".

4. *det:clf*: Danh từ chỉ loại là đặc trưng nổi bật của các ngôn ngữ Đông Á (tiếng Việt, tiếng Trung, tiếng Nhật, tiếng Hàn, tiếng Thái). Trong tập nhãn UD, các danh từ chỉ loại khi đi kèm với số từ sẽ được sử dụng nhãn `clf`. Tuy nhiên, trong tiếng Việt, khi danh từ chỉ loại không xuất hiện với

số từ hoặc đại từ xác định, danh từ chỉ loại có cùng đặc tính như từ hạn định. Do đó, mối quan hệ của danh từ phân loại với danh từ đứng đầu là *det:clf*, ví dụ như trong Hình 2.4.



Hình 2.4: Ví dụ về nhãn "det:clf".

Ngoài ra, một số nhãn con được thêm vào để phân biệt các loại bổ ngữ như: trả lời cho câu hỏi "với ai", "về chuyện gì", bổ ngữ danh từ cho tính từ, phó từ, các bổ ngữ là chủ thể trong cấu trúc bị động hoặc là đích đến của động từ trao tặng.

Tất cả các nhãn trong tập nhãn phụ thuộc tiếng Việt đều được định nghĩa chi tiết kèm theo các ví dụ cụ thể trong Tài liệu hướng dẫn gán nhãn[1]. Chi tiết so sánh và ánh xạ tới các tập nhãn UD được mô tả trong Phụ lục 4.4.

2.1.2. Kho ngữ liệu cú pháp phụ thuộc tiếng Việt

Sau khi hoàn thiện tập nhãn cú pháp phụ thuộc cho tiếng Việt, kho ngữ liệu cú pháp phụ thuộc tiếng Việt được xây dựng dựa vào các bước sau:

- Thu thập và tiền xử lí dữ liệu.
- Sử dụng thuật toán chuyển tự động từ kho ngữ liệu cú pháp thành phần Viettreebank sang kho ngữ liệu cú pháp phụ thuộc (thuật toán được mô tả cụ thể trong mục 2.3) để tiết kiệm thời gian gán nhãn của các chuyên gia.
- Đào tạo các chuyên gia gán nhãn với tập nhãn cú pháp phụ thuộc mới, thực hiện chỉnh sửa và gán nhãn thủ công trên toàn bộ tập dữ liệu đã chuyển.
- Tính toán độ tương đồng, đưa ra các thảo luận để tinh chỉnh và chuẩn hoá dữ liệu cuối cùng.

[1] `https://github.com/vietnamesedp/Thesis/tree/main/DependencyParsing/Guidelines`

Việc thu thập dữ liệu được thực hiện dựa vào các nghiên cứu đã có, chủ yếu là trích rút từ kho ngữ liệu cú pháp thành phần Viettreebank. Sau đó, để mở rộng miền nghiên cứu, dữ liệu cũng được thu thập từ nguồn khác như văn bản văn học (tiểu thuyết Hoàng tử bé), dữ liệu mạng xã hội (đánh giá về các nhà hàng, khách sạn). Dữ liệu sau khi thu thập được tiền xử lí, loại bỏ các câu trùng nhau, xử lí lỗi chính tả, tách từ và gán nhãn từ loại. Cụ thể, các dữ liệu đã thu thập và tiền xử lí gồm có:

- Dữ liệu huấn luyện: gồm 8,152 câu trong đó:
  - 6,480 câu từ Viettreebank [101].
  - 1,570 câu từ kho "Hoàng tử bé" tiếng Việt[2].
  - 100 câu đánh giá về khách sạn, nhà hàng (dữ liệu mạng xã hội) [95].
- Dữ liệu kiểm thử gồm 2 bộ:
  - Dữ liệu kiểm thử VLSP 2020: gồm 1,123 câu (906 câu từ Viettreebank và 217 câu được lấy ngẫu nhiên từ *VnExpress*[3]).
  - Dữ liệu kiểm thử mới: gồm 573 câu từ các trang mạng xã hội và dữ liệu thực tế.

Quá trình gán nhãn dữ liệu được thực hiện bởi ba nhà ngôn ngữ học, được đào tạo trong vòng hai tháng và thực hiện gán nhãn hai vòng độc lập. Việc đào tạo được thực hiện để đảm bảo rằng các nhà ngôn ngữ học sẽ hiểu rõ các nhãn phụ thuộc, các đặc trưng của tiếng Việt và những thay đổi trong tập nhãn để biểu thị các đặc trưng này. Vòng đầu tiên, mỗi chuyên gia sẽ được phân công một gói dữ liệu và sau đó thực hiện gán nhãn chéo cho vòng sau. Công cụ Inception[4] được cài đặt và chi tiết hoá theo tập nhãn tiếng Việt, hỗ trợ cho việc gán nhãn dữ liệu theo đúng định dạng. Bảng 2.1 thể hiện độ đồng thuận giữa các cặp chuyên gia (tính bằng độ đo F_1).

Có thể thấy rằng, mức độ đồng thuận giữa các cặp chuyên gia đạt tỷ lệ cao (trên 91%), điều này phản ánh sự cẩn thận và tỉ mỉ trong quá trình xây dựng kho ngữ liệu cú pháp phụ thuộc, đảm bảo tính tin cậy. Kết quả này không chỉ minh chứng cho chất lượng trong việc thu thập, xử lý và gán nhãn dữ liệu, mà

[2] `https://www.informatik.uni-leipzig.de/~duc/sach/prince/viet/`
[3] `https://vnexpress.net/`
[4] `https://inception-project.github.io/`

Bảng 2.1: Độ đồng thuận của ba chuyên gia gán nhãn cú pháp phụ thuộc.

Các cặp chuyên gia	**Độ đồng thuận (F_1)**
Ano1-Ano2	92.74%
Ano1-Ano3	89.98%
Ano2-Ano3	92.53%
Trung bình	91.75%

còn tạo nền tảng vững chắc cho một kho ngữ liệu cú pháp phụ thuộc chuẩn, phục vụ cho các nghiên cứu và ứng dụng trong tương lai. Tuy nhiên, vẫn có một số trường hợp nhầm lẫn nhãn do những lỗi không chủ ý và một số nhãn thường gây nhầm lẫn cho các chuyên gia trong quá trình gán nhãn, chẳng hạn như các cặp nhãn: định ngữ danh từ (`nmod`) và từ ghép (`compound`), bổ ngữ mệnh đề khuyết (`xcomp`) và bổ ngữ mệnh đề (`ccomp`), cũng như một số loại con của nhãn từ ghép (`compound`) và tổ hợp từ (`flat`).

Một số thống kê trên bộ dữ liệu cú pháp phụ thuộc được mô tả trong Bảng 2.2.

Bảng 2.2: Một số thống kê trên bộ dữ liệu cú pháp phụ thuộc tiếng Việt.

Dữ liệu	**Số câu**	**Độ dài <30**	**Độ dài 30-50**	**Độ dài >50**	**Độ dài Trung bình**
Bộ huấn luyện Package1	5,069	4,882	159	28	14.40
Bộ huấn luyện Package2	3,083	1,942	1,005	136	24.96
Dữ liệu kiểm thử VLSP 2020	1,123	852	229	42	23.29
Dữ liệu kiểm thử mới	573	573	0	0	7.1

Bên cạnh đó, một tập con gồm 3,000 câu trong kho ngữ liệu tiếng Việt (với độ dài $<$ 25 từ) đã được trích rút và tích hợp vào dự án UD phiên bản 2.11, ngày 15/11/2022[5]. Tập 3,000 câu này được chỉnh sửa theo các thay đổi như cách tách từ, gán nhãn từ loại, các khai báo về tập nhãn từ loại, nhãn phụ thuộc, các yêu cầu về cú pháp phụ thuộc mà dự án UD đưa ra[6].

Việc tích hợp tập con gồm 3,000 câu này vào dự án UD phiên bản 2.11 không chỉ khẳng định tính toàn diện và chuẩn hoá của kho ngữ liệu tiếng Việt, mà còn mở rộng khả năng ứng dụng của nó trong các nghiên cứu ngữ nghĩa sâu hơn. Đây là một đóng góp hữu ích cho cộng đồng xử lý ngôn ngữ nói chung và tiếng Việt nói riêng, giúp dữ liệu tiếng Việt có thể được sử dụng hiệu quả trong các mô hình đa ngôn ngữ. Toàn bộ kho ngữ liệu cú pháp phụ thuộc, tài liệu hướng

[5] `https://universaldependencies.org/treebanks/vi_vtb/index.html`
[6] `https://github.com/UniversalDependencies/tools/`

dẫn gán nhãn[7] cũng được chia sẻ rộng rãi cho cộng đồng xử lý ngôn ngữ tự nhiên tiếng Việt.

2.1.3. Thử nghiệm một số thuật toán phân tích cú pháp phụ thuộc

Sau khi xây dựng thành công kho ngữ liệu cú pháp phụ thuộc chuẩn, một số thuật toán học sâu đã được thử nghiệm nhằm cải thiện hiệu năng của hệ thống phân tích cú pháp phụ thuộc tiếng Việt và kiểm định chất lượng của kho ngữ liệu đã được gán nhãn. Các mô hình này được xây dựng và mô tả chi tiết theo chuẩn của thẻ mô hình, như đã trình bày trong Mục 1.3.1. Tiếp theo, trong Mục 2.1.3.4, các lỗi của các hệ thống phân tích cú pháp phụ thuộc được phân tích trên cơ sở bộ dữ liệu đã xây dựng. Những phân tích này nhằm xác định nguyên nhân gây lỗi và đề xuất các hướng cải tiến để nâng cao hiệu năng và độ chính xác của hệ thống.

2.1.3.1. Xây dựng mô hình phân tích cú pháp phụ thuộc tiếng Việt

Dựa vào mô hình Deep bi-affine [38] và mô hình con trỏ ngăn xếp (*Stack pointer*) [85], tác giả và nhóm nghiên cứu đã phát triển 8 mô hình để phân tích cú pháp phụ thuộc tiếng Việt, được mô tả chi tiết trong Bảng 2.3. Các mô hình này thử nghiệm những biểu diễn phân bố từ khác nhau như Word2vec [119], PhoBERT-base và PhoBERT-large [31], BARTPho [93] và XLM-RoBERTa [27]. Bên cạnh đó, các phương pháp huấn luyện đa dạng kết hợp nhãn từ loại (POS) hoặc không sử dụng chúng cũng được thử nghiệm.

Các thông số cài đặt trong các mô hình gồm có: kích thước véc tơ từ, nhãn POS và kí tự lần lượt là 300, 50, 50, sử dụng 6 BiLSTM với số lớp ẩn là 500. Sử dụng thuật toán tối ưu hóa Adam với $\beta_1 = \beta_2 = 0.9$, tốc độ học $\eta = 0.001$ và hệ số dropout = 0.33.

2.1.3.2. Độ đo đánh giá

Các mô hình phân tích cú pháp phụ thuộc được đánh giá bằng độ đo LAS (*Labeled Attachment Score*). LAS được xác định bằng cách so sánh các quan hệ phụ thuộc đúng của dữ liệu chuẩn và các quan hệ phụ thuộc do hệ thống xác định. Cụ thể là:

[7] `https://github.com/vietnamesedp/Thesis/tree/main/DependencyParsing`

Bảng 2.3: Các mô hình phân tích cú pháp phụ thuộc.

STT.	**Mô hình**	**Mô tả**
1	deepbiaf	Mô hình Deep bi-affine attention, Word2vec và postag
2	deepbiaf_PhoBERT-base	Mô hình Deep bi-affine attention, PhoBERT-base
3	deepbiaf_PhoBERT-large	Mô hình Deep bi-affine attention, PhoBERT-large
4	deepbiaf_PhoBERT-base_wo_pos	Mô hình Deep bi-affine attention, PhoBERT-base, postag
5	deepbiaf_BARTpho	Mô hình Deep bi-affine attention, BARTPho
6	deepbiaf_xlmr	Mô hình Deep bi-affine attention, XLM-RoBERTa
7	stackptr	Mô hình mạng Stack-pointer, Word2vec, postag
8	stackptr_PhoBERT-base	Mô hình mạng Stack-pointer, PhoBERT-base

$$P = \frac{\text{Số quan hệ đúng (có HEAD và LABEL đúng)}}{\text{Tổng số quan hệ được dự đoán (predicted)}}$$

$$R = \frac{\text{Số quan hệ đúng (có HEAD và LABEL đúng)}}{\text{Tổng số quan hệ của dữ liệu chuẩn (gold standard)}}$$

$$LAS = \frac{2 * P * R}{(P + R)}$$

Tương tự như đối với cuộc thi phân tích cú pháp phụ thuộc tại hội thảo CoNLL 2018[8], một quan hệ đúng được định nghĩa khi quan hệ đó đúng với cả từ phụ thuộc, từ chính và nhãn phụ thuộc. Ví dụ với quan hệ phụ thuộc `acl:tonp`, sẽ xem xét nhãn này từ từ nào phụ thuộc với từ nào trong câu, và chỉ quan tâm tới nhãn chính là `acl`, nhãn con `tonp` sẽ không tính điểm.

Ngoài ra, độ đo UAS (*Unlabeled Attachment Score*) cũng được sử dụng để tính toán hiệu suất của các mô hình. Đối với điểm số này, chỉ xem xét hai từ phụ thuộc có đúng không mà không quan tâm tới nhãn phụ thuộc là gì. Công cụ đánh giá trong CoNLL 2018[9] được sử dụng để đánh giá các hệ thống phân tích cú pháp phụ thuộc tiếng Việt.

[8] `https://universaldependencies.org/conll18/`

[9] `https://universaldependencies.org/conll18/conll18\_ud\_eval.py`

2.1.3.3. Kết quả

Đầu vào của một hệ thống phân tích cú pháp phụ thuộc có thể có hai định dạng: một là dữ liệu được xử lý trước (định dạng CoNLL-U[10]) và hai là dữ liệu chưa được xử lý (định dạng văn bản thô). Đầu ra của hệ thống sẽ được xuất dưới định dạng cột theo CoNLL-U. Hai kịch bản kiểm thử được thực hiện là: huấn luyện với bộ dữ liệu Dataset1 (chỉ gồm Viettreebank - *Package1*) và huấn luyện với bộ dữ liệu Dataset2 (kết hợp dữ liệu Viettreebank *Package1* và các dữ liệu khác *Package2*). Chi tiết các thống kê trên bộ dữ liệu huấn luyện và kiểm thử đã được mô tả trong Bảng 2.2.

Bảng 2.4, 2.5 tóm tắt kết quả của các mô hình đã xây dựng để phân tích cú pháp phụ thuộc ở định dạng CoNLL-U. Trong số các cấu hình này, mô hình Deep bi-affine với PhoBERT được huấn luyện trên Dataset2 đã đạt được kết quả tốt nhất trên cả hai tập dữ liệu kiểm thử. Khi so sánh với mô hình tốt nhất trong VLSP 2020, kết quả của chúng tôi cho thấy sự cải thiện khoảng 1% ở UAS và 2% ở LAS.

Bảng 2.4: Huấn luyện với Dataset1, đầu vào: CoNLL-U.

Mô hình	DL kiểm thử VLSP 2020		DL kiểm thử mới	
	UAS	**LAS**	**UAS**	**LAS**
deepbiaf	82.14	74.58	85.07	77.81
deepbiaf_PhoBERT-base	**85.05**	**77.41**	**88.39**	**81.05**
deepbiaf_PhoBERT-large	83.98	76.63	87.78	80.45
deepbiaf_PhoBERT_wo_pos	83.74	75.23	87.73	78.97
deepbiaf_bartpho	81.17	71.45	86.18	76.87
deepbiaf_xlmr	83.57	76.29	88.1	80.52
stackptr	82.02	74.33	85.36	78.51
stackptr_PhoBERT	83.55	75.72	87.44	79.36

Với kiểu dữ liệu kiểm thử là văn bản thô, chúng tôi đã huấn luyện 4 mô hình và nhận được kết quả như mô tả trong Bảng 2.6, 2.7. Bốn mô hình được trình bày trong phần này cho thấy hiệu suất giảm đáng kể khi so sánh với các mô hình trong Bảng 2.4, 2.5, với mỗi chỉ số giảm gần 5%. Điều này có thể giải thích vì việc thiếu các đặc trưng về tách từ và loại từ có nghĩa là mô hình không thể nắm bắt được các đặc điểm riêng biệt của những từ này, vì thế dẫn đến độ chính xác thấp hơn.

Sau đó, một cách tiếp cận khác được thử nghiệm là huấn luyện mô hình chỉ

[10]`https://universaldependencies.org/format.html`

Bảng 2.5: Huấn luyện với Dataset2, đầu vào: CoNLL-U.

Mô hình	DL kiểm thử VLSP 2020		DL kiểm thử mới	
	UAS	**LAS**	**UAS**	**LAS**
deepbiaf	82.42	74.65	86.64	79.07
deepbiaf_PhoBERT-base	**85.27**	**78.05**	**89.04**	**81.66**
deepbiaf_PhoBERT-large	83.93	76.20	88.77	81.3
deepbiaf_PhoBERT_wo_pos	83.46	74.98	87.52	78.78
deepbiaf_bartpho	83.75	75.56	87.88	79.87
deepbiaf_xlmr	83.58	76.25	87.59	80.38
stackptr	82.02	74.06	87.73	80.23
stackptr_PhoBERT	84.21	76.55	88.17	80.33

Bảng 2.6: Huấn luyện với Dataset1, đầu vào: văn bản thô.

Mô hình	DL kiểm thử VLSP 2020		DL kiểm thử mới	
	UAS	**LAS**	**UAS**	**LAS**
deepbiaf_raw	76.74	68.88	79.61	72.88
deepbiaf_PhoBERT_raw	**79.93**	**72.33**	**83.06**	**76.5**
stackptr_raw	76.64	68.64	79.64	73.15
stackptr_PhoBERT_raw	78.37	70.57	81.65	74.1

Bảng 2.7: Huấn luyện với Dataset2, đầu vào: văn bản thô.

Mô hình	DL kiểm thử VLSP 2020		DL kiểm thử mới	
	UAS	**LAS**	**UAS**	**LAS**
deepbiaf_raw	76.91	68.93	80.92	73.98
deepbiaf_PhoBERT_raw	**80.10**	**72.79**	**83.72**	**76.96**
stackptr_raw	76.48	68.37	82.19	75.14
stackptr_PhoBERT_raw	79.19	71.43	83.01	75.77

bằng cách sử dụng các nhãn phụ thuộc chính, không sử dụng các nhãn phụ. Kết quả được trình bày trong Bảng 2.8, 2.9. Các kết quả chỉ ra rằng có rất ít sự khác biệt giữa các mô hình trong cài đặt này và các mô hình trong kịch bản ban đầu, với mỗi số liệu chỉ chênh lệch khoảng 1% trên các độ đo UAS và LAS.

Bảng 2.8: Huấn luyện với các nhãn chính của Dataset1.

Mô hình	DL kiểm thử VLSP 2020		DL kiểm thử mới	
	UAS	**LAS**	**UAS**	**LAS**
deepbiaf_maintype	82.23	74.81	85.63	78.13
deepbiaf_PhoBERT_maintype	84.41	**76.91**	87.73	80.09
stackptr_maintype	81.95	74.23	86.69	79.26
stackptr_PhoBERT_maintype	**84.46**	76.69	**89.35**	**80.4**

Bảng 2.9: Huấn luyện với các nhãn chính của Dataset2.

Mô hình	DL kiểm thử VLSP 2020		DL kiểm thử mới	
	UAS	**LAS**	**UAS**	**LAS**
deepbiaf_maintype	82.41	74.62	87.37	79.53
deepbiaf_PhoBERT_maintype	**85.14**	**77.98**	**89.45**	**81.93**
stackptr_maintype	81.75	74.04	86.98	79.7
stackptr_PhoBERT_maintype	84.38	76.40	88.1	77.38

Có thể thấy rằng, trong ba kịch bản đã thử nghiệm, tiền xử lý đóng vai trò quan trọng trong hệ thống phân tích cú pháp phụ thuộc. Các mô hình sử dụng tiền xử lí gồm tách từ và gán nhãn từ loại cho hiệu quả tốt nhất, cao hơn khoảng 6% so với kết quả không tiền xử lý trước. Đồng thời, các mô hình được huấn luyện trên tập dữ liệu Dataset2 thường mang lại kết quả tốt hơn trên cả hai tập dữ liệu kiểm thử (cao hơn gần 1%). Điều này có thể được giải thích bằng số lượng câu lớn hơn và tính đa dạng cú pháp lớn hơn mà mô hình có thể học được. Ngoài ra, các câu trong tập dữ liệu kiểm thử mới cho hiệu quả tốt hơn so với tập kiểm thử của VLSP 2020, khoảng 3% vì các câu trong tập này ngắn hơn nên độ phức tạp của cú pháp cũng sẽ giảm đáng kể.

Ngoài ra, Bảng 2.10 mô tả kết quả của việc thử nghiệm mô hình tiếng Việt tốt nhất với bộ dữ liệu UD tiếng Anh. Bộ dữ liệu UD-En-EWT bao gồm 254,820 từ và 16,622 câu, được lấy từ các thể loại khác nhau như blog cá nhân, nhóm thảo luận, email, bài đánh giá và câu trả lời trên Yahoo!. Cây cú pháp phụ thuộc ban đầu được chuyển đổi tự động và được chỉnh sửa thủ công để tuân theo UD. Toàn bộ các chú thích phụ thuộc cơ bản đều được gán nhãn một lần, một phần nhỏ trong số đó được gán nhãn hai vòng và đã được chỉnh sửa lại để nâng cao tính nhất quán. Các khía cạnh khác của kho ngữ liệu, như nhãn từ loại phổ quát, đặc trưng từ vựng và phụ thuộc mở rộng, chủ yếu được thực hiện tự động với rất ít sự chỉnh sửa thủ công.

Bảng 2.10: Kết quả của hai mô hình tốt nhất đối với tập dữ liệu tiếng Anh.

Mô hình	Dữ liệu En_EWT	
	UAS	**LAS**
deepbiaf	90.68	88.85
deepbiaf_PhoBERT-base	90.01	84.86

Những kết quả này chỉ ra rằng các mô hình được xây dựng hoạt động khá tốt khi áp dụng cho dữ liệu UD tiếng Anh. Tuy nhiên, kết quả của tiếng Việt mặc

dù có sự cải thiện nhưng vẫn còn tương đối thấp so với tiếng Anh. Kết quả này có thể do nhiều nguyên nhân khác nhau. Phần tiếp theo sẽ phân tích chi tiết hơn về các yếu tố ảnh hưởng đến hiệu quả của hệ thống phân tích cú pháp phụ thuộc tiếng Việt.

2.1.3.4. Thảo luận

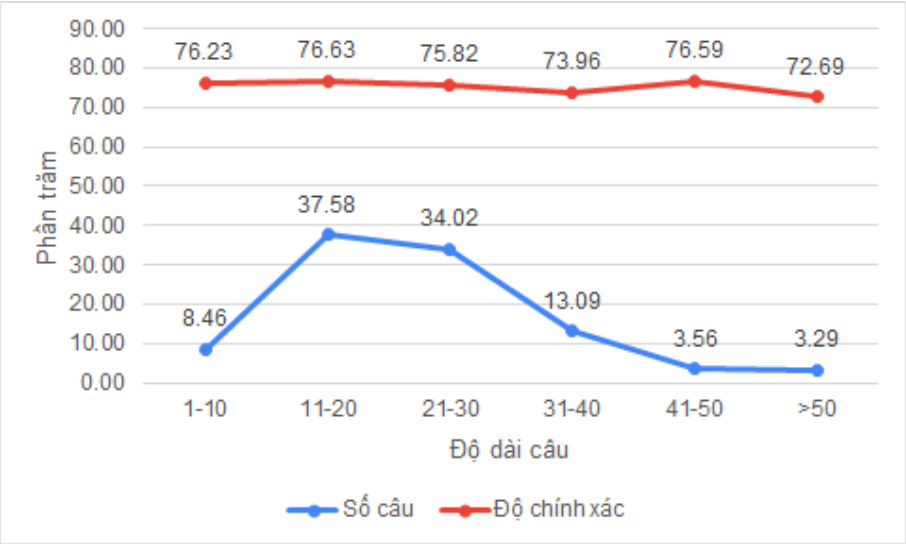
Mục tiêu chính của phần này là tiến hành phân tích các lỗi trong hệ thống phân tích cú pháp phụ thuộc với bộ dữ liệu cú pháp phụ thuộc đã xây dựng. Việc phân tích này là cần thiết bởi khi sử dụng cùng một mô hình lại cho kết quả tốt hơn đáng kể trên UD tiếng Anh (hơn khoảng 5% trên UAS và cao hơn 10% trên LAS). Các yếu tố ảnh hưởng tới hiệu quả của hệ thống phân tích cú pháp phụ thuộc có thể bắt nguồn từ bộ dữ liệu (do sự không nhất quán vẫn tồn tại trong quá trình gán nhãn) hoặc do yếu tố ngôn ngữ và các yếu tố khác. Dựa vào một số nghiên cứu trước đó [79] và [66], các yếu tố này sẽ được phân thành ba loại chính: yếu tố số độ dài (liên quan đến độ dài câu và phụ thuộc), yếu tố đồ thị (bao gồm khoảng cách đến gốc của cây phụ thuộc) và các yếu tố ngôn ngữ (bao gồm nhãn từ loại và các loại phụ thuộc). Cụ thể, các yếu tố này sẽ được định nghĩa như sau:

- Độ dài câu: Số lượng các từ trong câu.
- Độ dài phụ thuộc: Khoảng cách từ từ gốc (i) tới từ phụ thuộc (j) là $|i - j|$.
- Khoảng cách tới gốc: Độ dài đường đi từ từ phụ thuộc của một cung tới gốc của cây phụ thuộc.

Các yếu tố độ dài

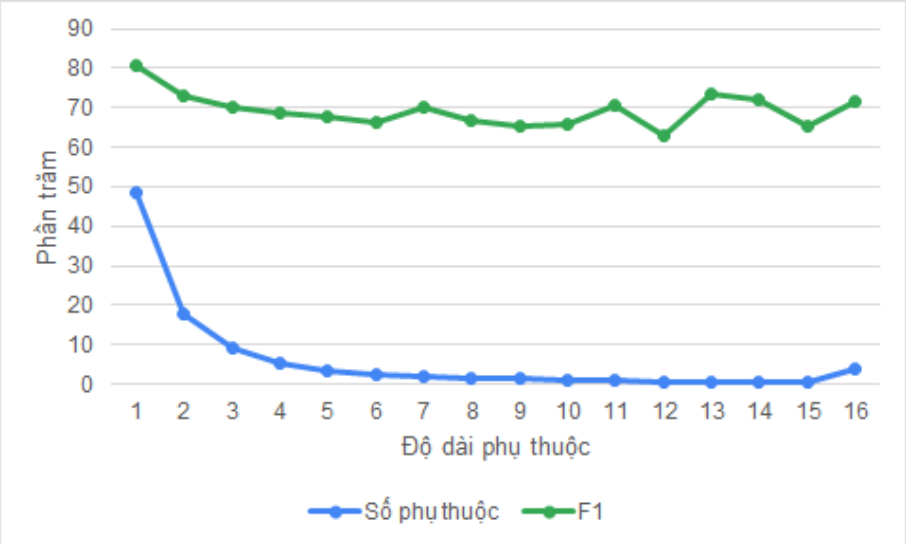
Độ dài của câu ảnh hưởng rất lớn đến việc câu đó được phân tích cú pháp chính xác hay không. Thông thường, các hệ thống phân tích cú pháp phụ thuộc thường có độ chính xác thấp hơn đối với các câu dài hơn. Các lỗi được thống kê theo độ dài câu, như mô tả trong Hình 2.5.

Trong tiếng Việt, câu ngắn có độ chính xác cao hơn câu dài khoảng 2%. Điều này khá dễ hiểu vì các câu ngắn thường có cấu trúc đơn giản: chủ ngữ - vị ngữ - tân ngữ (S-V-O) hoặc thiếu một trong các thành phần trên. Ngược lại, câu dài thường có cấu trúc phức tạp, chẳng hạn như có thêm mệnh đề thời gian, mệnh đề phụ thuộc hoặc là các câu phức (chỉ nguyên nhân, mục đích, điều kiện và kết quả). Những cấu trúc phức tạp này khó nắm bắt và huấn luyện nên độ chính xác của những câu dài hơn thường thấp hơn.



Hình 2.5: Thống kê độ chính xác dựa vào độ dài câu.

Một thuộc tính rất thú vị liên quan đến độ chính xác của hệ thống là độ dài của phần phụ thuộc, được hiển thị trong Hình 2.6. Độ dài của phần phụ thuộc từ từ w_i đến từ w_j là $|i-j|$. Thông thường, các phần phụ thuộc dài biểu thị các từ bổ nghĩa của động từ hoặc gốc trong câu là các quan hệ kết hợp và mệnh đề phụ (thời gian, địa điểm, mục đích, nguyên nhân, điều kiện, ...). Các phần phụ thuộc ngắn thường nằm trong cụm danh từ, động từ hoặc tính từ (thường là từ bổ nghĩa của phần chính của cụm từ).



Hình 2.6: Thống kê các độ đo dựa vào độ dài của phụ thuộc.

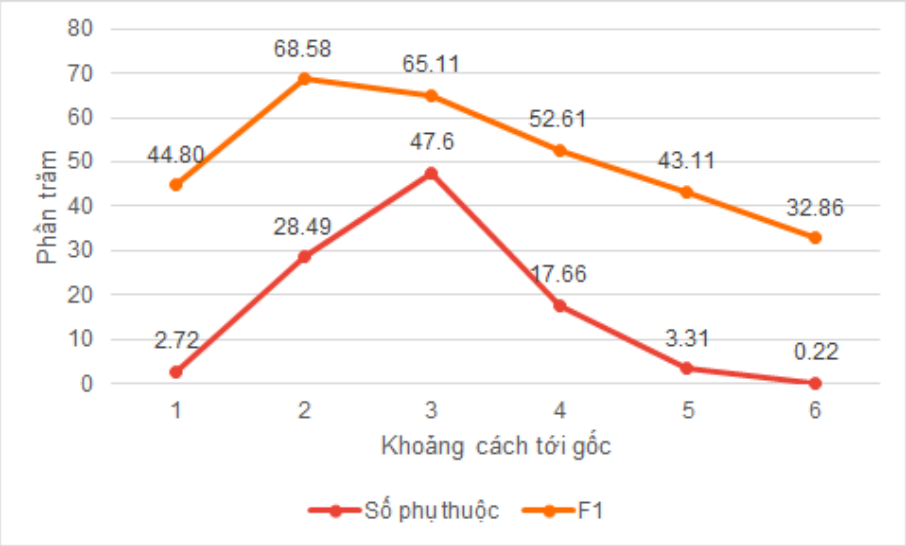
Có thể nhận thấy rằng trong kho ngữ liệu, phần lớn các phụ thuộc có độ dài ngắn, tức là các từ phụ thuộc và từ gốc của chúng thường nằm gần nhau trong

câu. Kết quả phân tích cho thấy độ đo F_1 của những phụ thuộc này cao hơn đáng kể so với các phụ thuộc có độ dài lớn, tức là khi từ phụ thuộc cách xa từ gốc của nó.

Điều này có thể được giải thích bởi khả năng của mô hình trong việc dự đoán các mối quan hệ giữa những từ gần nhau một cách hiệu quả hơn. Các từ ở gần thường có mối liên hệ ngữ nghĩa hoặc cú pháp chặt chẽ hơn, đồng thời ít bị ảnh hưởng bởi các yếu tố làm nhiễu như độ phức tạp của các thành phần câu hoặc khoảng cách lớn trong cây phụ thuộc. Ngoài ra, đặc điểm này cũng phản ánh bản chất của kho ngữ liệu, trong đó các phụ thuộc ngắn có xu hướng phổ biến hơn và đóng vai trò quan trọng trong việc hình thành cấu trúc cơ bản của câu. Kết quả này nhấn mạnh tầm quan trọng của việc cải thiện khả năng xử lý các phụ thuộc dài để nâng cao chất lượng tổng thể của hệ thống phân tích cú pháp.

Các yếu tố đồ thị

Một trong các yếu tố đồ thị cần được xem xét là khoảng cách đến gốc (*ROOT*). Đối với một cung trong biểu đồ phụ thuộc, khoảng cách này là số lượng cung trong đường dẫn ngược từ từ phụ thuộc đến ROOT. Hình 2.7 hiển thị độ đo F_1 của các cung phụ thuộc liên quan đến khoảng cách đến gốc.



Hình 2.7: Thống kê các độ đo dựa vào khoảng cách tới root.

Hình 2.7 thể hiện mối tương quan giữa số lượng phụ thuộc và độ đo F_1 theo khoảng cách tới gốc trong cây phụ thuộc. Nhìn chung, có thể thấy rằng hai đại lượng này có mối quan hệ tương quan thuận: khi số phụ thuộc ở một khoảng cách tăng thì độ đo F_1 tương ứng cũng tăng, và ngược lại. Cụ thể, tại khoảng cách 2 và 3 – nơi tập trung nhiều phụ thuộc nhất (chiếm lần lượt 28.49% và

47.6%) – độ đo F_1 cũng đạt giá trị cao nhất (68.58% và 65.11%). Trong khi đó, tại các khoảng cách 1, 5 và 6 – nơi số phụ thuộc rất thấp – độ đo F_1 cũng giảm mạnh, lần lượt chỉ còn 44.68%, 43.11% và 32.86%. Điều này cho thấy mô hình hoạt động hiệu quả hơn tại các khoảng cách có mật độ phụ thuộc cao, có thể do dữ liệu phong phú hơn giúp mô hình học tốt hơn. Ngược lại, ở các khoảng cách hiếm gặp, mô hình gặp khó khăn trong việc học và suy diễn, dẫn đến kết quả F_1 thấp. Đặc biệt, các phụ thuộc có khoảng cách nhỏ (2 và 3) thường gắn với các thành phần như bổ ngữ trực tiếp hoặc trạng ngữ gần gốc, trong những câu đơn giản, ít lớp từ xen giữa. Ngược lại, khi khoảng cách đến gốc tăng lên, F_1 giảm do cấu trúc câu trở nên phức tạp hơn – các phụ thuộc xa thường thuộc về các thành phần nằm trong mệnh đề phụ, câu ghép hoặc cấu trúc kéo dài, gây khó khăn cho mô hình trong việc nhận diện chính xác quan hệ phụ thuộc.

Các yếu tố ngôn ngữ

Các số liệu về một số loại phụ thuộc được trình bày chi tiết trong Bảng 2.11. Có thể nhận thấy rằng các kiểu phụ thuộc phổ biến và ít gây nhầm lẫn với các trường hợp khác, chẳng hạn như *root*, *obj*, *nsubj*, *case*, *cc*, *conj*, *advmod*, thường đạt độ đo Precision và Recall cao. Ngược lại, các nhãn phụ thuộc còn lại thường có kết quả thấp hơn, đặc biệt là một số nhãn như *obl*, *parataxis*, *csubj*, *list*.

Bảng 2.11: Thống kê theo nhãn cú pháp phụ thuộc.

Nhãn CPPT	**Số lượng**	**P**	**R**	**Nhãn CPPT**	**Số lượng**	**P**	**R**
acl	42	28.36	45.24	discourse	139	66.00	71.22
advcl	307	54.69	66.45	dislocated	4	16.67	75.00
advmod	1384	**90.24**	87.50	fixed	12	33.33	50.00
amod	539	83.87	67.53	list	18	33.33	44.44
appos	87	44.23	52.87	mark	469	77.73	84.86
aux	162	76.96	**90.74**	nmod	1395	61.96	57.92
case	1565	**94.73**	88.37	nsubj	1576	86.24	83.12
cc	396	85.75	88.13	**nummod**	784	**94.97**	77.04
ccomp	218	55.60	63.76	obj	2011	87.08	79.46
clf	130	81.36	73.85	obl	567	51.30	41.62
compound	962	56.91	76.20	parataxis	214	58.79	50.00
conj	1314	67.51	64.69	punct	3558	82.54	81.59
cop	201	**94.05**	86.57	root	1178	85.84	81.83
csubj	22	12.50	27.27	**vocative**	3	50.00	**100.00**
det	528	89.13	**91.67**	xcomp	880	69.14	72.05

Ngoài ra, các nhãn phụ thuộc có nhiều nhãn con cũng được phân tích chi tiết. Bảng 2.12 trình bày số liệu thống kê về các nhãn con của *compound*. Một số nhãn có độ chính xác thấp đáng kể, chẳng hạn như *compound:prt* (cụm động từ) và *compound:pron* (kết hợp giữa danh từ và đại từ). Trong tiếng Việt, nhãn

compound:pron được sử dụng để chỉ sự kết hợp giữa danh từ và đại từ, ví dụ như: "cô ấy", "cậu ấy". Tuy nhiên, nhãn này thường bị nhầm lẫn với *det:pmod*, dẫn đến giảm độ chính xác.

Bảng 2.12: Thống kê theo nhãn con của compound.

Nhãn cú pháp phụ thuộc	**Số lượng**	**P**	**R**
compound:vmod	**341**	**82.55**	**72.14**
compound:svc	268	66.44	72.39
compound:dir	169	78.47	66.86
compound:prt	20	13.64	15.00
compound:atov	8	20.00	25.00
compound:amod	7	8.11	85.71
compound:pron	16	39.13	56.25
compound:verbnoun	7	12.90	57.14
compound:adj	14	50.00	71.43

Mỗi ngôn ngữ sở hữu những đặc điểm riêng, dẫn đến sự khác biệt về hiệu quả của các mô hình phân tích cú pháp phụ thuộc dựa trên các yếu tố cụ thể. Những khảo sát, phân tích lỗi này là một thông tin hữu ích để cải thiện việc xây dựng tập nhãn phụ thuộc và kho dữ liệu thủ công cũng như các thuật toán phân tích cú pháp. Dựa trên các lỗi đã thống kê, có thể tăng cường dữ liệu cho các trường hợp có tần suất thấp và tỷ lệ lỗi cao, giúp mô hình phân tích cú pháp phụ thuộc có thể học hiệu quả hơn.

2.2. Kho ngữ liệu cú pháp thành phần cho tiếng Việt

Đối với bài toán phân tích cú pháp thành phần tiếng Việt, nhóm tác giả Nguyễn Phương Thái và cộng sự [101] đã xây dựng kho ngữ liệu gán nhãn cú pháp thành phần Viettreebank, trong khuôn khổ đề tài "Nghiên cứu phát triển một số sản phẩm thiết yếu về xử lý tiếng nói và văn bản tiếng Việt", mã số KC01.01/06-10. Kho ngữ liệu này chú giải cú pháp thành phần cho 10,165 câu tiếng Việt, gồm các thông tin về nhãn từ loại, nhãn cú pháp thành phần. Đây là một kho ngữ liệu được chia sẻ công khai cho cộng đồng xử lí ngôn ngữ, rất hữu ích và được sử dụng trong rất nhiều các tác vụ xử lí ngôn ngữ tự nhiên tiếng Việt như phân tích cú pháp phụ thuộc [91], dịch máy [117], biểu diễn và phân tích ngữ nghĩa [74], ... Tuy nhiên, Viettreebank đã được xây dựng khá lâu và chưa được cập nhật thường xuyên theo những thay đổi về các quan điểm ngôn ngữ mới, chẳng hạn như trong cách tách từ, gán nhãn từ loại, những thay đổi

trong cách xác định cụm từ, nhãn chức năng. Trong quá trình xây dựng kho ngữ liệu cú pháp phụ thuộc, việc tách từ, gán nhãn từ loại phải chỉnh sửa thủ công, tốn rất nhiều thời gian và công sức. Vì thế, mục tiêu tiếp theo của luận án là xây dựng một kho ngữ liệu cú pháp thành phần tiếng Việt chuẩn hóa, đảm bảo tính nhất quán và phù hợp với các quan điểm ngôn ngữ phổ quát.

Để xây dựng kho ngữ liệu này, luận án đã nghiên cứu các quan điểm ngôn ngữ phổ quát, cập nhật tập nhãn cú pháp thành phần tiếng Việt, sau đó thực hiện gán nhãn các nguồn dữ liệu khác nhau để làm phong phú kho ngữ liệu. Về phương pháp, các mô hình phân tích cú pháp thành phần sẽ được khảo sát và đánh giá kết quả trên kho dữ liệu đã xây dựng.

2.2.1. Xây dựng tập nhãn cú pháp thành phần tiếng Việt

Việc xây dựng tập nhãn cú pháp thành phần tiếng Việt sẽ được thực hiện trên các phần: tách từ, gán nhãn từ loại, nhãn ngữ đoạn, nhãn chức năng cú pháp và nhãn mệnh đề.

2.2.1.1. Tách từ

Tách từ là quá trình phân chia một chuỗi văn bản thành các đơn vị từ vựng riêng biệt. Trong các ngôn ngữ như tiếng Anh, việc tách từ khá dễ dàng nhờ vào việc sử dụng dấu cách. Tuy nhiên, trong tiếng Việt, việc này gặp nhiều khó khăn hơn do không sử dụng dấu cách để phân tách các từ, khiến cho việc xác định ranh giới giữa các từ trở nên phức tạp. Nguyên tắc chung về tách từ tiếng Việt dựa trên việc xác định các tổ hợp âm tiết có ý nghĩa ngữ pháp và ngữ nghĩa hoàn chỉnh. Ngoài việc giữ lại những nguyên tắc tách từ cho tiếng Việt, khi nghiên cứu về cách tách từ của dự án UD[11], các quan điểm về tách từ trong tiếng Việt cũng có một số thay đổi quan trọng như:

- Đối với tên riêng: trước đây, người ta thường coi một tên riêng là một từ hoàn chỉnh. Tuy nhiên, trong các bộ dữ liệu gần đây được xây dựng cho cộng đồng xử lí ngôn ngữ tự nhiên, các từ trong tên riêng hiện được coi là các từ riêng biệt[12]. Việc xác định các thành phần của tên cần được thực hiện rõ ràng và nhất quán giữa tên riêng và địa danh. Nhiệm vụ nhận dạng đơn vị tên riêng thuộc về công cụ nhận dạng thực thể có tên (NER), không còn là trách nhiệm của công cụ tách từ.

[11] `https://universaldependencies.org/u/overview/tokenization.html`
[12] `https://universaldependencies.org/u/pos/all.html#al-u-pos/PROPN`

Ví dụ: ở phiên bản cũ, `(NP (NNP Hồ_Chí_Minh))` sẽ được chuyển thành `(NP-PN (NNP Hồ) (NNP Chí) (NNP Minh))` trong phiên bản mới.

- Thành ngữ (*Multi-Word Expression - MWE*): Trong ngôn ngữ, có một số từ thường đi liền với nhau tạo thành nghĩa đầy đủ. Với quan niệm cũ, thành ngữ sẽ được coi tương đương như một từ. Tuy nhiên hiện nay, các thành ngữ sẽ được tách thành các từ và gán nhãn từ loại cho chúng. Sau đó sẽ được gán lại bằng nhãn chức năng thành ngữ MWE để thể hiện đó là các từ đi cùng với nhau trong một thành ngữ. Cách tiếp cận này sẽ làm cho các cấu trúc cú pháp mịn và chi tiết hơn so với cách tiếp cận cũ.

  Ví dụ, ở các phiên bản trước: `(MW hang_cùng_ngõ_hẻm)` được cập nhật thành `(NP-MWE (N hang) (ADJ cùng) (N ngõ) (ADJ hẻm))`.

2.2.1.2. Gán nhãn từ loại

Gán nhãn từ loại là việc xác định loại từ của mỗi từ trong văn bản, như danh từ, động từ, hay tính từ. Trong những năm gần đây, việc gán nhãn từ loại cũng có nhiều sự thay đổi, đặc biệt phải kể tới việc xây dựng tập nhãn từ loại phổ quát (*Universal Part of Speech - UD POS*), trong dự án UD - một dự án phát triển hệ thống chú giải ngữ pháp nhất quán cho nhiều ngôn ngữ. Khi rà soát lại các nhãn trong Viettreebank, luận án đã tham chiếu tới tập nhãn UD POS, thực hiện ánh xạ và đưa ra một số thay đổi để phù hợp với việc đối sánh đa ngữ. Ngoài các nhãn phổ biến, một số nhãn từ loại mới được luận án thêm vào để có thể nắm bắt những đặc trưng của tiếng Việt. Cụ thể là:

- Đối với động từ: Trong phiên bản trước đây chỉ dùng một nhãn động từ chung (V) cho tất cả các động từ trong câu. Tuy nhiên, trong phiên bản mới, luận án đã thêm các nhãn cho hệ từ "là", động từ tình thái, động từ bị động để có thể làm rõ đặc điểm của các động từ này.
  - V:cop: Nhãn mới cho hệ từ "`là`" trong câu. Ví dụ: `(VP (V-H là) (NP (N-H sinh viên)` được chuyển thành `(VP (V:cop-H là) (NP-PRD (N-H sinh viên)`.
  - V:mod: Nhãn mới cho động từ tình thái như `nên, cần, phải, bị, được, dám`,.... Nếu các từ "`bị, được`" không mang nghĩa bị động thì cũng sẽ để là nhãn `V:mod`.

  - V:pass: Nhãn mới cho động từ bị động. Trong tiếng Việt, không có việc chia động từ hoặc thể hiện bị động dựa vào các thì, mà dựa vào các động từ bị động như "`bị, được`".

- Đối với tính từ: Luận án bổ sung các nhãn phân biệt tính từ là định từ, tính từ có vai trò phụ từ (*adverb*).

  - ADJ:adv: Nhãn mới cho các từ thường là tính từ nhưng đóng vai trò phụ cho động từ, tính từ. Trong tiếng Anh, thường các tính từ và phó từ có biến đổi tình thái, nhưng đối với tiếng Việt thì không. Ví dụ "`nhanh`" trong cụm "`chạy nhanh`", "`giỏi`" trong cụm "`học giỏi`".
  - ADJ:det: Nhãn mới cho tính từ chỉ số lượng. Ví dụ: "`nhiều tiền`", "`gần 200 người`", ...

- Đối với đại từ: Bổ sung một số nhãn con cho các đại từ số lượng, nhân xưng, nghi vấn và chỉ định. Cụ thể là:

  - PRO:Det: Nhãn đại từ chỉ số lượng. Ví dụ: "`tất_cả`". Nhãn này phân biệt với nhãn DET (`những, các, một, trên, dưới`, ...)
  - PRO:per: Nhãn đại từ nhân xưng. Ví dụ: "`tôi, tao, mình, ...`"
  - PRO:dem: Nhãn đại từ chỉ định. Ví dụ: "`này, ấy, đấy, nọ, kia, ...`"
  - PRO:wh: Nhãn đại từ nghi vấn. Ví dụ: "`ai, gì, nào, ...`"

- Nhãn X: X được sử dụng cho các từ hoặc cụm từ không xác định được từ loại trong tiếng Việt. Thông thường, những đơn vị từ loại này bị để trống thông tin từ loại (không chú giải từ loại) trong từ điển tiếng Việt. Trong quan niệm chú giải mới, thường không sử dụng nhãn X, mà sẽ thay thế nó bằng các nhãn tương ứng với vai trò ngữ pháp của chúng trong câu.

  Ví dụ: `hình_như (X)` được đổi thành `hình_như (ADV)`.

- Nhãn Z: trong tiếng Việt, nhãn Z được sử dụng cho các từ có ý nghĩa cụ thể, thường không thể đứng một mình mà chỉ kết hợp với một từ khác để tạo ra một thực thể hoặc hành động mới. Trong chú giải mới, các từ gán nhãn Z cũ sẽ được coi là một danh từ. Do đó, nhãn Z này không còn được sử dụng nữa.

Ví dụ: `(NP (Z phó) N (chủ_tịch))` được chuyển thành `(NP (N phó) (N chủ_tịch))`.

- Dấu câu: Trong lược đồ chú giải trước đó, dấu câu trong văn bản được gắn nhãn theo hai nhãn là CH và SYM. Ngoài ra, nhãn SYM cũng được sử dụng cho các kí hiệu khác. Trong lược đồ mới, các dấu hết câu được gắn nhãn là PUNCT và các ký hiệu khác được gắn nhãn là SYM. Sự điều chỉnh này được ánh xạ tới tập nhãn UD POS.

Ngoài những thay đổi trên, luận án đã ánh xạ các nhãn tiếng Việt sang tập nhãn POS của UD[13], mô tả chi tiết trong Bảng 2.13.

Bảng 2.13: Bảng ánh xạ nhãn từ loại tiếng Việt và UD.

STT	Nhãn từ loại tiếng Việt	Nhãn từ loại UD	Định nghĩa
1	N	NOUN	danh từ
2	NNP	PROPN	danh từ riêng
3	NC	NOUN	danh từ chỉ loại
4	NU	NOUN	danh từ đơn vị
5	NUX	NOUN	tổ hợp danh từ chỉ đơn vị mở rộng
6	NUM	NUM	Số từ
7	NUMX	NUM	số từ mở rộng
8	V	VERB	động từ
9	V:cop	AUX	hệ từ LÀ
10	V:mod	AUX	Động từ tình thái
11	V:pass	AUX	Động từ bị động
12	ADJ	ADJ	tính từ
13	ADJ:adv	ADV	Từ thường là tính từ nhưng đóng vai trò phụ cho động từ, tính từ
14	ADJ:det	ADJ	tính từ chỉ số lượng
15	DET	DET	lượng từ
16	PRO	PRON	đại từ (xưng hô, chỉ định, nghi vấn)
17	PRO:det	DET	đại từ chỉ số lượng
18	PRO:per	PRON	đại từ nhân xưng
19	PRO:dem	PRON	đại từ chỉ định
20	PRO:wh	PRON	đại từ nghi vấn
21	ADV	ADV	phụ từ
22	PRE	ADP	giới từ
23	CC	CCONJ	liên từ song song
24	SC	SCONJ	liên từ phụ thuộc
25	I	INTJ	cảm từ, thán từ
26	PRT	PART	trợ từ, tiểu từ, từ tình thái
27	Z	X	yếu tố cấu tạo từ
28	PUNCT	PUNCT	dấu câu
29	SYM	SYM	kí hiệu
30	FW		Từ nguyên dạng tiếng nước ngoài

2.2.1.3. Nhãn ngữ đoạn

Đối với nhãn ngữ đoạn, luận án đã nghiên cứu đối sánh giữa tập nhãn của Viettreebank và nhãn ngữ đoạn của Penn Treebank. Sau đó, xây dựng và thêm

[13] `https://universaldependencies.org/u/pos/index.html`

vào một số nhãn mới, điều này giúp cho việc phân biệt các ngữ đoạn rõ ràng hơn. Các nhãn mới được đưa vào trong tập nhãn ngữ đoạn từ gồm có:

- Đối với các cụm danh từ: một số nhãn mới được thêm vào để chi tiết hoá các cụm danh từ trong câu. Cụ thể:
  - Cụm tên riêng: Trong phiên bản trước đó, cụm tên riêng được xác định chung với các cụm danh từ khác, sử dụng NP. Ở phiên bản mới, nhãn NP-PN được thêm vào để biểu thị cụm tên riêng. Ví dụ: "`(NP-PN Ông Trần Ngọc Lâm)` đang phát biểu."
  - Các đầu mục của danh sách, tiêu đề báo: Sử dụng nhãn LST.
  - Cụm từ là danh ngữ, dùng để hỏi về tình trạng, trạng thái, thời điểm, số lượng: sử dụng nhãn WHADVP. Ví dụ: "*(WHADVP Tại sao)* em khóc?".
  - Cụm danh từ chứa giới ngữ bổ nghĩa cho một danh từ khác hoặc thành phần phụ chú cho danh từ: sử dụng nhãn NAC. Ví dụ: "`(NAC Phần mềm kế toán về doanh nghiệp)`".
  - Cụm danh ngữ có nhiều cụm danh ngữ cùng cấp và có vai trò tương đương nhau: sử dụng nhãn NX. Ví dụ: "Những `(NX (NX quả táo màu đỏ)` và `(NX quả chuối màu xanh))`".
  - Phần trích dẫn, giải thích, bổ sung: sử dụng nhãn PRN. Ví dụ: "Nam `(PRN anh trai tôi)` vừa đi đá bóng."
- Đối với các cụm động từ:
  - Cụm động từ gồm hai động từ có ý nghĩa tương đương hoặc diễn ra theo quá trình thời gian: sử dụng nhãn VCD. Ví dụ: "Cảnh sát `(VCD tiến hành điều tra)` vụ án".
  - Cụm động từ gồm một động từ và một liên từ: sử dụng nhãn VCP. Ví dụ: `(VCP coi như_là)`.
  - Cụm động từ được cấu tạo bởi động từ thứ hai là kết quả của động từ thứ nhất: sử dụng nhãn VRD. Ví dụ: "Nam đã `(VRD thi đỗ)` đại học".
- Đối với các cụm tính từ:
  - Cụm từ là tính từ kết hợp với đại từ nghi vấn dùng để hỏi về tính chất của sự vật, hiện tượng: sử dụng nhãn WHADJP. Ví dụ: "Ngoài trời lạnh `(WHADJP thế nào)`?".

- Các ngữ đoạn khác:
  - Cụm trợ từ: trong phiên bản mới sử dụng nhãn TP. Ví dụ: "Nó thích cô ấy rồi `(TP còn gì)`!!!"
  - Cụm cảm từ: nhãn IP được thêm vào để biểu thị cụm cảm từ trong tiếng Việt. Ví dụ: "`(IP Ôi trời đất ơi)`, sao tôi khổ thế này!"
  - Cụm từ gồm hai hay nhiều thành phần không cùng loại được nối với nhau bằng liên từ đẳng lập hoặc dấu phẩy: sẽ sử dung nhãn UCP. Ví dụ: "Sản phẩm `(UCP rẻ và chất lượng tốt)`."
  - Cụm liên từ: sử dụng CONJP. Ví dụ: "`(CONJP Và trên hết)`, tôi sẽ cố gắng hết sức để đạt được nó."
  - Mệnh đề tỉnh lược: sử dụng nhãn FRAG. Ví dụ: "Hôm nay tôi không đến, `(FRAG thì ngày mai)`."
  - Mệnh đề rút gọn bổ nghĩa cho danh từ trong một danh ngữ (vị từ không phải là cụm động từ mà là cụm danh từ, tính từ, trạng từ, giới từ: sử dụng nhãn RRC. Ví dụ: "Tôi đọc quyển sách `(RRC trên kệ hôm qua)`".

2.2.1.4. Nhãn chức năng và nhãn mệnh đề

Các nhãn chức năng và mệnh đề cũng được so sánh và thêm vào các nhãn mới dựa vào tập nhãn UD. Tập nhãn chức năng được mô tả chi tiết trong Bảng 2.14.

Bảng 2.14: Các nhãn chức năng cú pháp.

STT.	**Nhãn chức năng**	**Mô tả**	**STT.**	**Nhãn chức năng**	**Mô tả**
1	H	Từ trung tâm của cụm	14	TMP	Thành phần chỉ thời gian
2	SUB	Chủ ngữ	15	LOC	Thành phần chỉ nơi chốn
3	DOB	Tân ngữ trực tiếp	16	DIR	Thành phần chỉ hướng
4	IOB	Tân ngữ gián tiếp	17	MNR	Thành phần chỉ cách thức, phương tiện, công cụ
5	DTV	Bổ ngữ gián tiếp cho động từ trao tặng	18	BNF	Bổ ngữ gián tiếp không phải cho động từ trao tặng
6	TPC	Thành phần khởi ngữ	19	PRP	Thành phần chỉ mục đích, lí do, nguyên nhân
7	CMP	Thành phần so sánh	10	COM	Thành phần bổ ngữ khác
8	PRD	Vị ngữ không phải cụm động từ	21	CND	Thành phần chỉ điều kiện
9	LGS	Chủ ngữ logic của câu ở thể bị động	22	CNC	Thành phần trạng ngữ
10	EXT	Thành phần chỉ tần suất	23	ADV	Nhãn trạng từ
11	VOC	Thành phần than gọi	24	EXC	Thành phần cảm thán
12	TRN	Thành phần chuyển tiếp	25	MOD	Thành phần tình thái
13	CIT	Thành phần trích dẫn			

Tập nhãn mệnh đề được mô tả trong Bảng 2.15.

Chi tiết về tập nhãn cú pháp thành phần cho tiếng Việt được mô tả trong Tài

Bảng 2.15: Nhãn mệnh đề.

STT	Nhãn mệnh đề	Định nghĩa
1	SQ	Câu hỏi
2	S-EXC	Câu cảm thán
3	S-CMD	Câu mệnh lệnh
4	S-EQU	Câu đẳng thức (câu có hệ từ)
5	S-PV	Câu bị động
6	S-TC	Câu có thành phần khởi ngữ
7	SBAR	Mệnh đề phụ kết (bổ nghĩa cho danh từ, động từ, và tính từ)

liệu hướng dẫn gán nhãn[14]. Sau khi rà soát và xây dựng lại tập nhãn cú pháp thành phần, việc tiếp theo là xây dựng kho ngữ liệu cú pháp thành phần tiếng Việt.

2.2.2. Kho ngữ liệu cú pháp thành phần tiếng Việt

Kho ngữ liệu cú pháp thành phần tiếng Việt được xây dựng và tuân theo quy trình chuẩn hóa đã được mô tả ở Mục 1.3.1. Về cơ sở lý luận, mục tiêu là xây dựng kho ngữ liệu cú pháp thành phần chuẩn hóa cho tiếng Việt, được chú giải cú pháp thành phần, gồm các mô tả về nhãn từ loại, nhãn ngữ đoạn, nhãn chức năng và nhãn câu, theo định dạng đặt ngoặc.

Ví dụ, một câu tiếng Việt sẽ được gán nhãn cú pháp thành phần dạng đặt ngoặc với đầy đủ các thông tin như sau: `(S (NP-SUB (NNP Nam)) (VP (ADV đang) (V-H làm) (NP (N bài_tập))) (. .))`. Các nhãn NNP, N, ADV, V là các nhãn từ loại, NP, VP là các nhãn cụm từ và S là nhãn mệnh đề.

Các dữ liệu được trích rút và kế thừa từ các nghiên cứu trước đó như kho ngữ liệu Viettreebank [101], kho ngữ liệu thực thể có tên cho tiếng Việt NER-VLSP 2021 [75]. Ngoài ra, dữ liệu văn học (tiểu thuyết Hoàng Tử Bé) và bộ dữ liệu bệnh án điện tử công khai cũng được thu thập để chú giải cú pháp thành phần. Tất cả các câu trong tập huấn luyện và tập kiểm thử đều được tách từ, gán nhãn từ loại và rà soát lại bởi các chuyên gia gán nhãn. Cụ thể, kho dữ liệu đã xây dựng được phân thành các gói như sau:

- Dữ liệu huấn luyện: 8,242 câu từ kho ngữ liệu VTB, được lấy từ trang báo "Tuổi trẻ" (`https://tuoitre.vn/`).
- Dữ liệu kiểm thử: 1,520 câu từ bộ dữ liệu NER-VLSP 2021 [75], bộ dữ liệu

[14] `https://github.com/vietnamesedp/Thesis/tree/main/ConstituencyParsing/Guidelines`

EMR và dữ liệu văn học (tiểu thuyết Hoàng tử bé). Dữ liệu kiểm thử được chia thành hai gói: kiểm thử công khai gồm có 500 và kiểm thử không công khai gồm có 1,020 câu.

Sau khi hoàn thiện tập nhãn cú pháp thành phần cho tiếng Việt, quá trình gán nhãn kho ngữ liệu cú pháp thành phần được bắt đầu. Việc gán nhãn dữ liệu được thực hiện bởi ba nhà ngôn ngữ học. Các chuyên gia gán nhãn được huấn luyện về tập nhãn và các trường hợp cụ thể của tiếng Việt trong vòng 2 tháng. Sau đó, quá trình gán nhãn gồm hai vòng, mỗi người sẽ được phân gán nhãn một gói dữ liệu ở vòng 1, thực hiện kiểm tra chéo vòng 2. Sau giai đoạn kiểm tra chéo, những chuyên gia gán nhãn sẽ tham gia thảo luận để có được sự thống nhất tốt hơn về việc gán nhãn. Bảng 2.16 thể hiện sự thống nhất giữa các cặp chuyên gia.

Bảng 2.16: Độ đồng thuận của ba chuyên gia gán nhãn cú pháp thành phần.

Các cặp chuyên gia	**Độ đồng thuận** (F_1)
Ano1-Ano2	95.64%
Ano1-Ano3	93.72%
Ano2-Ano3	94.32%
Trung bình	94.56%

Có thể thấy rằng, độ đồng thuận của các cặp chuyên gia là khá cao ($> 94\%$). Điều này thể hiện rằng kho ngữ liệu VCP 2023 đã được xây dựng một cách tỉ mỉ, chính xác và đáng tin cậy. Bảng 2.17 thống kê các thông số trong tập dữ liệu VCP 2023 đã được xây dựng.

Bảng 2.17: Thống kê dữ liệu VCP 2023.

Nhãn	**DL huấn luyện**	**DL kiểm thử 1**	**DL kiểm thử 2**
Số câu	8,242	500	1,020
Độ dài trung bình	21	19	20
VP	31,800	1,933	4,017
NP	49,437	3,048	5,735
AP	5,980	440	974
PP	10,054	624	1,434
RP	948	27	180
WHADVP	56	3	16

Ngoài ra, Bảng 2.18 thống kê về sự xuất hiện của một số nhãn từ loại cơ bản trong các tập dữ liệu được xây dựng ở trên.

Bảng 2.18: Thống kê trên tập nhãn từ loại.

STT.	Nhãn từ loại (POS)	DL Huấn luyện	DL Kiểm thử 1	DL Kiểm thử 2
1	ADJ	9,697	755	1,424
2	ADV	11,120	562	1,538
3	DET	4,164	188	543
4	N	42,604	2,600	5,336
5	NNP	10,296	473	220
6	NUM	4,332	332	451
7	PRE	10,018	617	1431
8	V	35,462	2,122	4,448

Kho ngữ liệu VCP 2023 đã được sử dụng trong cuộc thi phân tích cú pháp thành phần VCP-VLSP 2023. Nhóm nghiên cứu xây dựng kho ngữ liệu chịu trách nhiệm cập nhật và bảo trì kho ngữ liệu theo các thay đổi về quan điểm ngôn ngữ trong những năm tới. Kho ngữ liệu này[15] cũng được chia sẻ miễn phí, và sử dụng rộng rãi trong cộng đồng xử lý ngôn ngữ tự nhiên và các lĩnh vực liên quan.

2.2.3. Khảo sát các công cụ phân tích cú pháp thành phần cho tiếng Việt

Các phương pháp phân tích cú pháp thành phần mà các nhóm nghiên cứu đã phát triển và trình bày tại Hội thảo về Xử lí ngôn ngữ và tiếng nói tiếng Việt, VLSP 2022[16], 2023[17] sẽ được khảo sát và đánh giá kết quả. Các mô hình cũng được thảo luận theo những thành phần đã được liệt kê của một thẻ mô hình chuẩn hóa đã mô tả ở mục 1.3.1: chi tiết về mô hình, mục đích sử dụng, các nhân tố, độ đo, dữ liệu huấn luyện và kiểm thử, các phân tích về kết quả đạt được.

Độ đo đánh giá

Các hệ thống phân tích cú pháp thành phần nhận đầu vào là một câu văn dưới dạng chuỗi các từ (*tokens*). Đầu ra của mô hình là một cây cú pháp thành phần, được biểu diễn dưới dạng đặt ngoặc (*bracketed notation*). Các câu được bao bởi cặp thẻ "<s></s>" kèm theo thuộc tính id - đại diện cho mã số duy nhất của câu trong tập dữ liệu. Các hệ thống phân tích cú pháp được đánh giá theo độ đo F_1 dựa vào hai trường hợp cụ thể:

[15] https://github.com/vietnamesedp/Thesis/tree/main/ConstituencyParsing/VCP-2023

[16] https://vlsp.org.vn/vlsp2022/eval/vcp

[17] https://vlsp.org.vn/vlsp2023/eval/vcp

- Trường hợp 1: thực hiện so sánh tất cả các nhãn trong đầu ra của hệ thống với tất cả các nhãn trong tập dữ liệu chuẩn.
- Trường hợp 2: nếu có một số nhãn có cùng khoảng (*span*), chỉ so sánh nhãn trong dấu ngoặc đơn trong cùng, không xem xét các nhãn chức năng và phần tử rỗng trong cây phân tích thành phần.

Các độ đo được tính cụ thể theo công thức sau:

$$P = \frac{\#\text{ Số nhãn thành phần của đúng của hệ thống cho câu s}}{\#\text{ Tổng nhãn thành phần của hệ thống cho câu s}}$$

$$R = \frac{\#\text{ Số nhãn thành phần của đúng của hệ thống cho câu s}}{\#\text{ Tổng nhãn thành phần của dữ liệu chuẩn cho câu s}}$$

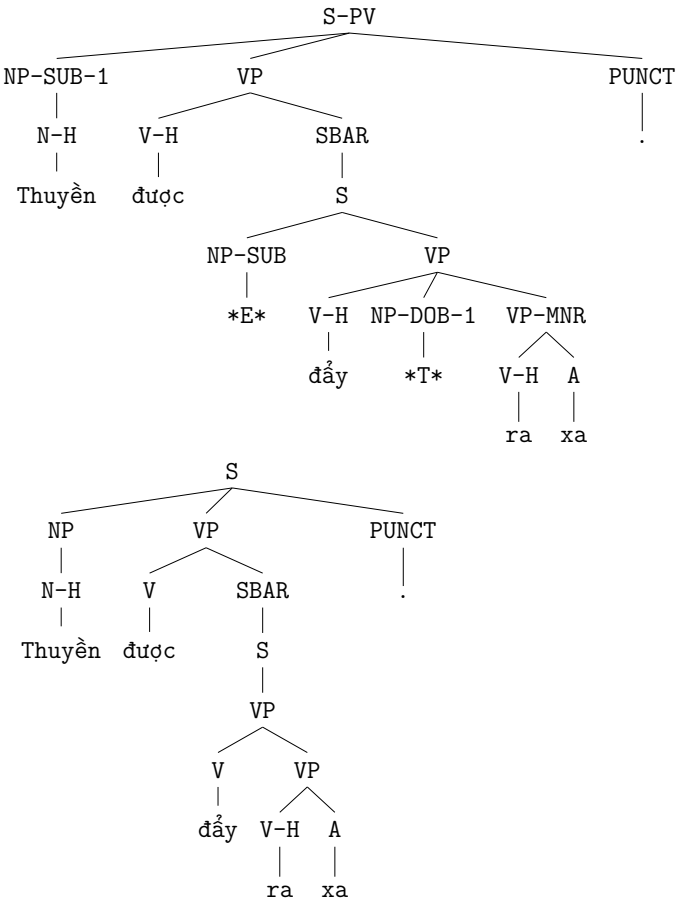
$$F_1 = \frac{2 * P * R}{(P + R)}$$

Ví dụ, với một câu tiếng Việt: "`Thuyền được đẩy ra xa.`" có hai cách phân tích cú pháp trong Hình 2.8. Cây bên trên thể hiện các nhãn cú pháp chức năng như *PV*, *SUB-1*, *-H*, *DOB-1*, *MNR*, và bao gồm một số yếu tố rỗng như **T** và **E**. Ngược lại, cây bên dưới không có các nhãn này. Tuy nhiên, dựa trên định nghĩa đã đề cập của hai trường hợp đánh giá, miễn là cấu trúc câu và các thành phần chính không thay đổi, cả hai câu này đều được coi là đúng và cho kết quả $F_1 = 1$.

Thông tin chi tiết về các mô hình

Bảng 2.19 mô tả chi tiết về mô hình và kết quả khi huấn luyện và kiểm thử trên kho ngữ liệu chuẩn đã được xây dựng. Các mô hình phân tích cú pháp thành phần được phát triển rất đa dạng, từ các mô hình truyền thống như CRF tới các mô hình tiên tiến như mạng nơ ron.

Các mô hình được mô tả dựa vào 4 tiêu chí: dữ liệu huấn luyện, các biểu diễn phân bố từ, mô hình tối ưu hoá và kết quả đạt được. Mô hình CRF hai giai đoạn được phát triển dựa vào nghiên cứu [130], huấn luyện trên 8,160 câu tiếng Việt, sử dụng hai loại biểu diễn phân bố từ là xlm-roberta-large [27] và PhoBERT-large [31]. Mô hình này sử dụng phương pháp tối ưu hóa AdamW và đạt độ chính xác 83.46%. Mô hình thứ hai được phát triển một hệ thống phân tích cú pháp mạng nơ ron thành phần dựa vào bộ phân tích có thứ tự (*in-order parser*), có sự cải tiến nhỏ với độ chính xác 83.93%. Mô hình này sử dụng bộ gán nhãn từ loại Stanza và huấn luyện lại trên 8,160 câu tiếng Việt, sử dụng PhoBERT



Hình 2.8: Hai cách phân tích cú pháp thành phần cho một câu tiếng Việt.

large và phương pháp tối ưu hóa AdaDelta và Madgrad. Mô hình thứ ba sử dụng mạng nơ ron đồ thị và cấu trúc Attach-juxtapose [127], huấn luyện trên 8,242 câu tiếng Việt và đạt độ chính xác 86.15%. Mô hình này thử nghiệm nhiều biểu diễn phân bố từ như Word2vec, PhoBERT-base, PhoBERT-large, cùng với phương pháp tối ưu hóa AdamW, cho thấy khả năng vượt trội trong việc xử lý cú pháp thành phần. Cuối cùng, phương pháp sử dụng văn phạm cấu trúc ngữ đoạn hướng trung tâm (*Head-Driven Phrase Structure Grammar - HPSG [21, 131]*) kết hợp với công cụ gán nhãn Stanza-tagger [104] và PhoBERT-large đạt hiệu quả tốt nhất trên kho ngữ liệu VCP-VLSP 2023 nhờ vào sự kết hợp hài hòa giữa phân tích cú pháp và ngữ nghĩa. HPSG giúp mô hình nắm bắt cấu trúc ngữ pháp phức tạp của tiếng Việt, trong khi Stanza-tagger cung cấp khả năng gán nhãn chính xác các từ loại và thông tin ngữ pháp. PhoBERT-large bổ sung sức mạnh xử lý ngữ nghĩa, đảm bảo mô hình hiểu tốt hơn các mối quan hệ ngữ nghĩa trong câu. Việc sử dụng các thuật toán tối ưu hóa AdaDelta và Madgrad trên tập dữ liệu 8,242 câu giúp mô hình học hiệu quả, giảm rủi ro quá khớp

Bảng 2.19: Kết quả của các mô hình phân tích cú pháp thành phần.

STT	**Mô hình**	**DL huấn luyện**	**Word Embedding**	**Tối ưu hoá**	F_1
1	Mô hình CRF hai giai đoạn	8,160	xlm-roberta-large, PhoBERT large	AdamW	83.46%
2	Mô hình mạng nơ-ron dựa vào phân tích cú pháp và tách từ Stanza	8,160	PhoBERT large	AdaDelta, Madgrad	83.93%
3	Mô hình mạng nơ-ron sử dụng Attach-juxtapose	8,242	Word2vec, PhoBERT base, large	AdamW	86.15%
4	Mô hình HPSG kết hợp với Stanza-tagger	8,242	PhoBERT large	AdaDelta, Madgrad	90.15%

(*overfitting*). Độ chính xác 90.15% khẳng định sự phù hợp của phương pháp này đối với tiếng Việt.

Thảo luận

Dựa trên kết quả của các mô hình, một số phân tích sẽ được tiến hành nhằm xác định các yếu tố ảnh hưởng đến hiệu suất mô hình. Các phân tích sẽ dựa trên các yếu tố: miền dữ liệu, nhãn từ loại, và nhãn thành phần và được thực hiện trên hai mô hình có độ chính xác tốt nhất (mô hình 3 và mô hình 4).

Về miền dữ liệu, vì dữ liệu huấn luyện chủ yếu đến từ Viettreebank, trong khi dữ liệu kiểm thử gồm ba loại: dữ liệu Viettreebank (*news*), dữ liệu y tế (*med*), và dữ liệu văn học (*lit*) nên kết quả sẽ có sự chênh lệch giữa các miền dữ liệu này, chi tiết được mô tả trong Bảng 2.20. Kết quả cho thấy rằng dữ liệu văn học hiện có hiệu suất tốt nhất với tất cả các mô hình. Điều này có thể lí giải được do các câu trong tập văn học khá ngắn (độ dài trung bình là 15), vì thế các câu này thường có cấu trúc đơn giản. Các mô hình phân tích cú pháp sẽ hoạt động tốt hơn đối với câu ngắn, trong khi câu dài với cấu trúc phức tạp, nhiều thành phần sẽ khó xử lý hơn. Tập dữ liệu kiểm thử từ Viettreebank đạt hiệu suất cao thứ hai, hơn dữ liệu y tế, do phong cách ngôn ngữ phù hợp với dữ liệu huấn luyện. Ngược lại, dữ liệu y tế thường có các đặc điểm khác biệt về tách từ, gán

nhãn từ loại, và xác định thuật ngữ, cụm từ, điều này giải thích tại sao hiệu suất phân tích với dữ liệu y tế lại thấp nhất.

Bảng 2.20: Kết quả thống kê trên các miền dữ liệu.

Mô hình	**P_news**	**R_news**	**P_med**	**R_med**	**P_lit**	**R_lit**
3	86.47%	84.16%	81.91%	78.49%	88.96%	87.47%
4	89.49%	88.24%	88.13%	87.41%	91.24%	90.53%

Về nhãn từ loại, mỗi mô hình của các nhóm nghiên cứu sử dụng các công cụ tách từ, gán nhãn từ loại khác nhau. Vì thế cũng sẽ mang lại nhiều sự khác biệt trong kết quả. Bảng 2.21 mô tả về các lỗi trên nhãn từ loại của từng mô hình. Khi so sánh với tổng số nhãn đã được liệt kê trong Bảng 2.18, có thể thấy rằng mô hình thứ 2, sử dụng HPSG kết hợp với bộ tách từ Stanza-tagger đang hoạt động vượt trội hơn hẳn. Mô hình này huấn luyện lại toàn bộ bộ tách từ Stanza trên tập nhãn từ loại mới nhất của VCP-2023, trong khi mô hình đầu tiên sử dụng mô hình PhoBERT cho việc tách từ và gán nhãn từ loại. Số lỗi trên các nhãn thường xuyên xuất hiện như động từ, danh từ, tính từ, ... của mô hình thứ 2 ít hơn mô hình thứ 1 rất nhiều. Điều này cũng ảnh hưởng tới kết quả khi so sánh các cụm có cùng khoảng cách (chỉ lấy nhãn trong cùng - thường là nhãn từ loại).

Bảng 2.21: Thống kê lỗi trên các nhãn từ loại.

Mô hình	**ADJ**	**ADV**	**DET**	**N**	**NNP**	**NUM**	**PRE**	**V**
3	561	66	26	817	61	103	179	681
4	512	144	43	327	45	64	167	635

Đối với các nhãn thành phần, Bảng 2.22 thống kê các lỗi trên các nhãn ngữ đoạn như cụm danh từ, cụm động từ, cụm tính từ, ... Có thể thấy rằng, cả hai mô hình đều gặp khó khăn trong việc phân tích các cụm danh từ, mặc dù cụm danh từ xuất hiện nhiều nhất trong dữ liệu huấn luyện, nhưng cả hai mô hình đều có số lượng lỗi cao nhất. Mô hình sử dụng văn phạm HPSG và Stanza-tagger có hiệu suất tốt hơn hẳn trên các nhãn cụm AP, NP, nhưng lại không tốt bằng mô hình sau ở các cụm VP, SBAR và PP.

Các phân tích trên cho thấy, dù các mô hình đều đạt độ chính xác khá cao ($> 83\%$), mỗi mô hình lại có cách thức hoạt động khác biệt tùy thuộc vào loại miền dữ liệu, nhãn từ loại và nhãn thành phần. Điều này được giải thích bởi sự phức tạp trong cấu trúc dữ liệu cũng như các phương pháp tiền xử lý và huấn

Bảng 2.22: Thống kê lỗi trên các nhãn thành phần.

Mô hình	AP	NP	PP	QP	VP	SBAR
3	142	434	73	16	231	22
4	109	332	76	14	474	31

luyện của từng mô hình. Nhìn chung, kết quả đã chứng minh rằng bộ dữ liệu VCP 2023 không chỉ được xây dựng kỹ lưỡng, nhất quán và tuân thủ các chuẩn mực, tạo điều kiện thuận lợi cho việc huấn luyện mô hình, mà các mô hình được khảo sát cũng rất đa dạng và áp dụng nhiều kỹ thuật tiên tiến. Đây là một đóng góp quan trọng về cả dữ liệu lẫn mô hình cho bài toán phân tích cú pháp thành phần tiếng Việt, đồng thời đặt nền móng cho các nghiên cứu tiếp theo như phân tích ngữ nghĩa mà luận án đang hướng tới.

2.3. Thuật toán chuyển từ phân tích cú pháp thành phần sang cú pháp phụ thuộc và ngược lại

Mặc dù các mô hình phân tích cú pháp đã đạt được kết quả khả quan trên các kho ngữ liệu đã xây dựng, việc chuyển đổi giữa hai kho ngữ liệu vẫn cần được chú trọng và tiếp tục phát triển. Các thuật toán này cũng được sử dụng trong bước gán nhãn dữ liệu, để tiết kiệm thời gian và công sức của các chuyên gia gán nhãn. Phần này sẽ trình bày việc xây dựng công cụ chuyển đổi từ cú pháp thành phần sang cú pháp phụ thuộc và ngược lại.

2.3.1. Từ cú pháp thành phần sang cú pháp phụ thuộc

Đầu vào của thuật toán chuyển từ cú pháp thành phần sang cú pháp phụ thuộc là câu đã được phân tích cú pháp thành phần, sử dụng định dạng đặt ngoặc, gồm có các thông tin về từ loại, ngữ đoạn, nhãn chức năng, nhãn câu và mệnh đề. Công cụ chuyển từ cú pháp thành phần sang cú pháp phụ thuộc sẽ được xây dựng theo các bước:

- Tiền xử lí: xoá bỏ qua những nhãn rỗng (**E*, *T*, *0**), thay thế một vài kí tự đặc biệt như “LBKT”, “RBKT” bằng “(” và “)”.
- Xây dựng luật xác định các từ trung tâm.
- Xây dựng luật xác định nhãn của một quan hệ phụ thuộc.

2.3.1.1. Xây dựng luật xác định từ trung tâm

Dựa vào nghiên cứu [91], các luật xác định từ trung tâm (*head-rules*) đã được cập nhật (sửa luật, thay thế nhãn) để phù hợp với tập nhãn trong kho ngữ liệu tiếng Việt hiện tại. Tập luật xác định từ trung tâm được mô tả trong Bảng 2.23.

Bảng 2.23: Luật xác định từ trung tâm của các cụm từ.

1	S	→	-H;S;VP;AP;NP;.*
2	SQ	→	-H;SQ;VP;AP;NP;.*
3	SBAR	→	-H;SBAR;S;VP;AP;NP;.*
4	NP	→	-H;NP;Nc;Nu;Np;N;PRO;.*
5	NP-PN	→	-H;NP-PN;NNP;.*
6	VP	→	-H;VP;V;ADJ;AP;N;NP;S;.*
7	AP	→	-H;AP;ADJ;N;S;.*
8	RP	←	-H;RP;ADV;PRT;NP;.*
9	PP	→	-H;PP;NP;PRE;VP;SBAR;AP;QP;.*
10	QP	→	-H;QP;NUM;.*
11	TP	→	-H;TP;PRT;.*
12	IP	→	-H;IP;I;.*
13	UCP	→	-H;UCP;NP;AP;ADJ;NC;.*
14	WHNP	→	-H;WHNP;NP;Nc;Nu;Np;N;PRO;.*
15	WHPP	→	-H;WHPP;PRE;PRO;FW;.*
16	WHADVP	→	-H;WHADVP;PRO;PRE;PRT;FW;.*
17	WHADJP	→	-H;WHADJP;ADJ;N;V;PRO;FW;.*
18	PRN	→	-H;PRN;NP;PP;.*
19	VCP	→	-H;VCP;V;SC;.*
20	VCD	→	-H;VCD;V;.*
21	VRD	→	-H;VRD;V;.*

Ví dụ: đối với luật S → -H;S;VP;AP;NP;.*, để tìm từ trung tâm (*head*) của câu S ta duyệt từ trái qua phải để tìm phần tử đầu tiên được đánh dấu là "-H", nếu tìm thấy phần tử này, nó sẽ là gốc (*root*) của câu. Nếu không có phần tử "-H", ta tìm phần tử "S" để làm trung tâm. Nếu không tìm thấy "S", ta tiếp tục tìm phần tử "VP", và cứ tiếp tục như vậy. Nếu không tìm thấy bất kỳ phần tử nào trong số này, mặc định lấy phần tử đầu tiên từ bên trái làm phần tử trung tâm.

Những thay đổi trong tập luật xác định từ trung tâm gồm có việc bỏ đi ba luật cũ vì các nhãn này đã không còn tồn tại trong bộ nhãn mới, cụ thể là:

- XP → -H;XP;X;.*
- YP → -H;YP;Y;.*

- MDP → -H;MDP;T;I;A;P;R;X;.*

Sau đó, thêm vào năm luật mới đối với các cụm từ mới được xây dựng trong tập nhãn mới:

- VCP → -H;VCP;V;SC;.*
- VCD → -H;VCD;V;.*
- VRD → -H;VRD;V;.*
- TP → -H;TP;PRT;.*
- IP → -H;IP;I;PUNCT;.*

Ngoài ra, một số nhãn đã được đổi tên, vì thế cũng có sự thay đổi trong tập luật. Ví dụ: A → ADJ, R → ADV, E → PRE, T → PRT, P → PRO, X → FW, WHRP → WHADVP, WHAP → WHADJP.

2.3.1.2. Xây dựng luật xác định nhãn phụ thuộc

Sau khi đã xác định được các từ trung tâm, có nghĩa là đã xác định được hai từ trong một câu có mối quan hệ với nhau. Việc tiếp theo chính là đặt tên cho mối quan hệ đó. Các quan hệ phụ thuộc này có thể được phân biệt dựa vào các thông tin như nhãn từ loại của từ phụ thuộc, từ trung tâm, từ bên trái, bên phải, từ ông. Bộ 60 luật đã được xây dựng để giải quyết vấn đề này. Một số luật tiêu biểu được mô tả như ở trong Bảng 2.24.

Bảng 2.24: Luật sinh nhãn phụ thuộc.

Nhãn từ phụ thuộc	**Nhãn từ trung tâm**	**Nhãn trái**	**Nhãn phải**	**Nhãn ông**	**Nhãn phụ thuộc**
V-H	VP		NP	NP-*	acl:subj
V-H	VP			NP-*; S	acl
V		N-*			compound:vmod
ADJ	NP*				amod
ADJ	VP				acomp
ADJ-H	VP*; AP*			NP	amod
ADJ-H	VP*; AP*			S	conj
ADJ-H					acomp
PRE-H	PP-*				case

Ví dụ: nhãn từ loại của từ phụ thuộc là ADJ và nhãn từ loại của từ trung tâm là VP, thì nhãn của quan hệ phụ thuộc sẽ là `acomp`.

2.3.1.3. Kết quả

Kết quả của công cụ chuyển đổi từ cú pháp thành phần sang cú pháp phụ thuộc được thực hiện trên 5,908 câu được mô tả trong Bảng 2.25.

Bảng 2.25: Kết quả chuyển cú pháp thành phần sang cú pháp phụ thuộc.

Số câu	**UAS**	**LAS**
5,908	66.20%	52.63%

Có thể thấy kết quả đạt được còn khá hạn chế do số lượng nhãn cú pháp phụ thuộc lớn, các luật chuyển đổi chưa thể nắm bắt được những trường hợp phức tạp. Vì thế, cần phải cải thiện bộ luật để đạt được kết quả tốt hơn.

2.3.2. Từ cú pháp phụ thuộc sang cú pháp thành phần

Đối với thuật toán chuyển từ cú pháp phụ thuộc sang cú pháp thành phần, đầu vào là câu đã được phân tích phụ thuộc. Thông thường, câu này sẽ ở định dạng CoNLL-U[18] (gồm có các cột thông tin về từ gốc, nhãn từ loại trong tiếng Việt, nhãn từ loại phổ quát tương ứng, từ phụ thuộc, nhãn phụ thuộc). Việc xây dựng công cụ chuyển từ cú pháp phụ thuộc sang cú pháp thành phần sẽ được xây dựng theo các bước:

- Tiền xử lí:
  - Thay thế dấu mở ngoặc "(" thành "LBKT" và dấu đóng ngoặc ")" thành "RBKT" (vì cú pháp thành phần sử dụng định dạng đặt ngoặc "()")
  - Chuẩn hoá các nhãn từ loại: ví dụ chuyển ":G" sẽ được chuẩn hóa thành "G".
- Xây dựng luật chuyển đổi:
  - Sinh cụm và từ trung tâm của cụm.
  - Tạo nhãn cụm, nhãn mệnh đề và nhãn câu.

2.3.2.1. Xây dựng luật chuyển đổi từ cú pháp phụ thuộc sang cú pháp thành phần

Để sinh được cú pháp thành phần cho các câu, trước tiên cần phải nhóm các từ thành cụm, sau đó sinh nhãn thành phần ngữ đoạn, nhãn chức năng cú

[18] `https://universaldependencies.org/format.html`

pháp, nhãn mệnh đề và nhãn câu. Thuật toán 2.1 đã được xây dựng để tìm được các cụm từ trong câu. Thuật toán này được mô tả như sau: đầu vào của thuật toán là một mảng chứa thông tin cú pháp phụ thuộc của một câu dưới dạng CoNLL-U và id của từ đang xét. Đầu ra sẽ là một mảng danh sách id của các từ phụ thuộc vào từ đang xét. Tất cả các từ phụ thuộc vào từ đang xét, sẽ tạo thành một cụm trong câu. Thuật toán này có độ phức tạp là $O(n)$, với n là số từ trong câu đang xét.

Thuật toán 2.1 getPhraserById(sentence,id)

Đầu vào:

- *sentence*: Là một mảng chứa các thông tin CoNLL-U của câu đang xét
- *id*: Là id của từ đang xét

Đầu ra : *id_lists* - một mảng danh sách id của các từ là con của từ có id

```
id_lists ← []  // Gán danh sách các id là rỗng
 for row ∈ sentence do
    // Với mỗi dòng trong câu sentence
     id ← int(row[0])  // Lấy id là phần tử đầu tiên trong dòng
     head ← int(row[6])  // Lấy từ trung tâm là phần tử thứ 6 trong dòng
     if head == id then
        // Kiểm tra điều kiện nếu từ phụ thuộc là id đang xét
         id_list.append(id)  // Thêm id đó vào danh sách ban đầu
         // Tiếp tục lấy cụm của id đang xét và thêm vào danh sách ban đầu
         id_list+ = getPhraserById(sentence, id)
return id_lists
```

Sau khi xác định được cụm của một từ trong câu, sẽ xác định được cụm của tất cả các từ. Bước tiếp theo sẽ gán nhãn thành phần cho các cụm này. Thông thường, các cụm từ trong tiếng Việt sẽ có nhãn tương ứng với nhãn của từ trung tâm của cụm từ đó. Ví dụ: cụm động từ (VP) thì từ trung tâm sẽ có nhãn động từ (V). Ngoài ra, các thông tin chức năng của đoạn cũng sẽ được sinh dựa vào nhãn từ loại của từ trung tâm cụm. Ví dụ: nếu một từ có nhãn *nsubj* thì sẽ nằm trong cụm danh từ làm chủ ngữ, có nhãn ngữ đoạn là NP và nhãn chức năng là SUB, kết hợp thành NP-SUB. Từ những khảo sát này, luận án đã nghiên cứu và xây dựng được một bộ luật gồm 22 luật sinh nhãn cụm cho cú pháp thành phần và 14 nhãn cú pháp phụ thuộc được đánh dấu trọng tâm (-H). Một số luật xác định nhãn cú pháp thành phần được mô tả trong Bảng 2.26.

Dựa vào các luật đã xây dựng, thuật toán 2.2 thực hiện kết hợp luật với các

Bảng 2.26: Bảng một số luật chuyển đổi từ cú pháp phụ thuộc sang cú pháp thành phần.

Nhãn CPPT	**Nhãn CPTP**	**Nhãn -H**	**Nhãn S**
nsubj	NP-SUB	x	x
root_V	VP	x	
root_N	NP	x	
obj	NP		
amod	AP		
nmod	NP		
appos:nmod	VP		
aux	VP	x	
acl:subj	VP		
obl:comp	NP	x	
expl	NP-TPC	x	
aux:pass	VP	x	
case	PP	x	
parataxis	VP		
ccomp	VP		
cop	VP	x	
xcomp	VP		
case_PRE	PP		

nhãn từ loại (XPOS) và nhãn từ loại phổ quát (UPOS) và cùng với các nhãn cú pháp phụ thuộc để xác định nhãn cú pháp thành phần cho mỗi từ, cụm từ.

Sau khi đã hoàn thiện hai thuật toán quan trọng là nhóm các từ (thuật toán 2.1) và tạo nhãn thành phần cho từng từ (thuật toán 2.2), bước tiếp theo là kết hợp chúng để xây dựng một cấu trúc thành phần hoàn chỉnh cho câu dựa vào thuật toán 2.3. Ý tưởng của thuật toán này là xét từng từ trong câu, kiểm tra xem từ đó có đang nằm trong một cụm nào không, nếu có thì kết hợp với các từ khác trong cụm để tạo thành cụm, ngược lại thì chỉ cần tạo ra cụm cho chính từ đó. Cuối cùng là kết hợp tất cả các cụm từ vừa được tạo thành cú pháp thành phần của câu đó. Thuật toán 2.2 có độ phức tạp là $O(1)$ do chỉ lấy cụm cho từ đang xét và thuật toán 2.3 có độ phức tạp là $O(n^3)$.

2.3.2.2. Kết quả

Sau khi hoàn thiện, công cụ thực hiện chuyển kho ngữ liệu cú pháp phụ thuộc gồm 8,152 câu tiếng Việt sang cú pháp thành phần. Sử dụng hai kịch bản đánh giá như trong VCP-VLSP 2023, kết quả cụ thể được mô tả trong Bảng 2.27.

Thuật toán 2.2 convert(sentence, id, ruleCP, ruleHead)

Đầu vào:

- *sentence*: Là một mảng chứa các thông tin CoNLL-U của câu đang xét
- *id*: id của từ đang xét
- *ruleCP*: tập luật xác định nhãn CPTP theo key và value (cột 2 trong Bảng 2.26)
- *ruleHead*: tập luật xác định nhãn -H (cột 3 trong Bảng 2.26)

Đầu ra : *vcp*: cụm của từ đang xét

```
row ← sentence[id-1]
// Các bước tiền xử lí như thay thế khoảng trắng bằng dấu _
word ← row[1].replace(' ', '_')
pos ← row[4] // Lấy ra nhãn từ loại của từ đang xét
label ← row[7] // Lấy ra nhãn phụ thuộc của từ đang xét
// Gộp hai nhãn đó lại với nhau thành một nhãn mới
newLabel ← label +'_' + pos.upper()
if label ∈ key of ruleCP then
    // Nếu nhãn phụ thuộc nằm trong tập key của luật
    // Thực hiện gán nhãn so sánh là nhãn phụ thuộc
    compareLabel ← label
else if newLabel ∈ key of ruleCP then
    // Nếu nhãn mới nằm trong tập key của luật
    // Thực hiện gán nhãn so sánh là nhãn mới
    compareLabel ← newLabel
if compareLabel ∈ key of ruleCP then
    // Nếu nhãn so sánh nằm trong tập key của luật thành phần
    if compareLabel ∈ key of ruleHead then
        // Nếu nhãn mới nằm trong tập key của luật trung tâm
        // Gán cụm vcp là các thông tin thành phần và từ trung tâm
        vcp ← '(' + ruleCP[compareLabel] + '(' + pos + ruleHead[compareLabel] + ' ' + word +' )'
    else
        // Gán cụm vcp là các thông tin thành phần và từ
        vcp ← '(' + ruleCP[compareLabel] + '(' + pos + ' ' + word +' )'
else
    vcp ← '(' + pos + ' ' + word + ' )'
return vcp
```

Thuật toán 2.3 convertDP2VCP(sentence, ruleCP, ruleHead, ruleS)

Đầu vào:

- *sentence*: Là một mảng chứa các thông tin CoNLL-U của câu đang xét
- *ruleCP*: Một tập luật key-value để tạo thêm nhãn CP
- *ruleHead*: tập các nhãn sẽ được làm Head
- *ruleS*: tập các nhãn sẽ sinh thêm S

Đầu ra : *result* - kết quả phân tích cú pháp thành phần của câu *sentence*

```
result, vcp ← '' // Khởi tạo kết quả và cụm rỗng
idRoot ← id_root // Lấy ra id của root của câu
idHeadRoot ← list_id // Lấy ra id của các từ phụ thuộc vào root
idHeadRoot.append(idRoot) // Thêm id của root vào mảng
for row ∈ sentence do
    // Với mỗi dòng trong câu, lấy ra id là phần tử đầu tiên của dòng
    id ← int(row[0])
    if id ∈ idHeadRoot then
        // Nếu id của từ đang xét nằm trong mảng idHeadRoot
        if id == idRoot then
            // Nếu từ đang xét là root, lấy ra cụm chứa root
            vcp += convert(sentence, id, ruleCP, ruleHead)
        else
            // Nếu ngược lại, lấy ra tất cả các id trong cụm
            idList ← getPhraseById(sentence, id)
            idList.append(id)
            // Sắp xếp lại theo id sort(idList)
            // Với mỗi id trong danh sách, lấy ra cụm của id đó
            for i in idList do
                if type(i) == list then
                    for k in i do
                        temp += convert(sentence, k, ruleCP, ruleHead)
                    vcp += '(' + temp + ')'
                else
                    vcp += convert(sentence, i, ruleCP, ruleHead)
    result += vcp // Nối cụm đã lấy ra vào kết quả cuối cùng
    vcp ← ''
return '(S' + result + ')'
```

Kết quả F_1 đạt 80.83% cho thấy các nhãn thành phần đã được chuyển đổi khá tương đồng với bộ dữ liệu gán nhãn thủ công. Bảng 2.28 thống kê một số

Bảng 2.27: Kết quả chuyển cú pháp phụ thuộc sang cú pháp thành phần.

Kịch bản	**P**	**R**	F_1**-score**
Đánh giá thô	88.12%	74.66%	**80.83%**
Đánh giá chính xác	82.25%	71.04%	76.23%

lỗi chuyển đổi từ cú pháp phụ thuộc sang cú pháp thành phần.

Bảng 2.28: Thống kê một số lỗi chuyển đổi từ cú pháp phụ thuộc sang cú pháp thành phần.

Nhãn chuyển đổi	**Nhãn chuẩn**	**Số lỗi**	**Nhãn chuyển đổi**	**Nhãn chuẩn**	**Số lỗi**
NP	**S**	**2,290**	NP	SBAR	117
SC	CC	649	NP	VP	73
NP	PP	474	VP	S	73
VP	NP	221	C	CC	70
AP	NP	169	VP	PP	51
VP	AP	163	AP	S	29
S	SQ	162	PP	NP	25
PRE	CC	121	PP	VP	21

Bảng trên cho thấy các lỗi từ "NP" sang "S" (2,290), "SC" sang "CC" (649), "NP" sang "PP" (474) hoặc "VP" (73), ... cho thấy công cụ chuyển đổi chưa xử lý tốt các cụm danh từ (NP) và chưa xác định chính xác vai trò cú pháp của chúng trong các ngữ cảnh khác nhau. Dựa trên những lỗi được thống kê, việc điều chỉnh và hoàn thiện tập luật là cần thiết để cải thiện khả năng nhận diện và xử lý các trường hợp phức tạp, từ đó giảm thiểu lỗi và nâng cao hiệu quả chuyển đổi.

2.4. Kết luận chương 2

Chương này trình bày quá trình xây dựng kho ngữ liệu cú pháp phụ thuộc và cú pháp thành phần cho tiếng Việt theo hướng tiếp cận đối sánh đa ngữ. Đối với kho ngữ liệu cú pháp phụ thuộc, tập nhãn đã được phát triển dựa trên tập nhãn phụ thuộc phổ quát UD. Sau đó, kho ngữ liệu bao gồm 9,848 câu đã được xây dựng, trong đó 3,000 câu được tích hợp vào kho ngữ liệu cú pháp phụ thuộc đa ngôn ngữ. Các mô hình phân tích cú pháp phụ thuộc đã được thực nghiệm trên nhiều kịch bản, và kết quả cho thấy hiệu quả cao nhất đạt được cho bài toán này.

Đối với cú pháp thành phần, luận án tiến hành cập nhật và chỉnh lí tập nhãn từ loại, nhãn ngữ đoạn và nhãn cú pháp thành phần. Đồng thời, gán nhãn 9,762

câu trong kho ngữ liệu và khảo sát các phương pháp phân tích cú pháp thành phần. Cuối cùng, luận án đề xuất thuật toán chuyển đổi giữa hai kho ngữ liệu và đánh giá kết quả đạt được.

Chương 3
XÂY DỰNG TÀI NGUYÊN VÀ CÔNG CỤ CHÚ GIẢI NGỮ NGHĨA TIẾNG VIỆT

Sau khi hoàn thành việc xây dựng các tài nguyên và công cụ chú giải cú pháp, luận án sẽ tiếp tục phát triển các tài nguyên và công cụ chú giải ngữ nghĩa cho tiếng Việt. Phân tích ngữ nghĩa sẽ được xem xét qua hai nhiệm vụ chính: gán nhãn vai nghĩa (biểu diễn ngữ nghĩa nông) và phát triển mô hình chú giải ngữ nghĩa (biểu diễn ngữ nghĩa sâu). Với bài toán biểu diễn ngữ nghĩa nông, phần 3.1 sẽ mô tả việc xây dựng tập nhãn vai nghĩa và kho dữ liệu gán nhãn vai nghĩa tiếng Việt có đối chiếu với khung vai nghĩa tiếng Anh. Đối với bài toán biểu diễn ngữ nghĩa sâu, phần 3.2 sẽ trình bày việc phát triển tập nhãn, xây dựng kho ngữ liệu ngữ nghĩa cho tiếng Việt dựa trên mô hình AMR và tập nhãn vai nghĩa LIRICS, xây dựng công cụ gán nhãn ngữ nghĩa, và thử nghiệm một số mô hình ngôn ngữ lớn để sinh biểu diễn ngữ nghĩa cho tiếng Việt.

Các kết quả của chương này đã được công bố trong các bài báo [P1, P4, P5, P7, P9] trong "Danh mục công trình công bố" của luận án.

3.1. Kho ngữ liệu có gán nhãn vai nghĩa cho tiếng Việt theo cách tiếp cận liên ngữ

Gán nhãn vai nghĩa (*Semantic Role Labeling - SRL*) là bài toán xác định vai nghĩa cho các thành phần trong câu, làm rõ cấu trúc, mối quan hệ giữa vị từ và các tham tố của nó. Có rất nhiều định nghĩa khác nhau về khái niệm vai nghĩa, các nhà ngôn ngữ cũng dùng nhiều thuật ngữ khác nhau như: cách (*cases*), quan hệ ngữ nghĩa (*semantic relations*), vai nghĩa (*roles*, *cases – roles*), hoặc vai tham tố (*thematic roles*). Thông thường, các nhãn ngữ nghĩa sẽ trả lời cho câu hỏi: "Ai, làm gì, cho ai, ở đâu, khi nào (*who did what to whom at where when*)" [14]. Gán nhãn vai nghĩa thực chất là một dạng phân tích ngữ nghĩa nông, nhưng lại là một bước cần thiết trong các tác vụ NLP.

Cũng như đối với các bài toán khác trong xử lí ngôn ngữ tự nhiên, việc sở hữu một kho ngữ liệu có gán nhãn vai nghĩa chuẩn là không thể thiếu. Đã có nhiều phương pháp được đề xuất để giải quyết vấn đề này như: dựa vào học

máy thống kê, dựa vào cú pháp phụ thuộc, ... Năm 2015, nhóm tác giả *Alan* và cộng sự [10] đã đề xuất một phương pháp mới để có thể sinh tự động kho ngữ liệu có gán nhãn vai nghĩa đa ngôn ngữ. Đây là một phương pháp thay thế hiệu quả cho việc gán nhãn vai nghĩa thủ công, được phát triển dựa vào việc ánh xạ các nhãn ngữ nghĩa từ một ngôn ngữ giàu tài nguyên (tiếng Anh) sang ngôn ngữ nghèo tài nguyên khác trên kho ngữ liệu song ngữ.

Đối với tiếng Việt, các thông tin ngữ nghĩa đã được trích rút qua một số tác vụ cụ thể như trích rút thực thể có tên, trích rút quan hệ, biểu diễn và trích rút thông tin thời gian, ... Về bài toán trích rút thực thể có tên: Một số nhóm nghiên cứu đã giải quyết bài toán này theo các hướng tiếp cận khác nhau như: nhóm tác giả Trí Trần và cộng sự [115] sử dụng SVM với độ chính xác là 87.75%, tác giả Phạm Xuân Thảo và cộng sự [124] sử dụng SVM với độ chính xác là 83.56%. Bài toán trích rút thực thể có tên cũng là một trong những nội dung của các cuộc thi VLSP 2016[1], VLSP 2018[2], 2021[3], ... Với bài toán trích rút quan hệ: Một số nhóm nghiên cứu đã phát triển các mô hình trích rút quan hệ cho văn bản tiếng Việt như: nhóm tác giả Trương Diễm và cộng sự [36] với độ chính xác 83.71%, nhóm tác giả Nguyễn Văn Nhật và cộng sự [116] đã đạt 94.89%. Bài toán trích rút quan hệ cũng là một trong số những bài toán có rất nhiều đội quan tâm và tham gia trong khuôn khổ hội thảo VLSP 2020[4]. Ngoài các bài toán đó, một số công trình của các nhóm nghiên cứu đã thực hiện xây dựng tập nhãn vai nghĩa cho tiếng Việt, xây dựng kho ngữ liệu có gán nhãn vai nghĩa và xây dựng các phương pháp để tự động xây dựng kho ngữ liệu có gán nhãn vai nghĩa. Vào năm 2015, nhóm tác giả [2] đã xây dựng một tập nhãn vai nghĩa gồm 24 nhãn và xây dựng kho ngữ liệu gán nhãn vai nghĩa gồm 5,460 câu. Sau đó, nhóm tác giả Lê Hồng Phương [102] đã phát triển thuật toán gán nhãn vai nghĩa cho tiếng Việt, với độ chính xác là 74.80%.

Nhận thấy hiệu quả của các mô hình gán nhãn vai nghĩa tiếng Việt còn khá hạn chế, cùng với kho ngữ liệu gán nhãn vai nghĩa chưa được ánh xạ tới các kho ngữ liệu ngữ nghĩa khác. Luận án đã tiếp tục xây dựng tập nhãn và kho ngữ liệu có gán nhãn vai nghĩa cho tiếng Việt theo hướng tiếp cận liên ngữ - kho ngữ liệu này được đặt tên là *viPropbank*.

Các nhãn vai nghĩa cho tiếng Việt được định nghĩa dựa vào nhãn PropBank

[1] `https://vlsp.org.vn/vlsp2016/eval/ner`
[2] `https://vlsp.org.vn/vlsp2018/eval/ner`
[3] `https://vlsp.org.vn/vlsp2021/eval/ner`
[4] `https://vlsp.org.vn/vlsp2020/eval/re`

tiếng Anh và mở rộng từ phiên bản PropBank của tiếng Việt trước đó [2]. Tập nhãn mới được xây dựng bao gồm 42 nhãn, được mô tả chi tiết trong Tài liệu gán nhãn PropBank[5] và Bảng 3.1.

Bảng 3.1: Tập nhãn vai nghĩa tiếng Việt.

Nhãn	**Định nghĩa**	**Nhãn**	**Định nghĩa**
Arg0	Tác thể	Arg4-GOL	Điểm kết thúc
Arg0-Carrier	Đương thể	ArgM	Vai phụ
Arg0-Identified	Bị đồng nhất thể	ArgM-ADJ	Tính từ
Arg0-PAG	Tác thể nguyên thuỷ	ArgM-ADV	Trạng ngữ
Arg1	Bị thể	ArgM-CAU	Nguyên nhân
Arg1-ATTRIBUTE	Thuộc tính thể	ArgM-COM	Kẻ cùng hành động
Arg1-DIR	Hướng đến	ArgM-DIR	Hướng
Arg1-GOL	Đích đến	ArgM-DIS	Diễn ngôn
Arg1-IDENTIFIER	Đồng nhất thể	ArgM-DSP	Lời thoại trực tiếp
Arg1-LOC	Địa điểm	ArgM-EXT	Mức độ
Arg1-PATIENT	Bị thể	ArgM-GOL	Đích đến
Arg1-POSSESSOR	Chủ sở hữu	ArgM-I	Cảm thán
Arg1-PPT	Bị thể nguyên thuỷ	ArgM-LOC	Địa điểm
Arg1-REASON	Nguyên nhân	ArgM-MNR	Cách thức
Arg2	Công cụ, người hưởng lợi, thuộc tính	ArgM-MOD	Tình thái
Arg-DIR	Hướng	ArgM-NEG	Phủ định
Arg2-GOL	Người hưởng lợi	ArgM-PAR	Tiểu từ tình thái
Arg2-LOC	Địa điểm	ArgM-PRD	Vị từ thứ hai
Arg2-PRD	Vị từ thứ hai	ArgM-PRP	Mục tiêu
Arg3	Điểm xuất phát	ArgM-REC	Đại từ phản thân
Arg4	Đích đến	ArgM-TMP	Thời gian

Việc xây dựng kho ngữ liệu có gán nhãn vai nghĩa còn được thực hiện dựa vào việc dóng hàng các khung PropBank từ tiếng Việt sang tiếng Anh.

Ví dụ một câu tiếng Việt: "Và bạn sẽ thấy tất_cả đều thay_đổi." sẽ được gán nhãn vai nghĩa như sau:

REL: thấy-01 (see.01)

ArgM-DIS: và

Arg0: bạn

ArgM-TMP: sẽ

Arg1: tất_cả đều thay_đổi

Có thể thấy rằng, quan hệ chính trong câu này là "thấy-01", tức là nét nghĩa thứ nhất của từ "thấy" trong từ điển VCL[6] và nét nghĩa này được dóng sang động từ see.01[7] trong PropBank tiếng Anh. Việc dóng hàng các nét nghĩa tiếng

[5] `https://github.com/vietnamesedp/Thesis/tree/main/MeaningRepresentation/viPropBank/Guidelines`

[6] `https://vlsp.hpda.vn/demo/?page=vcl`

[7] `https://propbank.github.io/v3.4.0/frames/see.html`

Việt sang tiếng Anh sẽ giúp cho có thể liên kết các nghĩa song ngữ, đồng thời có thể so sánh các tham tố của vị từ. Điều này rất có ý nghĩa trong các nghiên cứu ngữ nghĩa liên ngữ và có thể sử dụng trong các bài toán dịch máy, trích rút thông tin, hoặc xây dựng các hệ thống NLP đa ngôn ngữ.

Quá trình gán nhãn dữ liệu được thực hiện bởi 6 chuyên gia trong vòng 6 tháng. Trong 2 tháng đầu tiên, việc xây dựng tập nhãn và đào tạo các chuyên gia gán nhãn được thực hiện. Sau đó, quá trình gán nhãn dữ liệu diễn ra trong vòng 4 tháng. Bảng 3.2 mô tả độ đồng thuận giữa các cặp chuyên gia. Có thể thấy rằng độ đồng thuận trung bình đạt 89.15%, một chỉ số khá cao trong các nghiên cứu liên quan đến gán nhãn dữ liệu. Điều này cho thấy mức độ nhất quán đáng kể giữa các chuyên gia, chứng minh rằng quy trình đào tạo và chuẩn bị đã được thực hiện một cách hiệu quả, giúp các chuyên gia có thể đưa ra các quyết định gán nhãn phù hợp và thống nhất với nhau.

Bảng 3.2: Độ đồng thuận của các cặp chuyên gia gán nhãn.

Cặp chuyên gia	P	R	F_1
Anno1 Anno2	90.20%	88.30%	89.20%
Anno1 Anno3	92.80%	87.80%	90.20%
Anno1 Anno4	87.60%	88.40%	88.00%
Anno1 Anno5	89.90%	87.00%	88.40%
Anno1 Anno6	86.10%	90.60%	88.30%
Anno2 Anno3	91.10%	88.40%	89.70%
Anno2 Anno4	84.40%	86.80%	85.60%
Anno2 Anno5	87.60%	88.20%	87.90%
Anno2 Anno6	86.60%	91.30%	88.90%
Anno3 Anno4	87.80%	90.70%	89.20%
Anno3 Anno5	89.20%	92.80%	91.00%
Anno3 Anno6	85.00%	94.60%	89.50%
Anno4 Anno5	94.20%	93.00%	93.60%
Anno4 Anno6	89.00%	91.70%	90.30%
Anno5 Anno6	83.70%	91.70%	87.50%
Trung bình	**88.35%**	**90.09%**	**89.15%**

Kho ngữ liệu có gán nhãn vai nghĩa cho tiếng Việt đã được xây dựng gồm có 2,570 câu[8]. Trong đó có 1,570 câu từ dữ liệu văn học (tiểu thuyết Hoàng tử bé) và 1,000 câu từ Viettreebank (với độ dài < 25 từ). Trước khi đưa vào gán nhãn vai nghĩa, các dữ liệu đã trải qua các bước tiền xử lý như tách từ, gán nhãn từ

[8] `https://github.com/vietnamesedp/Thesis/tree/main/MeaningRepresentation/viPropBank`

loại, và gán nhãn cú pháp phụ thuộc. Chi tiết các thống kê về tập dữ liệu được mô tả trong Bảng 3.3.

Bảng 3.3: Thống kê trên từng tập dữ liệu trong viPropBank.

Nhãn	**Hoàng Tử Bé**	**Viettreebank**
Số câu	1,570	1,000
Số từ	18,096	13,968
Vị từ (động từ)	2,278	2,018
Số lượng nhãn vai nghĩa	15,537	14,654
Tập nhãn	42	30

Ngoài ra, số lượng nhãn trong kho ngữ liệu cũng được mô tả chi tiết trong Bảng 3.4.

Bảng 3.4: Thống kê số lượng nhãn trong kho ngữ liệu PropBank tiếng Việt.

Nhãn	**Số lượng**	**Nhãn**	**Số lượng**
Arg0	5,306	Arg4-GOL	53
Arg0-CARRIER	243	ArgM	1
Arg0-IDENTIFIED	144	ArgM-ADJ	19
Arg0-PAG	1	ArgM-ADV	1,960
Arg1	10,163	ArgM-CAU	558
Arg1-ATTRIBUTE	202	ArgM-COM	53
Arg1-DIR	3	ArgM-DIR	91
Arg1-GOL	3	ArgM-DIS	798
Arg1-IDENTIFIER	207	ArgM-DSP	2,084
Arg1-LOC	21	ArgM-EXT	324
Arg1-PATIENT	92	ArgM-GOL	32
Arg1-POSSESSOR	8	ArgM-I	19
Arg1-PPT	7	ArgM-LOC	790
Arg1-REASON	5	ArgM-MNR	482
Arg2	1,290	ArgM-MOD	306
Arg2-DIR	3	ArgM-NEG	397
Arg2-GOL	79	ArgM-PAR	266
Arg2-LOC	32	ArgM-PRD	680
Arg2-PRD	11	ArgM-PRP	888
Arg3	94	ArgM-REC	27
Arg4	36	ArgM-TMP	2,413

Sau khi đã hoàn thiện việc xây dựng các kho ngữ liệu cú pháp và kho ngữ liệu có gán nhãn vai nghĩa tiếng Việt, mục tiêu tiếp theo của luận án là xây dựng một mô hình biểu diễn ngữ nghĩa sâu cho tiếng Việt.

3.2. Mô hình biểu diễn ngữ nghĩa cho tiếng Việt

Việc xây dựng mô hình biểu diễn ngữ nghĩa cho tiếng Việt được thực hiện qua những giai đoạn sau:

- Khảo sát và lựa chọn mô hình biểu diễn ngữ nghĩa làm cơ sở: nghiên cứu các mô hình đã có như AMR, DCS, UCCA, GMB, cùng với các tập vai nghĩa từ PropBank, VerbNet, LIRICS, và FrameNet. Qua quá trình phân tích các ưu nhược điểm của từng mô hình, AMR và LIRICS đã được lựa chọn làm cơ sở nền tảng cho mô hình biểu diễn ngữ nghĩa tiếng Việt, nhờ khả năng linh hoạt và phù hợp với các yêu cầu biểu diễn thông tin ngữ nghĩa đối với các ngôn ngữ khác.

- Xây dựng tập nhãn ngữ nghĩa tiếng Việt: tập nhãn được xây dựng để biểu diễn các thông tin ngữ nghĩa như sự kiện, vai nghĩa, thông tin thời gian, địa điểm, thực thể có tên, đồng sở chỉ, ...

- Phát triển công cụ gán nhãn ngữ nghĩa cho tiếng Việt: để tiết kiệm thời gian và công sức của các chuyên gia gán nhãn, một công cụ hỗ trợ gán nhãn đã được thiết kế và phát triển theo đúng các chức năng của mô hình biểu diễn ngữ nghĩa cho tiếng Việt.

- Xây dựng kho ngữ liệu gán nhãn ngữ nghĩa cho tiếng Việt: sau khi hoàn thiện tập nhãn và công cụ, luận án sẽ xây dựng kho ngữ liệu theo đúng hướng dẫn gán nhãn.

Các phần tiếp theo của luận án sẽ mô tả chi tiết từng giai đoạn này.

3.2.1. Các mô hình vai nghĩa và mô hình biểu diễn ngữ nghĩa

Nhiều mô hình gán nhãn vai nghĩa đã được các nhóm nghiên cứu phát triển nhằm phục vụ cho các mục tiêu biểu diễn và phân tích ngữ nghĩa, điển hình là các mô hình như FrameNet, PropBank, và VerbNet. Mỗi mô hình có một cách tiếp cận và mức độ chi tiết riêng biệt trong việc chú giải vai nghĩa, phản ánh các khía cạnh ngữ nghĩa của ngôn ngữ ở các cấp độ khác nhau.

FrameNet là một trong những hệ thống gán nhãn vai nghĩa nổi bật, được thiết kế dựa trên khái niệm khung (*frame*). Mô hình này xây dựng một tập hợp các khung sự kiện bao quát các tình huống, sự kiện, hoặc trạng thái, từ các khung tổng quát đến cụ thể. Mỗi khung trong FrameNet mô tả các vai nghĩa

tương ứng với các thực thể hoặc khái niệm tham gia vào sự kiện hoặc trạng thái đó. Đặc biệt, FrameNet còn phát triển một hệ thống quan hệ phức tạp giữa các khung, trong đó mối quan hệ "`is-a`" là một ví dụ điển hình. Quan hệ này cho phép khung con thừa kế tất cả các vai nghĩa từ khung cha, nhưng đồng thời duy trì ít nhất một sự khác biệt cụ thể để phân biệt giữa các khung. Nhờ vào hệ thống quan hệ này, FrameNet có khả năng tổ chức thông tin vai nghĩa một cách linh hoạt và liên kết giữa các khung sự kiện khác nhau.

PropBank tập trung vào việc bổ sung thông tin vai nghĩa vào cấu trúc cú pháp của Penn Treebank. Mô hình này mô tả vai nghĩa dựa trên từng nghĩa cụ thể của động từ, sử dụng một hệ thống tham tố được đánh số thứ tự (`Arg0`, `Arg1`, ...). Mỗi tham tố đại diện cho các vai nghĩa cơ bản liên quan đến hành động của động từ, chẳng hạn như người thực hiện hành động (*Agent*) hay đối tượng bị tác động (*Patient*). Thông qua cách tiếp cận này, PropBank giúp tăng cường khả năng phân tích ngữ nghĩa dựa trên cấu trúc cú pháp, từ đó hỗ trợ tốt hơn cho các nhiệm vụ phân tích ngữ nghĩa sâu.

VerbNet là một mô hình mạng động từ, phát triển dựa trên các lớp động từ của Levin [72]. Mô hình này mở rộng các lớp động từ thành 247 lớp, bao gồm 5,257 nghĩa động từ, với một tập hợp 39 vai nghĩa được xác định rõ ràng. VerbNet không chỉ tổ chức các động từ theo lớp mà còn cung cấp thông tin chi tiết về cú pháp và ngữ nghĩa của các động từ này, bao gồm các vai nghĩa liên quan. Với mục tiêu xây dựng một hệ thống phân loại động từ toàn diện, VerbNet hỗ trợ các nghiên cứu liên quan đến hành vi ngữ nghĩa của động từ và cách thức chúng tương tác với các thực thể trong câu.

Nhìn chung, ba hệ thống gán nhãn vai nghĩa FrameNet, VerbNet và PropBank tiếp cận việc định nghĩa vai nghĩa theo các cách khác nhau, mỗi cách có những ưu điểm cũng như hạn chế riêng về tính nhất quán, độ chi tiết, và khả năng mở rộng.

FrameNet sử dụng các khung sự kiện để định nghĩa vai nghĩa, giúp phản ánh các tình huống ngữ nghĩa cụ thể nhưng không dựa vào một tập hợp vai nghĩa chung. Do đó, tên gọi của các vai nghĩa trong FrameNet có thể không nhất quán và thiếu rõ ràng khi so sánh giữa các khung. Ví dụ, cùng một vai nghĩa "Speaker" nhưng có nhiều định nghĩa khác nhau tùy theo khung sự kiện.

PropBank nổi bật với độ chi tiết trong việc xác định vai trò của các tham tố cho từng động từ. Hệ thống này phân biệt rõ ràng vai nghĩa của từng tham tố động từ, với 6 vai chính như Arg0, Arg1, ..., và thêm 11 vai ngữ cảnh cụ thể

tùy thuộc vào ngữ cảnh. Tuy nhiên, PropBank định nghĩa vai nghĩa theo các động từ riêng lẻ và dựa vào cấu trúc cú pháp chi tiết, điều này dẫn đến tính khái quát hóa thấp. Do mỗi động từ có một tập hợp vai nghĩa cụ thể, PropBank gặp khó khăn khi áp dụng cho các ngữ cảnh phức tạp hoặc đa dạng, nơi các khía cạnh ngữ nghĩa rộng và trừu tượng hơn cần được xem xét.

Ngược lại, VerbNet sử dụng một tập hợp vai nghĩa cụ thể như Theme1, Theme2, Patient1, và Patient2. Hệ thống này có ưu điểm là khái quát hóa được các vai nghĩa quan trọng và cung cấp nhiều cấp độ chi tiết cho các vai trò cụ thể của động từ. Tuy nhiên, các vai nghĩa trong VerbNet thường được định nghĩa dựa trên cấu trúc cú pháp và từ vựng cụ thể. Chẳng hạn, vai Agent trong VerbNet thường được xác định dựa trên chủ ngữ, nhưng cách định nghĩa này không phù hợp cho các cấu trúc bị động, nơi chủ ngữ không phải là tác nhân thực hiện hành động. Ngoài ra, VerbNet có một số vai nghĩa chỉ áp dụng cho các lớp động từ nhất định, gây trùng lặp với các vai nghĩa rộng hơn (ví dụ: vai Experiencer có thể trùng khớp hoặc thay thế bởi Patient hoặc Pivot), tạo ra sự mơ hồ trong gán nhãn.

Từ những nhận định đó, cùng với việc mục tiêu hướng tới chuẩn hóa dữ liệu, luận án đã nghiên cứu LIRICS[9] (*Linguistic InfRastructure for Interoperable ResourCes and Systems* [98]) - một mô hình vai nghĩa được thiết kế nhằm xây dựng chuẩn ISO về tập vai nghĩa. LIRICS định nghĩa vai nghĩa không dựa vào cú pháp hay từ vựng, mà là các khái niệm ngữ nghĩa phân biệt qua các thuộc tính đặc trưng. Các vai nghĩa này không bị giới hạn trong một vài lớp từ và được định nghĩa như các khái niệm quan hệ, mô tả cách tham tố tham gia vào sự kiện. Bộ vai nghĩa của LIRICS gồm 29 vai được đánh giá về tính dư thừa, đầy đủ và độ tin cậy. Các vai nghĩa trùng lặp đã bị loại bỏ (như *Recipient*, *Stimulus*, *Experiencer*) và việc kiểm tra tính đầy đủ được thực hiện cả về lý thuyết và thực nghiệm. Bộ kiểm tra đa ngôn ngữ cho các ngôn ngữ Anh, Hà Lan, Ý và Tây Ban Nha đã được sử dụng để kiểm tra tính tin cậy thông qua mức độ đồng thuận giữa các nhà gán nhãn.

Bên cạnh đó, trong số các mô hình biểu diễn ngữ nghĩa đã nghiên cứu, mô hình biểu diễn ngữ nghĩa trừu tượng AMR [14] là mô hình biểu diễn ngữ nghĩa trực quan và linh hoạt nhất, được thiết kế để nắm bắt bản chất ngữ nghĩa mà không cần quan tâm đến những phức tạp về ngữ pháp. AMR không chú giải các từ riêng lẻ trong câu như phân tích cú pháp phụ thuộc mà có khả năng trừu

[9] `https://semantic-annotation.uvt.nl/LIRICS_semroles.htm#guidelines`

tượng hoá các cấu trúc cú pháp, tức là một câu có cùng ý nghĩa nhưng được diễn đạt bằng nhiều cách khác nhau sẽ có cùng biểu diễn AMR. AMR có khả năng mở rộng và đã được phát triển cho nhiều ngôn ngữ như tiếng Trung, tiếng Hàn, và tiếng Pháp. Tập nhãn AMR được thiết kế để biểu diễn các thông tin ngữ nghĩa như:

- Các vai nghĩa chính: AMR sử dụng 5 nhãn vai nghĩa chính chỉ các tham tố trong khung ngữ nghĩa OntoNotes [105] (*:Arg0, :Arg1, :Arg2, :Arg3, :Arg4, :Arg5*).

- Các vai nghĩa phụ: gồm 41 loại mô tả vai trò ngữ nghĩa của các thành phần trong câu như người cùng hành động, người hưởng lợi, thời gian, địa điểm, nguyên nhân, mục đích (*:accompanier, :age, :beneficiary, :cause, ...*). Trong các vai nghĩa phụ có thêm các thông tin như:

  – Thông tin thời gian: AMR thiết kế 15 nhãn chỉ thông tin thời gian như ngày, giờ, tuần, tháng, mùa, năm, thập kỉ ... (*:day, :month, :year, :time, :season, ...*).

  – Thông tin lượng từ: 28 loại mô tả các thông tin về lượng từ thể hiện các đơn vị đo lường trong tiền tệ, thời gian, khoảng cách, diện tích, nồng độ, ... (*monetary-quantity, distance-quantity, temporal-quantity, ...*).

  – Thông tin về giới từ: AMR sử dụng 20 loại quan hệ để mô tả thông tin về các giới từ trong câu như trên, dưới, trong, ngoài, tới, ... (*:prep-out-of, :prep-to , :prep-toward, :prep-under, :prep-with, :prep-without, ...*).

  – Thông tin về các câu và các quan hệ liệt kê: AMR sử dụng 10 loại nhãn để thể hiện thông tin về các câu đi kèm và các quan hệ liệt kê liên tiếp (*:snt1*, ..., *:snt10, :op1*, ..., *:op10*).

  – Thực thể có tên[10]: Bất cứ một thực thể có tên nào trong AMR đều được biểu diễn bằng nhãn `:name`. Tuy nhiên, sau đó sẽ được chuẩn hoá bằng nhãn `:wiki` với các thông tin lấy từ Wikipedia (tiếng Anh). Các thực thể có tên gồm rất nhiều loại như: người, tổ chức, địa điểm, sự kiện, sản phẩm, ...

Có thể thấy số lượng nhãn trong AMR là rất lớn, chi tiết và tỉ mỉ, tuy nhiên vẫn tồn tại một số nhược điểm trong mô hình này như:

[10] `https://uhermjakob.github.io/amr/lib/ne-types.html`

- AMR hướng tới tiếng Anh và sử dụng các vốn từ của tiếng Anh, vì thế khi mở rộng cho các ngôn ngữ khác, cần thay đổi khá nhiều nhãn.
- AMR không xử lí các đồng sở chỉ vượt ranh giới của câu.
- AMR bỏ qua một số thông tin về thì, thể, số từ, lượng từ, các mối quan hệ sâu giữa các thành phần như danh từ - danh từ, danh từ - tính từ.

Mặc dù AMR vẫn tồn tại một số hạn chế, nhưng những nhược điểm này có thể được khắc phục bằng cách thêm các nhãn phù hợp với đặc trưng và nhu cầu của các ngôn ngữ khác.

Vì những ưu điểm trên, LIRCIS và AMR được lựa chọn làm nền tảng cho việc xây dựng tập nhãn ngữ nghĩa tiếng Việt.

3.2.2. Xây dựng tập nhãn ngữ nghĩa tiếng Việt

Để xây dựng tập nhãn ngữ nghĩa tiếng Việt, một số khác biệt giữa các cách diễn đạt ý nghĩa trong tiếng Anh và tiếng Việt đã được nghiên cứu. Việc thiết kế thêm một số nhãn ngữ nghĩa để có thể nắm bắt các thành phần này là rất cần thiết. Mục tiêu của mô hình biểu diễn ngữ nghĩa không chỉ là trả lời câu hỏi đơn giản "Ai đang làm gì với ai", mà còn để thêm các thông tin khác như: ở đâu, khi nào, tại sao và như thế nào. Ngoài ra, luận án cũng muốn khắc phục một số hạn chế của AMR như thêm vào các biểu diễn cho đồng sở chỉ, thì-thể và một số nhãn để diễn đạt các từ chức năng và các từ bổ nghĩa. Các nhãn ngữ nghĩa tiếng Việt được xây dựng theo các thành phần chính như sau:

- Vị từ (*predicate*): Trong câu tiếng Việt, vị từ là một thành phần quan trọng, biểu thị hành động, trạng thái hoặc quá trình của hoạt động đó. Vị từ trong tiếng Việt thường là động từ, có một số trường hợp đặc biệt mà các danh từ, tính từ, động từ tình thái, ... có vai trò là vị từ trong câu. Vị từ kết hợp với các thành phần khác như chủ ngữ, tân ngữ, trạng ngữ để tạo thành câu và biểu thị ý nghĩa hoàn chỉnh, rõ ràng. Đối với việc xác định vị từ của tiếng Việt, các trường hợp khác nhau của vị từ đã được xem xét và tách nhỏ. Ví dụ: Các vị từ là động từ tình thái trong tiếng Việt như: có thể, muốn, phải, khả năng, nên, ... vẫn được xác định là gốc của ngữ nghĩa, tức là người nói muốn diễn đạt rằng họ cảm thấy điều gì là cần thiết, nên làm, được phép làm, hoặc có thể xảy ra.

  Ví dụ về vị từ trong tiếng Việt như sau:

```
(p / possible-01
    :topic (g / giúp đỡ-02
        :agent (t / ta))
        :beneficiary (c/ cậu))
```

- Các vai nghĩa chính (*core roles*): Các vai nghĩa chính trong mô hình biểu diễn ngữ nghĩa của tiếng Việt được kết hợp từ LIRICS và AMR tiếng Anh, gồm có 29 nhãn. Danh sách các vai nghĩa chính và ánh xạ sang LIRICS, AMR được mô tả chi tiết trong Bảng 3.5.

Bảng 3.5: Ánh xạ giữa các nhãn LIRICS, nhãn vai nghĩa chính trong viAMR và các nhãn AMR.

LIRICS	**viAMR**	**enAMR**
agent	agent	Arg0
partner	agent(and op1, op2) patient(and op1, op2) accompanier	Arg0, Arg1, accompanier
cause	cause	Arg0
instrument	instrument	instrument
patient	patient	Arg1
pivot	pivot domain	Arg0
theme	theme	Arg1
beneficiary	beneficiary beneficiary:Arg0 beneficiary:Arg1 beneficiary:Arg2	Arg0 Arg1 Arg2 beneficiary
source	source	source
goal	goal	Arg2
result	result:Arg1 result:Arg2 result	Arg1
reason	reason	cause
purpose	purpose	purpose

Bảng 3.5: Ánh xạ giữa các nhãn LIRICS, nhãn vai nghĩa chính trong viAMR và các nhãn AMR.

LIRICS	**viAMR**	**enAMR**
time	time	time
initialTime	initialTime	
finalTime	finalTime	
duration	duration	duration
manner	manner	manner
medium	medium	
means	means	
setting	setting	location
location	location	location
initialLocation	initialLocation	
finalLocation	finalLocation	
path	path	path
distance	distance	distance quantity
amount	quant	quant
attribute	mod domain	
frequency	frequency	frequency

- Các vai nghĩa phụ (*non-core roles*): Các vai nghĩa phụ được xây dựng trong mô hình biểu diễn ngữ nghĩa tiếng Việt gồm có 88 nhãn mô tả các thành phần như: người cùng hoạt động, kẻ hưởng lợi, tuổi, điều kiện, mức độ, đích đến, hướng, công cụ, địa điểm, cách thức, ... Các nhãn vai nghĩa phụ đều đã được xem xét và đưa vào để phù hợp với các hiện tượng ngữ nghĩa trong văn bản tiếng Việt. Một số trường hợp cụ thể đối với tiếng Việt được mô tả như sau:

  – Danh từ chỉ loại: Trong tiếng Việt, một danh từ chỉ loại được sử dụng đứng trước một danh từ thường trong cụm danh từ. Các danh từ chỉ loại này nhằm mục đích phân loại các danh từ thường thành những cá thể nhất định như "cái nhà", "mảnh đất", "con mèo", ... Tương tự như đối với tiếng Trung [73], luận án mong muốn có thể biểu diễn thông

tin ngữ nghĩa này đối với tiếng Việt. Nếu cụm danh từ đã được đề cập trước đó, thì những câu sau đó chỉ cần sử dụng danh từ chỉ loại là người đọc/người nghe có thể hiểu được đang nhắc tới cái gì. Ví dụ: *"Tôi có hai cái mũ, tôi thích cái màu xanh"*. Từ "*cái*" ở câu thứ hai có thể hiểu là *"cái mũ"* do đã được đề cập đến trong câu thứ nhất. Do vậy, với các từ chỉ loại: "chiếc", "cái", ... sẽ sử dụng quan hệ **:classifier**. Ví dụ:

```
(s/sách
    :classifier (q/quyển))
```

– Thì/thể: Trong AMR của tiếng Anh, các thông tin liên quan đến thì và thể không được biểu thị trực tiếp. Đối với tiếng Việt, sự tồn tại của các phạm trù thì/thể vẫn còn là vấn đề gây nhiều tranh cãi. Một số nhà ngôn ngữ học cho rằng tiếng Việt không có thì/thể theo nghĩa truyền thống, trong khi một số khác lại khẳng định sự hiện diện của các phạm trù này. Tuy nhiên, dù quan điểm khác nhau, thì/thể vẫn được xem là những thông tin quan trọng trong việc diễn đạt ngữ nghĩa. Vì lý do đó, mô hình biểu diễn ngữ nghĩa cho tiếng Việt vẫn thiết kế các nhãn nhằm thể hiện thông tin về thì/thể. Một vấn đề khác cũng gây tranh luận là danh sách các hư từ biểu thị thì và thể trong tiếng Việt. Cụ thể, các từ như đã (quá khứ), đang (hiện tại), và sẽ (tương lai) đóng vai trò quan trọng trong việc biểu thị thời gian diễn ra của hành động hoặc trạng thái trong câu tiếng Việt, và thường được gán nhãn **:tense**. Ví dụ, về mặt ngữ nghĩa từ vựng, từ đã có bốn nghĩa khác nhau[11], trong đó một nghĩa là phụ từ dùng để diễn đạt hành động đã xảy ra trong quá khứ. Xét theo khía cạnh cú pháp, đã là một phụ từ đứng trước động từ trong cụm vị ngữ, giúp xác định vị trí của hành động trên trục thời gian mà không cần thay đổi hình thái của động từ. Tuy nhiên, các từ chỉ thì này không phải lúc nào cũng thể hiện đúng thì tương ứng. Chẳng hạn, trong câu "Giờ này ngày mai, tôi đã đi rồi", từ đã không diễn tả hành động xảy ra trong quá khứ mà mang ý nghĩa tương lai, so với thời gian tham chiếu là "giờ này ngày mai". Do đó, việc xác định nghĩa thực sự của các từ này cần đặt trong ngữ cảnh cụ thể, có thể dựa vào mệnh đề lân cận hoặc các trạng ngữ đi kèm [4].

[11] `https://vlsp.hpda.vn/demo/?page=vcl`

```
(l/làm
    :agent (t/tôi)
    :tense (s/sẽ))
```

– Quan hệ "compound": quan hệ này được sử dụng để liên kết các biểu diễn ngữ pháp hoặc từ liên quan đến nhau. Đồng thời sẽ biểu diễn cho các từ có đa tố: Gồm hai hoặc nhiều từ tố kết hợp lại với nhau thành từ có nghĩa mới. Ví dụ từ "ăn uống" được ghép từ hai từ "ăn" và "uống", từ "vui vẻ" được ghép từ "vui" và "vẻ". Với những trường hợp này, luận án sử dụng nhãn **:compound**. Ví dụ:

```
(n/nhảy
    :compound (m/múa))
```

– Quan hệ "mod": được sử dụng để biểu diễn các từ hay cụm từ bổ nghĩa cho một khái niệm. Các trường hợp này sẽ dùng quan hệ **:mod**. Ví dụ:

```
(ô/ông
    :mod (t/ta)
```

– Các cụm từ chỉ nghề nghiệp: nhà nghiên cứu, nhà khoa học, nhà chính trị, người tắt đèn, ... Với các cụm từ chỉ nghề nghiệp sẽ được sử dụng quan hệ **:compound** hoặc **:agent-of**. Ví dụ:

```
(p/person
    :agent-of (t/thắp đèn))


(n/nhà
    :compound (đ/địa lí))
```

– Các nhãn thời gian:

  * Sử dụng nhãn **:time (a/always)** cho các trường hợp: luôn luôn, lúc nào (cũng + V), bao giờ (cũng + V), mãi mãi, đời đời, kiếp kiếp, hoài, ...

* Sử dụng nhãn **:time (s/sometime)** cho các trường hợp: đôi khi, đôi lúc, thi thoảng, thỉnh thoảng, thảng hoặc, lâu lâu, một thời gian, một lúc nào đó, ...
* Sử dụng nhãn **:time (n/now)** cho các trường hợp: bây giờ, bây chừ, hiện nay, hiện giờ, hiện thời, ...
* Sử dụng nhãn **:time (b/before)** cho các trường hợp: trước đây, trước khi, hôm qua, hôm rồi, năm trước, tháng trước, tuần trước, ...
* Sử dụng nhãn **:time (b/after)** cho các trường hợp: sau đấy, sau đây, sau khi, ngày mai, ngày kia, năm sau, tháng sau, tuần sau, ...

- Các thực thể có tên (*name entity*): Các thực thể có trong mô hình biểu diễn ngữ nghĩa tiếng Việt cũng được thiết kế giống AMR tiếng Anh. Các nhãn này được sử dụng để đánh dấu các thực thể trong văn bản như người, địa điểm, tổ chức, ... Sử dụng nhãn **:wiki** để tham chiếu tới thực thể được Wikipedia định nghĩa. Ví dụ:

```
(p / person
    :wiki "Hồ_Chí_Minh"
    :name (n / name
        :op1 "Hồ"
        :op2 "Chí"
        :op3 "Minh"))
```

- Đồng sở chỉ (*co-reference*): hiện tại, mô hình AMR tiếng Anh cho phép gán đồng sở chỉ trong phạm vi một câu. Tuy nhiên, với mô hình biểu diễn ngữ nghĩa tiếng Việt, luận án đã thiết kế để có thể gán đồng sở chỉ trong phạm vi một đoạn văn. Mỗi đoạn văn, câu văn và từ trong câu được đánh số theo thứ tự xuất hiện. Sau đó, đồng sở chỉ sẽ được xác định dựa vào các id này. Việc xác định đồng sở chỉ theo đoạn văn có nhiều ứng dụng trong các bài toán như tóm tắt văn bản tự động, phân tích ngữ nghĩa, hỏi đáp.

- Các loại câu (*sentence types*): một số nhãn đã được thiết kế để biểu diễn các loại câu cho tiếng Việt: câu yêu cầu (*Imperative*), câu cảm thán (*Expressive*), câu hỏi với hư từ (*Interrogative*), câu ghép (*Multi-sentence*), câu hỏi (*amr-unknown*), ... Ví dụ:

– Câu ghép chuỗi (*Multi-sentence*): Những câu có hai hoặc nhiều vế, mỗi vế có kiểu cấu tạo giống câu đơn, liên kết với nhau không có liên từ thì dùng nhãn **:multi-sentence** và nhãn thuộc tính **:snt1**, **:snt2**, **:snt3**, ...

Bộ nhãn vai nghĩa tiếng Việt xây dựng được gồm 29 vai chính, 74 vai phụ, 18 nhãn về thời gian, địa điểm và 5 nhãn về câu, được liệt kê trong Bảng 3.6, 3.7. Bộ nhãn được mô tả và hướng dẫn gán nhãn chi tiết trong Tài liệu hướng dẫn gán nhãn vai nghĩa tiếng Việt[12].

Bảng 3.6: Danh sách các nhãn thời gian, địa điểm và nhãn câu cho AMR tiếng Việt.

STT	**Thời gian, địa điểm**	**Câu**
1	location	Imperative
2	time	Expressive
3	duration	Interrogative
4	initialTime	Multi-sentence
5	finalTime	Amr-unknown
6	initialLocation	
7	finalLocation	
8	year	
9	month	
10	weekday	
11	date-interval	
12	calendar	
13	timezone	
14	season	
15	era	
16	day	
17	location-of	
18	time-of	

Sau khi đã xây dựng được tập nhãn và hướng dẫn gán nhãn, luận án tiếp tục xây dựng công cụ hỗ trợ gán nhãn và triển khai xây dựng bộ dữ liệu gồm 1,570 câu tiếng Việt dựa trên cuốn tiểu thuyết Hoàng Tử Bé của Saint-Exupéry.

[12]`https://github.com/vietnamesedp/Thesis/tree/main/MeaningRepresentation/TaiLieu`

Bảng 3.7: Danh sách các nhãn phụ trong mô hình biểu diễn ngữ nghĩa tiếng Việt.

STT	Tên nhãn	STT	Tên nhãn	STT	Tên nhãn
1	accompanier	26	unit	51	and
2	instrument	27	value	52	op5
3	source	28	op1	53	contrast
4	result	29	op2	54	or
5	reason	30	op3	55	op6
6	purpose	31	op4	56	op7
7	manner	32	subevent	57	op8
8	medium	33	degree	58	op9
9	means	34	tense	59	Arg1
10	setting	35	modality	60	Arg2
11	path	36	ord	61	topic-of
12	quant	37	polarity	62	source-of
13	mod	38	poss	63	result-of
14	domain	39	name	64	instead-of-91
15	frequency	40	wiki	65	resemble-01
16	topic	41	dayperiod	66	snt1
17	concession	42	classifier	67	snt2
18	condition	43	prep	68	snt3
19	part	44	compared-to	69	snt4
20	part-of	45	beneficiary	70	snt5
21	direction	46	amr-choice	71	Arg0
22	example	47	regardless-91	72	be-located-at-91
23	consist-of	48	about	73	range
24	extent	49	have-quant-91	74	conj-as-if
25	compound	50	have-org-role-91		

3.2.3. Xây dựng công cụ gán nhãn ngữ nghĩa cho tiếng Việt

Để xây dựng kho ngữ liệu biểu diễn ngữ nghĩa trừu tượng AMR cho tiếng Việt, công cụ gán nhãn ngữ nghĩa [13] đã được phát triển để có thể gán nhãn dữ liệu một cách nhanh chóng và chính xác.

Công cụ được thiết kế gồm hai màn hình chính là người quản trị và người gán nhãn, có các chức năng chính sau:

- Nhập dữ liệu: người quản trị có quyền nhập toàn bộ dữ liệu vào thay vì phải nhập từng câu. Công cụ sẽ hiển thị cấu trúc cây cho những câu đã có gán nhãn trước, nếu dữ liệu rỗng, chưa được gán nhãn thì phần biểu diễn

[13] `https://amr.hpda.vn/login`

AMR sẽ để trống cho đến khi có thông tin.

- Phân chia công việc: Khi đăng nhập vào công cụ với quyền người quản trị, người quản trị có thể phân công từng đoạn văn cho những người gán nhãn. Khi đã hoàn thành vòng 1, người quản trị có thể phân chia tiếp các vòng tiếp theo để người gán nhãn làm. Khi đó, dữ liệu đã làm sẽ được hiện lên để chỉnh sửa.

- Gán nhãn: Dữ liệu đưa vào công cụ đã được tách từ. Người gán nhãn sẽ chọn câu để làm và chọn nhãn phù hợp cho từng từ tương ứng. Kết quả sẽ được hiển thị dưới dạng cấu trúc cây.

- Chỉnh sửa nhãn đã gán: Người gán nhãn sau khi gán nhãn sẽ có thể chỉnh sửa các thông tin như nhãn, thay đổi nút cha, xóa từ đã gán khỏi cây, ...

- Thêm từ mới: Với những câu có ẩn đi một số từ, người gán nhãn sẽ được phép thêm từ mới để làm rõ nghĩa của câu.
  Ví dụ: "Khi tôi lên sáu" thì có nghĩa là "Khi tôi lên sáu tuổi". Từ "tuổi" ở đây không thuộc trong câu, vì vậy người gán nhãn sẽ sử dụng thêm chức năng thêm từ mới để thêm từ "tuổi" để có thể làm rõ nghĩa hơn.

- Liên kết giữa các câu: Khi một từ trong câu có liên quan tới câu trước đó, người gán nhãn sẽ sử dụng chức năng *corref* để chọn câu và từ mình muốn liên kết cho từ đó để chỉ ra đồng tham chiếu cho các từ trong đoạn văn.

- Xuất dữ liệu: Sau khi gán nhãn, người quản trị có quyền xuất ra các tệp dữ liệu dưới dạng excel hoặc dạng text.
  Dữ liệu xuất dạng text sẽ có dạng như sau:

```
::id 9
::snt tôi đặt nó chắc chắn trên đó .
(đ / đặt
    :agent(t / tôi)
    :patient(n / nó)
    :manner(c / chắc chắn)
    :location(đ1 / đó
        :prep(t1 / trên)))
```

3.2.4. Kho ngữ liệu gán nhãn ngữ nghĩa cho tiếng Việt

Hiện tại, luận án đã xây dựng được kho ngữ liệu gán nhãn ngữ nghĩa cho tiếng Việt gồm 1,570 câu từ tiểu thuyết Hoàng Tử Bé. Kho ngữ liệu được gán nhãn bán thủ công trên công cụ chuyển đổi (dựa vào luật) và công cụ gán nhãn đã xây dựng trước đó.

Một số thống kê về tập nhãn của kho ngữ liệu được mô tả trong Bảng 3.8.

Bảng 3.8: Thống kê 20 nhãn xuất hiện nhiều nhất trong kho dữ liệu ngữ nghĩa tiếng Việt.

Nhãn	**Số lần xuất hiện**	**Nhãn**	**Số lần xuất hiện**
mod	1,376	polarity	341
agent	1,193	domain	338
theme	655	op2	330
compound	522	manner	309
quant	481	op1	296
classifier	433	patient	289
pivot	415	time	276
degree	412	and	247
topic	383	poss	236
polarity	341	tense	177

Quá trình gán nhãn dữ liệu được thực hiện bởi một nhóm gồm 5 chuyên gia ngôn ngữ, với ba vòng nhằm đảm bảo chất lượng và tính chính xác của dữ liệu. Ở mỗi vòng, các chuyên gia đều tiến hành kiểm tra chéo để phát hiện và sửa chữa những lỗi có thể còn tồn tại. Bảng 3.9 mô tả độ đồng thuận giữa các cặp chuyên gia, đạt trung bình 86.00%. Điều này cho thấy rằng dữ liệu đã được gán nhãn tỉ mỉ và khá thống nhất. Với những câu có nhiều sự khác nhau trong cách gán nhãn giữa hai chuyên gia sẽ được gán thêm vòng thứ ba để đảm bảo chất lượng của kho ngữ liệu.

Bảng 3.9: Bảng đồng thuận giữa các cặp chuyên gia.

Người thực hiện	**Người kiểm tra**	**Độ đồng thuận**
Anno1	Anno2	73.59%
Anno2	Anno3	96.91%
Anno4	Anno5	95.75%
Anno5	Anno1	77.60%
Anno3	Anno4	86.16%
Trung bình		**86.00%**

Bảng 3.10 trình bày các trường hợp không đồng thuận trong quá trình gán nhãn ngữ nghĩa cho văn bản tiếng Việt. Dựa trên bảng này, có thể thấy một số nhãn như chủ đề (*topic*), *theme*, tác thể (*agent*), và trạng thể (*pivot*) thường dễ gây nhầm lẫn.

Theo định nghĩa, nhãn *theme* được áp dụng cho thực thể hoặc đối tượng tham gia vào sự kiện hoặc trạng thái, đóng vai trò quan trọng để sự kiện diễn ra nhưng không kiểm soát cách thức xảy ra sự kiện. Đặc biệt, *theme* không bị thay đổi cấu trúc bởi sự kiện và thường là đối tượng được trải nghiệm, nhưng không phải trung tâm của trạng thái như trạng thể (*pivot*).

Trong khi đó, nhãn *topic* được sử dụng để biểu thị nội dung liên quan đến một chủ đề hoặc chủ điểm cụ thể, thường xuất hiện trong các cụm giới từ như "về cái gì". Vì vậy, khi gán nhãn, cần dựa vào định nghĩa chi tiết và rõ ràng để tránh nhầm lẫn giữa các nhãn.

Ngoài ra, số lượng các nhãn bị thiếu hoặc thừa cũng được thống kê trong dữ liệu chuẩn, chủ yếu liên quan đến các nhãn như lĩnh vực (*domain*), tác thể (*agent*), mức độ (*degree*) và nhãn danh từ chỉ loại (*classifier*). Các nhãn bị thiếu thường xuất hiện do sự bỏ sót khi gán nhãn, trong khi các nhãn thừa có thể xuất phát từ việc gán nhầm hoặc gán không phù hợp với ngữ cảnh cụ thể. Điều này cho thấy trong quy trình gán nhãn dữ liệu cần có sự thảo luận và kiểm tra chặt chẽ các tiêu chí gán nhãn để đảm bảo tính nhất quán của dữ liệu.

Bảng 3.10: Thống kê các trường hợp không đồng thuận trong kết quả gán nhãn.

Nhãn sai	**Nhãn đúng**	**Số lần**	**Nhãn thừa**	**Số lần**	**Nhãn thiếu**	**Số lần**
topic	theme	62	domain	280	domain	170
domain	pivot	46	agent	142	agent	136
pivot	agent	44	theme	89	modality	110
theme	agent	23	classifier	56	theme	90
theme	topic	21	degree	55	topic	87
domain	agent	21	quant	53	pivot	77
goal	theme	21	manner	52	manner	76
theme	domain	19	compound	51	degree	70
theme	pivot	16	time	50	quant	61
result-Arg1	theme	15	topic	49	compound	57

Kho ngữ liệu gán nhãn ngữ nghĩa, các công cụ chuyển đổi, tài liệu hướng dẫn gán nhãn[14] được công bố và chia sẻ rộng rãi trong cộng đồng xử lý ngôn ngữ tự nhiên tiếng Việt.

[14] `https://github.com/vietnamesedp/Thesis/tree/main/MeaningRepresentation`

3.3. Xây dựng mô hình phân tích ngữ nghĩa cho tiếng Việt

Mô hình ngôn ngữ lớn (LLM) là những mô hình học sâu có quy mô rất lớn, được huấn luyện trên một lượng dữ liệu khổng lồ. Cốt lõi của mô hình là các mạng nơ-ron bao gồm bộ mã hóa và bộ giải mã, được thiết kế để tự tập trung vào các yếu tố quan trọng trong dữ liệu. Bộ mã hóa và bộ giải mã giúp trích xuất ý nghĩa từ một chuỗi văn bản và hiểu mối quan hệ giữa các từ và cụm từ trong chuỗi đó. Các mô hình ngôn ngữ lớn rất linh hoạt và được sử dụng để thực hiện các tác vụ hoàn toàn khác nhau, ví dụ như trả lời câu hỏi, tóm tắt tài liệu, dịch máy, sinh câu, ...

Hiện nay, các bài toán NLP có thể sử dụng LLMs bằng cách viết các hướng dẫn (*prompt*). Viết hướng dẫn cho các mô hình ngôn ngữ lớn là một bước quan trọng để mô tả và yêu cầu mô hình tạo ra nội dung mong muốn. Mỗi hướng dẫn cần viết rõ ràng và cụ thể để mô hình có thể hiểu đúng ý định và cung cấp các kết quả phù hợp. Một trong những điểm mạnh của LLM chính là nó có thể học theo kiểu few-shot (cung cấp cho mô hình một vài mẫu) và zero-shot (không cung cấp cho mô hình bất cứ mẫu nào). Cụ thể, nó có thể tạo ra kiến thức từ giai đoạn tiền huấn luyện (*pre-training*) và sau đó nhanh chóng thích ứng với những tác vụ hoặc lĩnh vực mới với lượng dữ liệu huấn luyện bổ sung không nhiều. Đối với học theo kiểu few-shot, người ta chỉ cần cho mô hình một lượng nhỏ các ví dụ (mẫu) để nó học và nó sẽ có được kiến thức mới để hoàn thành nhiệm vụ mà trước đó nó chưa được học. Đối với học kiểu zero-shot, mô hình sẽ chỉ dựa vào kiến thức có sẵn và các hướng dẫn để tạo ra câu trả lời.

Trong số các mô hình ngôn ngữ lớn đã trình bày ở Chương 1, GPT-4 là mô hình với khả năng xử lý ngôn ngữ tự nhiên mạnh mẽ, trong đó bao gồm cả tiếng Việt. So với các phiên bản trước như GPT-2 và GPT-3, GPT-4 mang lại nhiều cải tiến đáng kể về khả năng hiểu và xử lý ngôn ngữ tự nhiên. Trước hết, GPT-4 thể hiện độ chính xác cao hơn trong việc nắm bắt ngữ cảnh và tạo ra phản hồi nhất quán, đặc biệt trong các tác vụ yêu cầu suy luận logic hoặc duy trì thông tin xuyên suốt văn bản. Khả năng phân biệt sắc thái ngữ nghĩa tinh tế và nhận diện mối quan hệ giữa các thành phần ngôn ngữ được cải thiện rõ rệt, giúp mô hình xử lý tốt hơn các câu phức tạp và các ngữ cảnh mơ hồ. Ngoài ra, GPT-4 còn thể hiện khả năng đa ngôn ngữ vượt trội, hỗ trợ tốt hơn cho các ngôn ngữ không phải tiếng Anh, trong đó có tiếng Việt. Việc mở rộng và đa dạng hóa tập dữ liệu huấn luyện cũng giúp GPT-4 hoạt động ổn định hơn trong nhiều miền

ngữ liệu khác nhau. Nhìn chung, các cải tiến của GPT-4 giúp nâng cao độ tin cậy và hiệu quả khi ứng dụng vào các bài toán phân tích ngữ nghĩa chuyên sâu.

Bên cạnh đó, mô hình Gemini, với định hướng là một mô hình ngôn ngữ đa phương thức và đa ngữ mạnh, sở hữu một số ưu điểm nổi bật so với các mô hình ngôn ngữ lớn khác. Gemini được thiết kế tối ưu cho tính đa ngôn ngữ, trong đó có sự hỗ trợ hiệu quả hơn cho các ngôn ngữ không phải tiếng Anh, như tiếng Việt, nhờ vào chiến lược huấn luyện chú trọng cân bằng dữ liệu giữa các ngôn ngữ. Bên cạnh đó, mô hình này được tích hợp sâu với hệ sinh thái phong phú của Google giúp tăng cường khả năng cập nhật kiến thức thời gian thực và xử lý các biểu đạt ngôn ngữ mang tính địa phương, khẩu ngữ hoặc gắn liền với văn hóa bản địa. Một điểm mạnh khác của Gemini là khả năng hiểu ngữ cảnh trong tương tác hội thoại, nhờ vào cơ chế điều phối thông tin linh hoạt và phản hồi tự nhiên hơn, đặc biệt trong các tác vụ yêu cầu sự thích nghi với ngữ cảnh hoặc điều chỉnh cách diễn đạt theo đối tượng người dùng. Nhờ các đặc điểm này, Gemini thể hiện sự phù hợp cao với các ứng dụng ngôn ngữ thực tế, đặc biệt trong môi trường đa văn hóa và đa ngôn ngữ.

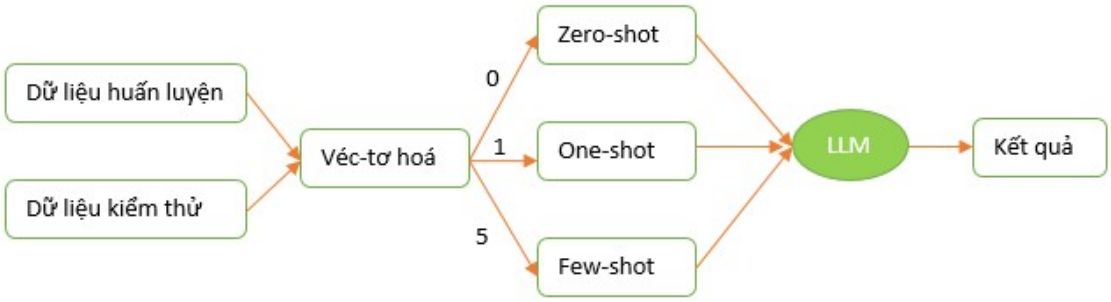
Vì thế, hai mô hình này được lựa chọn để viết hướng dẫn cho các bài toán gán nhãn vai nghĩa và phân tích ngữ nghĩa cho tiếng Việt. Cụ thể, luận án đã thực hiện viết các hướng dẫn cho 2 mô hình ngôn ngữ lớn là Gemini và GPT-4 theo cả ba phương pháp: zero-shot, one-shot và few-shot.

- Với zero-shot: Chỉ viết hướng dẫn yêu cầu sinh biểu diễn ngữ nghĩa cho các câu tiếng Việt trong tập kiểm thử. Sau đó mô hình sẽ dựa vào các kiến thức có sẵn để tạo câu trả lời.

- Với one-shot: Thực hiện chuyển các câu trong tập kiểm thử về dạng véc tơ. Sau đó với mỗi câu, sẽ thực hiện tìm câu có độ tương tự cao nhất trong tập dữ liệu huấn luyện và đưa vào các hướng dẫn của one-shot để mô hình học.

- Với few-shot: Tương tự như với one-shot, nhưng ở kịch bản này, lấy ra 5 câu có độ tương tự cao nhất (*5-shot*) với câu ở tập kiểm thử, sau đó đưa vào các hướng dẫn để mô hình học và tạo ra câu trả lời. Đồng thời, thử nghiệm với việc đưa toàn bộ tập dữ liệu huấn luyện vào cho mô hình ngôn ngữ lớn học và tạo ra câu trả lời (*few-shot*).

Luồng công việc sử dụng mô hình ngôn ngữ lớn sinh biểu diễn ngữ nghĩa cho tiếng Việt được mô tả cụ thể trong Hình 3.1.



Hình 3.1: Mô hình ngôn ngữ lớn sinh biểu diễn ngữ nghĩa cho tiếng Việt.

3.3.1. Các độ đo đánh giá

Phần này sẽ trình bày các độ đo đánh giá mô hình gán nhãn vai nghĩa và sinh biểu diễn ngữ nghĩa.

3.3.1.1. Độ đo đánh giá gán nhãn vai nghĩa

Trong các cuộc thi và hội thảo về xử lý ngôn ngữ tự nhiên, gán nhãn vai nghĩa đã trở thành một trong những nội dung thi phổ biến, được tổ chức nhiều lần tại các sự kiện như CoNLL 2004 [22], CoNLL 2005 [23], CoNLL 2009 [46].

Việc đánh giá các hệ thống gán nhãn vai nghĩa được thực hiện trên một tập dữ liệu kiểm tra riêng, chỉ bao gồm dữ liệu đầu vào đã được dự đoán. Hệ thống được đánh giá dựa trên độ chính xác (*Precision*), độ bao phủ (*Recall*), và $F_1 - score$. Độ chính xác (P) là tỷ lệ các tham tố được hệ thống dự đoán đúng. Độ bao phủ (R) là tỷ lệ các tham tố đúng được hệ thống dự đoán. Cuối cùng, $F_1 - score$ được tính theo công thức tính:

$$F_1 = \frac{2 * P * R}{(P + R)}$$

Để một tham tố được công nhận là đúng, cần xét hai khía cạnh: các từ ngữ tạo thành tham tố là đúng và nhãn vai nghĩa của nó phải chính xác.

Trong CoNLL 2009, hội thảo này đã kết hợp nhiệm vụ gán nhãn vai nghĩa với phân tích cú pháp phụ thuộc. Vì thế, định dạng của câu khi được gán nhãn vai nghĩa cũng được thay đổi. Cụ thể, đầu ra của một hệ thống phân tích cú pháp phụ thuộc và gán nhãn vai nghĩa sẽ gồm có các cột được định nghĩa như trong Bảng 3.11.

Khi sử dụng định dạng này, các quan hệ vai nghĩa được chuyển đổi thành các

Bảng 3.11: Bảng mô tả các trường trong tập dữ liệu ngữ nghĩa.

STT	**Tên cột**	**Mô tả**
1	ID	Bộ đếm token, bắt đầu từ 1 cho mỗi câu mới
2	FORM	Dạng từ hoặc ký hiệu dấu câu
3	LEMMA	Từ nguyên mẫu
4	POS	Nhãn từ loại chính
5	UPOS	Nhãn từ loại phổ quát tương ứng
6	FEAT	Các đặc điểm hình thái của ngôn ngữ (nếu có)
7	HEAD	Từ phụ thuộc của token hiện tại (ID hoặc 0 nếu là gốc)
8	DEPREL	Quan hệ cú pháp phụ thuộc (tới HEAD)
9	FILLPRED	'Y' cho các token là "vị từ"
10	PRED	Nét nghĩa của "vị từ"
11...	APREDn	Các cột chứa nhãn tham tố cho mỗi vị ngữ ngữ nghĩa (theo thứ tự ID)

phụ thuộc, tức là sẽ tạo ra n phụ thuộc ngữ nghĩa từ mỗi vị từ đến n tham tố của nó. Ngoài ra, một phụ thuộc ngữ nghĩa được tạo từ mỗi vị ngữ đến một nút ROOT ảo. Các phụ thuộc này sẽ được gán nhãn theo nghĩa của vị ngữ. Cách tiếp cận này đảm bảo rằng cấu trúc phụ thuộc ngữ nghĩa hình thành một đồ thị có gốc duy nhất, liên kết (nhưng không nhất thiết phải là đồ thị không có chu trình). Quan trọng hơn, chiến lược tính điểm này có nghĩa là nếu hệ thống gán sai nghĩa vị ngữ, nó vẫn nhận được một số điểm cho các đối số được gán đúng. Theo chiến lược này, độ chính xác, độ bao phủ, và $F_1 - score$ cho các phụ thuộc ngữ nghĩa (từ cột 9 trở đi) sẽ được tính.

3.3.1.2. Độ đo đánh giá biểu diễn ngữ nghĩa

Việc đánh giá chất lượng của mô hình sinh biểu diễn ngữ nghĩa thường sử dụng độ đo Smatch [19]. Một biểu diễn ngữ nghĩa có thể được xem ở dạng bộ ba logic mệnh đề đại diện quan hệ (biến, giá trị) hoặc quan hệ (biến, biến). Điểm Smatch được tính bằng tất cả số bộ ba có thể đối sánh tối đa trong tất cả các biến ánh xạ có thể có và nhận được $F_1 - score$, độ chính xác và độ thu hồi với:

$$P = \frac{\text{Số bộ ba khớp nhau giữa hai mô hình}}{\text{Tổng số bộ ba trong mô hình biểu diễn thứ nhất}}$$

$$R = \frac{\text{Số bộ ba khớp nhau giữa hai mô hình}}{\text{Tổng số bộ ba trong mô hình biểu diễn thứ hai}}$$

Điểm Smatch là $F_1 - score$ được tính bằng công thức: $F_1 = \frac{2*(P*R)}{(P+R)}$

Ví dụ, hai câu tiếng Anh ở định dạng AMR (dựa trên dạng PENMAN) ở Bảng 3.12 dưới đây:

Bảng 3.12: AMR của hai câu tiếng Anh ở dạng PENMAN.

Câu: the boy watches the tv.	**Câu: the girl watches the boy.**
(a / watch :Arg0 (b / boy) :Arg1 (c / tv))	(x / watch :Arg0 (y / girl) :Arg1 (z / boy))

Trong trường hợp này, Smatch tính toán và chọn các biến ánh xạ dưới đây:

a(watch) - x(watch); b(boy) - y(girl); c(tv) - z(boy)

điều đó dẫn tới 6 bộ ba khớp và không khớp được biểu diễn dưới dạng Logic ở Bảng 3.13.

Bảng 3.13: AMR của hai câu tiếng Anh ở dạng LOGIC.

Câu: the boy watches the tv.	**Câu: the girl watches the boy.**
$instance(a, watch) \wedge$ $instance(b, boy) \wedge$ $instance(c, tv) \wedge$ $TOP(a, watch) \wedge$ $Arg0(a, b) \wedge$ $Arg1(a, c)$	$instance(x, watch) \wedge$ $instance(y, girl) \wedge$ $instance(z, boy) \wedge$ $TOP(x, watch) \wedge$ $Arg0(x, y) \wedge$ $Arg1(x, z)$

Có 6 cách so khớp giữa 2 AMR và $F_1 - score$ tương ứng được mô tả ở Bảng 3.14.

Bảng 3.14: Các cách so khớp và điểm đánh giá

			M	**P**	**R**	$F_1 - score$
x = a	y = b	z = c	4	4/6	4/6	0.67
x = a	y = c	z = b	1	1/6	1/6	0.17
x = b	y = a	z = c	0	0/6	0/6	0.00
x = b	y = c	z = a	0	0/6	0/6	0.00
x = c	y = a	z = b	1	1/6	1/6	0.17
x = c	y = b	z = a	0	0/6	0/6	0.00
Smatch-score = max($F_1 - score$)						0.67

3.3.2. Kết quả

Phần này sẽ trình bày kết quả của mô hình gán nhãn vai nghĩa và sinh biểu diễn ngữ nghĩa cho tiếng Việt. Các mô hình ngôn ngữ lớn được sử dụng bằng

cách viết các hướng dẫn chi tiết để sinh vai nghĩa và biểu diễn ngữ nghĩa, sau đó được đánh giá dựa trên bộ dữ liệu chuẩn đã xây dựng.

3.3.2.1. Mô hình gán nhãn vai nghĩa

Mô hình GPT-4 được áp dụng để gán nhãn vai nghĩa cho tiếng Việt, sử dụng phương pháp *few-shot*. Phương pháp này giúp mô hình học từ một số lượng nhỏ ví dụ, qua đó nâng cao khả năng nhận diện và phân loại các tham tố của động từ. Các ví dụ cụ thể về vai trò ngữ nghĩa và cách gán nhãn được cung cấp, cùng với hướng dẫn chi tiết để mô hình có thể sinh ra các thông tin vai nghĩa cần thiết như id của vị từ, nhãn vai nghĩa và khoảng tương ứng.

Ví dụ về một câu được mô hình GPT-4 sinh gán nhãn vai nghĩa sau khi được hướng dẫn few-shot trong Bảng 3.15.

Bảng 3.15: Câu gán nhãn vai nghĩa sinh từ GPT-4.

1	Tôi	tôi	PRON	Pro	_	2	nsubj	_	_	Arg0	_	_
2	nghe	nghe	VERB	V	_	0	root	Y	hear.01	_	_	_
3	câu_chuyện	câu_chuyện	NOUN	N	_	2	obj	_	_	Arg1	_	_
4	anh	anh	NOUN	Nc	_	5	clf:det	_	_	Arg1	Arg0	_
5	Thanh	Thanh	PROPN	NNP	_	6	nsubj	_	_	Arg1	Arg0	_
6	cho	cho	VERB	V	_	3	acl	Y	give.01	Arg1	_	_
7	anh	anh	NOUN	Nc	_	8	clf:det	_	_	_	Arg2	Arg0
8	Đu	Đu	PROPN	NNP	_	6	iobj	_	_	Arg1	Arg2	Arg0
9	mượn	mượn	VERB	V	_	6	xcomp	Y	borrow.01	Arg1	Arg1	_
10	chú	chú	NOUN	Nc	_	11	clf:det	_	_	Arg1	Arg1	Arg1
11	heo	heo	NOUN	N	_	9	obj	_	_	Arg1	Arg1	Arg1
12	con	con	ADJ	Adj	_	11	amod	_	_	Arg1	Arg1	Arg1
13	.	.	PUNCT	.	_	2	punct	_	_	_	_	_

Với ví dụ này, mô hình GPT-4 đã có kết quả khá tốt khi nhận dạng đúng ba vị từ (nghe - hear.01, cho - give.01 và mượn - borrow.01), và các tham tố tương ứng (các cột từ 11-13).

Kết quả được mô tả chi tiết trong Bảng 3.16, đánh giá trên 4 chỉ số: thứ tự của vị từ (*PredicateId*), vị từ (*Predicate*), từ trung tâm của tham tố (*ArgumentHead*), khoảng của tham tố (*ArgumentSpan*).

Bảng 3.16: Kết quả đánh giá mô hình ngôn ngữ lớn gán nhãn vai nghĩa cho tiếng Việt.

Độ đo	Loại	Precision	Recall	F_1
conll09-head	Predicate	78.18%	78.18%	78.18%
conll09-head	ArgumentHead	53.99%	41.67%	47.04%
conll05-span	ArgumentSpan	50.13%	28.36%	36.22%

Mô hình đã đạt độ chính xác cao nhất với chỉ số "Predicate" (78.18%). Kết quả này cho thấy mô hình có hiệu quả tốt trong việc nhận diện các động từ và xác định chúng làm vị từ, từ đó hình thành nền tảng cho việc phân tích vai nghĩa. Tuy nhiên, các chỉ số liên quan đến tham tố ngữ nghĩa, như "ArgumentHead" (47.04%) và "ArgumentSpan" (36.22%), vẫn còn ở mức thấp. Điều này phản ánh rằng mô hình chưa thể nhận diện đầy đủ và chính xác các tham tố ngữ nghĩa trong câu, đặc biệt là các thành phần có cấu trúc phức tạp hoặc các ngữ cảnh không điển hình. Để nâng cao hiệu quả trong việc xác định các tham tố của động từ, cần tập trung nghiên cứu các kỹ thuật viết hướng dẫn chi tiết và lựa chọn mẫu dữ liệu phù hợp. Những kỹ thuật này sẽ giúp cải thiện khả năng hiểu và biểu diễn ngữ nghĩa của mô hình một cách toàn diện, đồng thời tối ưu hóa quá trình nhận diện và phân loại tham tố trong các ngữ cảnh khác nhau.

3.3.2.2. Mô hình phân tích ngữ nghĩa

Hai mô hình ngôn ngữ lớn là GPT-4 và Gemini được sử dụng để phân tích ngữ nghĩa cho văn bản tiếng Việt bằng cách viết các hướng dẫn (zero-shot, one-shot, few-shot). Ngoài ra, luận án còn thử nghiệm mô hình ViBART dựa vào công trình [71] đã xây dựng cho tiếng Anh với dữ liệu tiếng Việt. Mô hình được huấn luyện lại trên kho dữ liệu biểu diễn ngữ nghĩa tiếng Việt và sử dụng biểu diễn phân bố từ BARTPho [93]. Các kết quả phân tích biểu diễn ngữ nghĩa cho tiếng Việt được đánh giá dựa vào độ đo Smatch.

Một số ví dụ về việc sinh biểu diễn ngữ nghĩa cho tiếng Việt được mô tả chi tiết như sau:

ViBART	**Gemini**	**GPT-4**
(y/yêu :pivot (t/tôi) :mod (c/cũng) :theme (s/sa mạc))	(y/yêu :agent (t/tôi) :time (b/bao giờ) :modality (c/cũng) :patient (s/sa mạc))	(y/yêu :agent(t/tôi) :theme(s/sa mạc) :time(b/bao giờ) :degree(c/cũng))

Ở ví dụ trên, có thể thấy rằng GPT-4 và Gemini đưa ra kết quả gần tương tự nhau (chỉ khác quan hệ *:modality* và *:degree* cho từ "cũng"). Tuy nhiên, ViBART là mô hình đúng hơn với hai vai nghĩa chính: *:pivot* (tôi) và *:theme* (sa mạc).

Bảng 3.17: Kết quả sinh biểu diễn ngữ nghĩa tiếng Việt.

STT	Mô hình	Prompt	Smatch (F_1)	Số câu lỗi
1	ViBART	-	55.90%	0
2	GPT-4	Zero-shot	10%	16
		One-shot	47.88%	0
		5-shot	55.36%	0
		Few-shot	53.25%	0
3	Gemini	Zero-shot	16%	12
		One-shot	46.44%	1
		5-shot	57.72%	0
		Few-shot	54.90%	1

Bảng 3.17 mô tả cụ thể điểm F_1 mà mỗi mô hình đạt được. Có thể thấy rằng, Gemini là mô hình hoạt động khá tốt cho tất cả các kịch bản (zero-shot, one-shot, 5-shot và few-shot) với độ chính xác cao nhất đạt được là 57.72%. ViBART đang là mô hình đứng thứ 2, với độ chính xác là 55.90%, và cuối cùng là GPT-4 với độ đo cao nhất đạt được là 55.36%. Mô hình BART đã hoạt động khá tốt với tiếng Anh, tuy nhiên lại có kết quả chưa cao với tiếng Việt. Điều này có thể do một vài yếu tố: số lượng dữ liệu chưa nhiều, dữ liệu chưa tốt (chưa có sự đồng bộ giữa các chuyên gia gán nhãn), số lượng nhãn quá nhiều khiến cho mô hình học chưa được tốt, mô hình chưa phù hợp với các đặc trưng của tiếng Việt, ... Một số kịch bản, chẳng hạn như GPT-4 và Gemini (zero-shot), đã tạo ra kết quả với khá nhiều câu lỗi. Các lỗi này chủ yếu xoay quanh việc không thể sinh cây phân tích ngữ nghĩa hoàn chỉnh hoặc lặp lại các nút quá nhiều lần trong cây. Nguyên nhân có thể xuất phát từ việc dữ liệu huấn luyện chưa bao phủ đầy đủ các nhãn ngữ nghĩa cần thiết, dẫn đến hạn chế trong khả năng nhận diện và biểu diễn các cấu trúc phức tạp. Ngoài ra, cần tập trung nghiên cứu cách viết hướng dẫn chi tiết và rõ ràng hơn, nhằm giúp mô hình hiểu đúng đầu vào và đầu ra, cũng như các yêu cầu cụ thể trong việc sinh ra các phân tích ngữ nghĩa chính xác. Có thể thấy rằng đây là một kết quả khiêm tốn, tuy nhiên đây cũng là một bước khởi đầu khả quan cho các nghiên cứu sau này về mô hình biểu diễn và phân tích ngữ nghĩa cho tiếng Việt.

3.4. Kết luận chương 3

Chương này đã trình bày quá trình xây dựng kho ngữ liệu gán nhãn vai nghĩa và mô hình chú giải ngữ nghĩa cho tiếng Việt theo hướng tiếp cận liên ngữ. Kho ngữ liệu đã xây dựng được bao gồm 2,570 câu có gán nhãn vai nghĩa và 1,570 câu có gán nhãn ngữ nghĩa. Bên cạnh đó, luận án đã phát triển một công cụ hỗ trợ gán nhãn ngữ nghĩa và thử nghiệm xây dựng mô hình phân tích ngữ nghĩa cho tiếng Việt, đồng thời so sánh kết quả với các mô hình ngôn ngữ lớn hiện có. Mặc dù kết quả đạt được còn khiêm tốn, nhưng đây là mô hình biểu diễn ngữ nghĩa sâu đầu tiên dành riêng cho tiếng Việt. Kho ngữ liệu gán nhãn và các mô hình được phát triển trong luận án sẽ là nền tảng quan trọng cho các nghiên cứu sâu hơn về ngữ nghĩa trong tiếng Việt.

Chương 4
XÂY DỰNG MẠNG ĐỘNG TỪ TIẾNG VIỆT

Mạng động từ VerbNet [68] là một tài nguyên từ vựng được phát triển nhằm mô tả chi tiết cấu trúc ngữ pháp và ngữ nghĩa của các động từ trong tiếng Anh. Tài nguyên này không chỉ phân loại các động từ dựa trên các thuộc tính ngữ nghĩa và cú pháp mà còn bao gồm các khía cạnh về ngữ cảnh sử dụng, giúp tạo ra một hệ thống phân loại động từ chặt chẽ và có thể áp dụng cho nhiều mục đích nghiên cứu và ứng dụng NLP. VerbNet đã đóng vai trò nền tảng trong các lĩnh vực như phân loại tài liệu [70], xác định cảm xúc, phân tích và gán nhãn vai nghĩa [108, 44], tạo sinh ngôn ngữ [37], ... Hiện nay, VerbNet không chỉ được phát triển và áp dụng cho tiếng Anh mà còn mở rộng ra các ngôn ngữ khác như tiếng Ả Rập[1], tiếng Pháp [33], tiếng Tây Ban Nha[2], ... nhằm hỗ trợ các nghiên cứu đa ngữ và mở rộng phạm vi ứng dụng của các hệ thống NLP trên toàn cầu.

Tuy nhiên, đối với tiếng Việt, vẫn chưa có một kho ngữ liệu động từ nào với đầy đủ các thông tin. Sau khi đã xây dựng và hoàn thiện các kho ngữ liệu nền tảng như cú pháp và ngữ nghĩa, việc xây dựng một mạng động từ dành riêng cho tiếng Việt là vô cùng cấp thiết. Một mạng động từ như vậy sẽ không chỉ đóng vai trò như một nguồn tài nguyên ngữ liệu để phục vụ nghiên cứu mà còn là công cụ quan trọng trong việc phát triển các hệ thống tự động về phân tích ngữ nghĩa, hỗ trợ dịch máy, và tương tác người máy và nhiều tác vụ khác trong xử lí ngôn ngữ tiếng Việt. Ở chương này, luận án sẽ trình bày về việc xây dựng mạng động từ cho tiếng Việt (*viVerbNet*), cụ thể là các công việc sau: khảo sát các kho ngữ liệu từ vựng và ngữ nghĩa đã có của tiếng Việt, so sánh sự tương đồng đối với VerbNet, sau đó trích rút các động từ tiếng Việt, thực hiện các thuật toán phân cụm và xây dựng các thành phần của viVerbNet.

Các kết quả của chương này đã được công bố trong các bài báo [P3, P5] trong "Danh mục công trình công bố" của luận án.

[1] `https://github.com/JaouadMousser/Arabic-Verbnet`
[2] `https://clic.ub.edu/corpus/en/ancoranet`

4.1. Từ điển tiếng Việt cho máy tính VCL

Đối với tiếng Việt, từ điển tiếng Việt cho máy tính VCL [94] là một tài nguyên từ vựng hữu ích trong nghiên cứu ngữ pháp, ngữ nghĩa. Để xây dựng mạng động từ cho tiếng Việt, việc khảo sát các thông tin ngữ nghĩa hiện có của VCL và so sánh với VerbNet là rất cần thiết. Một số điểm quan trọng khi so sánh hai kho từ vựng này có thể được nêu rõ như sau:

- Về cách thức tổ chức: VCL mô tả từng mục từ và các thông tin mô tả của từ đó trên ba phương diện: hình thái học, cú pháp học và ngữ nghĩa học. Trong khi đó, VerbNet mô tả động từ và phân thành các lớp dựa vào đặc trưng về cấu trúc ngữ pháp, thông tin ngữ nghĩa và các ràng buộc đi kèm.
- Về các thông tin biểu diễn:
  - Thông tin hình thái (*Morphology*): VCL tập trung vào mô tả rõ các thông tin hình thái cho các từ. Từ tiếng việt trong cấu tạo không có căn tố và phụ tố, trong ngữ nghĩa không có các ý nghĩa thuộc phạm trù hình thái như giống, số, cách, không có sự biến hình và biểu hiện bằng trật tự từ. Vì thế khi xét về thông tin hình thái thường chỉ xét về vấn đề cấu tạo từ. Các dạng cấu tạo từ của tiếng Việt được phân thành từ đơn, từ ghép, từ láy, từ vay mượn, từ viết tắt và kí hiệu. Thông tin hình thái không được chú trọng và mô tả trong VerbNet, các mục từ trong VerbNet đều là động từ và được xét dưới dạng nguyên thể.
  - Thông tin cú pháp (*Syntactics*): VCL mô tả thông tin về 13 loại từ như danh từ, động từ, tính từ, ... và thông tin về 29 tiểu loại từ. VerbNet chỉ tập trung vào các động từ và không có tiểu loại từ.
  - Thông tin về khung vị từ (*Semantic Frames*): VCL tập trung xây dựng thông tin về mẫu động từ (*verb pattern*) mà chưa quan tâm đến các loại vị từ khác như tính từ, giới từ. VCL mô tả thông tin gồm 12,887 nghĩa của động từ. Các khung vị từ trong VCL gồm có 20 mẫu. Ví dụ về một số mẫu như sau:

    `Sub+V`: động từ không đòi hỏi bổ ngữ như "Chim bay", "Bé đang ngủ".

    `Sub+V+Obj`: động từ đòi hỏi một bổ ngữ như "Tôi đọc sách", "Nó ngồi xuống bàn".

    `Sub+V+Obj+Obj`: động từ đòi hỏi hai bổ ngữ như "Tôi tặng hoa cho mẹ", "Họ gọi ông là vị thánh sống".

Đối với thông tin về khung vị từ, VerbNet mô tả 8,537 nghĩa của động từ, chia thành 273 lớp, được ánh xạ tới 5,649 động từ trong khung PropBank, 4,186 khung của FrameNet và 4,898 nhóm ngữ nghĩa. VerbNet định nghĩa 309 loại khung cú pháp với tổng số 1,513 lần sử dụng các khung đó. Mỗi khung gồm các thông tin cú pháp, ngữ nghĩa, các vai nghĩa và ràng buộc ngữ pháp, ngữ nghĩa đi kèm. Một số loại động từ được mô tả như động từ nội động, ngoại động, cảm xúc, ... Các khung trong VerbNet bao gồm cấu trúc cú pháp, câu ví dụ và các vai nghĩa ánh xạ đến các tham tố cú pháp. Ví dụ như:

`Frame: NP V NP ADJ`

`Example: ‘‘The belt came undone’’`

`Syntax: Patient V Result`

`Semantics: state(result(E), Result, Patient)Pred(result(E), Patient)`

Có thể thấy rằng, số lượng các khung vị từ của VCL là khá hạn chế và chưa có bất cứ nghiên cứu nào đề cập để chứng minh số lượng 20 khung là đầy đủ cho các ngữ cảnh của tiếng Việt. Hầu hết các động từ chỉ được gắn vào một khung và thông tin của mỗi khung cũng chưa đầy đủ về mặt cú pháp và ngữ nghĩa.

– Thông tin vai nghĩa (*Semantic roles*): VCL định nghĩa 16 loại vai nghĩa[3] như Agent, Expriencer, Possessor, Patient, ... Bộ vai nghĩa này khá hạn chế khi so với 39 vai nghĩa trong VerbNet. Tập vai nghĩa được sử dụng trong từ điển VCL cũng không tương đương với các vai nghĩa trong VerbNet. Ví dụ:

  * VCL sử dụng vai nghĩa Content (Nội dung) - để biểu thị nội dung mà vị từ nói đến. Tuy nhiên vai nghĩa này có ranh giới rất khó đoán nhận khi và khi ánh xạ, nó được chia thành các vai cụ thể hơn trong VerbNet: Agent (chủ đề), Cause (nguyên nhân), Goal (mục tiêu), Purpose (mục đích), Result (kết quả), Manner (cách thức).
  * Trong nhiều trường hợp, các vai nghĩa trong VCL và VerbNet được định nghĩa khác nhau mặc dù trong cùng một ngữ cảnh. Ví dụ: `‘‘Tôi yêu mẹ’’`. Trong VCL, đối tố của chủ ngữ (`tôi`) của động từ “yêu (love)” được chỉ định là Agent (tác thể), trong khi trong VerbNet nó

[3] `https://vlsp.hpda.vn/demo/vcl/SemanticRole.htm`

được gắn nhãn là Experiencer (người trải nghiệm).

– Thông tin về ràng buộc lựa chọn (*Selectional restrictions*): dùng để xác định các ràng buộc ngữ nghĩa trên các tham tố của động từ, tức là các loại thực thể mà động từ có thể hoặc không thể kết hợp được. Những giới hạn này giúp làm rõ các vai trò tham tố dựa vào thuộc tính ngữ nghĩa của chúng, như con người, vật thể, sự kiện, trạng thái.

  Trong VCL, mỗi đơn vị từ vựng ngoài việc được gán nhãn từ loại ngữ pháp (học sinh – Nc) còn được gán thêm một nhãn từ loại ngữ nghĩa (học sinh – Person). Có hai loại ngữ nghĩa lớn, một loại biểu thị thực thể (thể từ) và một loại biểu thị thuộc tính của thực thể hoặc thuộc tính của thuộc tính (gọi là thuộc từ - mang ý nghĩa trừu tượng). Thông tin về loại ngữ nghĩa này chính là khái niệm ràng buộc lựa chọn. VCL tổ chức từ loại ngữ nghĩa theo mô hình quan hệ hình cây, gồm gần 100 tiểu loại[4]. Ví dụ các loại ràng buộc gồm có: Living thing (*People, Animal, FictionalAnimal, Microorganism, Plant, ...*), Non-living thing (*Food, Artifact, Part, Substance, ...*).

  Đối với VerbNet, ràng buộc lựa chọn được sử dụng gồm có 36 loại chính với tổng số 957 lần sử dụng. Các ràng buộc cũng gồm một số loại phổ biến như: Abstract (trừu tượng), Animal (động vật), Body_part (các bộ phận cơ thể), force (lực), machine (máy móc), ... VerbNet cung cấp một bộ giới hạn lựa chọn khá tổng quát, thường chỉ tập trung vào mối quan hệ động từ - tham tố. VCL thì ngược lại, cung cấp một phân loại ngữ nghĩa chi tiết hơn, áp dụng không chỉ cho động từ mà còn cho các loại từ khác.

– Thông tin ràng buộc cú pháp (*Syntax restrictions*): mô tả thông tin ràng buộc về cú pháp cho các thành phần trong câu khi một động từ cụ thể được sử dụng. Thông tin về ràng buộc cú pháp không được mô tả trong VCL một cách cụ thể. Tuy nhiên, đối với VerbNet, các ràng buộc cú pháp được mô tả đầy đủ gồm có 40 loại, với 532 lần sử dụng. Các loại cơ bản gồm có: động từ nguyên thể (ac_to_inf), động từ thêm ing (ac_ing), trạng từ chỉ vị trí (adv_loc), ... Các ràng buộc cú pháp trong VerbNet giúp mô hình hóa cú pháp của câu bằng việc xác định rõ cấu trúc cú pháp có thể hoặc không thể xảy ra với một động từ cụ thể.

[4] `https://vlsp.hpda.vn/demo/vcl/SemanticTree.htm`

– Thông tin vị từ ngữ nghĩa (*Semantic Predicates*): mô tả ý nghĩa của động từ bằng cách mô hình hoá các hành động, sự kiện và trạng thái mà động từ diễn tả. Thông tin này cũng không được làm rõ trong VCL. Đối với VerbNet, có tới 146 loại vị từ ngữ nghĩa được định nghĩa, với 3,222 lượt sử dụng.

  Như trong ví dụ trên, vị từ ngữ nghĩa được thể hiện là: `Semantics: state(result(E), Result, Patient)Pred(result(E), Patient)`. Có thể giải thích rằng, "become" là một trạng thái (*state*) của một bị thể (*patient*), và có mang tới một kết quả nào đó (*result*).

Các khảo sát cho thấy rằng VCL và VerbNet được thiết kế khác nhau đáng kể, từ cách tổ chức đến thông tin về ngữ pháp và ngữ nghĩa. Việc xây dựng một mạng động từ cho tiếng Việt là cần thiết, và VerbNet sẽ được chọn làm cơ sở lý thuyết. ViVerbNet sẽ được phát triển theo cấu trúc tương tự như VerbNet, bao gồm đầy đủ các thành phần: lớp động từ, các lớp con, vai nghĩa, khung cú pháp, ví dụ, ràng buộc lựa chọn và ngữ nghĩa, cùng các vị từ ngữ nghĩa.

Để xây dựng ViVerbNet, thông tin từ VCL có thể được tận dụng và phát triển thêm. Bước đầu sẽ dựa vào thông tin từ loại để trích xuất các động từ từ VCL. Sau đó, sẽ thu thập các thông tin tương ứng như ví dụ, khung vị từ, cú pháp, ngữ nghĩa, và ràng buộc ngữ nghĩa. Tiếp đến, các động từ sẽ được phân loại thử nghiệm thành các nhóm, và cấu trúc 5 thành phần của VerbNet sẽ được xây dựng cho những nhóm động từ tiếng Việt. Bên cạnh các thông tin từ VCL, có thể sử dụng cú pháp thành phần và cú pháp phụ thuộc dựa trên các động từ và ví dụ đi kèm để mô tả khung, ràng buộc lựa chọn và ngữ nghĩa, cũng như thông tin về vị từ tương ứng.

4.2. Phương pháp xây dựng viVerbNet

Phương pháp xây dựng viVerbNet được mô tả cụ thể trong Hình 4.1.
Các bước thực hiện được mô tả chi tiết như sau:

- Phân cụm động từ tiếng Việt: các động từ sẽ được trích xuất từ VCL, sau đó tìm kiếm trong kho ngữ liệu để lấy ngữ cảnh. Khi đã có ngữ cảnh của các động từ, sử dụng một số mô hình biểu diễn véc-tơ từ (*word embedding*) đã được huấn luyện bằng các ngữ liệu tiếng Việt để sinh véc tơ từ cho các động từ này. Các véc tơ từ sẽ được sử dụng là đầu vào cho các thuật toán phân cụm. Một số thuật toán phân cụm được sử dụng như K-means [48],

Hình 4.1: Mô hình xây dựng viVerbNet.

phân cụm phân cấp HCA [32]. Sau đó các cụm động từ thu được sẽ được đánh giá và loại bỏ những động từ không nằm trong nhóm.

- Xây dựng các thành phần của từng cụm động từ: giai đoạn này sẽ kết hợp thông tin ngữ pháp và ngữ nghĩa tiếng Việt, các kho ngữ liệu chú giải cú pháp thành phần và phụ thuộc tiếng Việt để xây dựng các thành phần của một cụm động từ. Các thành phần của một cụm động từ gồm có: vai nghĩa (*Thematic Roles*), các ràng buộc lựa chọn và cú pháp (*Selectional Restriction; Syntax Restriction*), khung cú pháp (*Syntactic Frame*) và vị ngữ ngữ nghĩa (*Semantic Predicate*).

Những phần sau sẽ mô tả chi tiết về các bước xây dựng viVerbNet cho tiếng Việt.

4.2.1. Biểu diễn véc-tơ từ

Biểu diễn véc-tơ từ là cách biểu diễn một từ thành một véc tơ nhiều thành phần giá trị thực, giúp cho máy tính có thể hiểu và xử lý các thông tin liên quan tới từ đó. Cụ thể, các từ sẽ được biểu diễn dưới dạng một véc tơ có số chiều cố định, mỗi chiều của véc tơ thể hiện một đặc điểm nào đó của từ. Các biểu diễn véc-tơ từ thường được học từ dữ liệu văn bản lớn bằng cách sử dụng các mô hình như Word2vec [87], Glove [100], FastText [16] hoặc các mô hình mạng

nơ-ron sâu như các mạng nơ-ron tái lập (*auto encoders*) hoặc mạng nơ-ron biến đổi (*transfomers*) (các mô hình này đã được trình bày chi tiết trong phần 1.2.3).

Véc tơ biểu diễn từ được sử dụng là đầu vào cho bài toán phân cụm động từ tiếng Việt. Các mô hình biểu diễn véc-tơ từ được sử dụng trong thuật toán phân cụm gồm có:

1. Mô hình Word2vec [87]: 2 mô hình Word2vec được sử dụng trong thử nghiệm gồm có:

   - Word2vec1: mô hình biểu diễn véc-tơ từ tự huấn luyện. Kho ngữ liệu gồm có 1 triệu câu đã được thu thập, tiền xử lí và huấn luyện để tạo word2vec1.
   - Word2vec2: mô hình biểu diễn véc-tơ từ được huấn luyện trước của tác giả Vũ Xuân Sơn và cộng sự [119].

2. Mô hình PhoBERT [31]: PhoBERT là một mô hình ngôn ngữ được phát triển dựa trên BERT (*Bidirectional Encoder Representations from Transformers*) và được tinh chỉnh, huấn luyện trên khoảng 20GB dữ liệu tiếng Việt, sử dụng VNCoreNLP để tách từ cho dữ liệu đầu vào.

3. Mô hình ngôn ngữ lớn Gemini[5]: trong số các mô hình ngôn ngữ lớn, Gemini là một trong những mô hình tiên tiến và có khả năng xử lí dữ liệu đa phương thức và đa ngôn ngữ. Vì thế, Gemini được lựa chọn để trích rút các véc tơ từ, làm đầu vào cho thuật toán phân cụm.

4.2.2. Phân cụm động từ tiếng Việt

Đối với bước phân cụm động từ, hai thuật toán được sử dụng là K-means và HCA. Một số kịch bản đã được thử nghiệm để có thể đưa ra kết quả phân cụm tốt nhất. Các kịch bản được mô tả chi tiết như sau:

- Kịch bản 1: Sử dụng 2 mô hình Word2vec hoặc Gemini

  1. Trích rút các động từ từ kho từ vựng VCL
  2. Sử dụng các mô hình Word2vec hoặc Gemini để trích rút véc tơ của các động từ, sau đó đưa vào phân cụm (với số cụm khi chạy K-means là k = 500, 1000, 1500, 2000, 2500 và 3000).

[5] `https://deepmind.google/technologies/gemini/`

3. So sánh và phân tích kết quả đạt được, đưa ra kết quả các cụm tốt nhất.

- Kịch bản 2: Sử dụng PhoBERT

1. Trích rút các động từ từ kho từ vựng VCL
2. Tìm kiếm các câu trong kho ngữ liệu chứa các động từ trên, mỗi động từ lấy 10 câu (làm ngữ cảnh), đồng thời trích rút các ví dụ trong VCL tương ứng với các động từ.
3. Sử dụng PhoBERT để trích rút véc tơ của các động từ với ngữ cảnh đó, sau đó đưa vào phân cụm (với số cụm khi chạy K-means là k = 500, 1000, 1500, 2000, 2500 và 3000).
4. So sánh và phân tích kết quả đạt được, đưa ra kết quả các cụm tốt nhất.

Ngoài việc sử dụng 2 kịch bản trên, một số các trường hợp khác về số cụm, dịch và ánh xạ các cụm tiếng Việt sang tiếng Anh, ... cũng được phát triển để có thể tìm ra các cụm tốt nhất. Một số kĩ thuật tìm số cụm tốt nhất cũng được thử nghiệm cho thuật toán K-means, như Elbow [6], Silhouette [107]. Kết quả nhận được sau khi sử dụng thuật toán HCA trên các bộ biểu diễn véc tơ cho các động từ là 283 cụm. Tuy nhiên, sau khi khảo sát về số cụm 283, kết quả cho thấy có rất nhiều cụm không thoả mãn điều kiện. Vì thế để phân nhóm động từ một cách hiệu quả hơn, số cụm 1000 đã được lựa chọn. Trong quá trình thử nghiệm thuật toán phân cụm các từ vựng theo đặc tính ngữ pháp – ngữ nghĩa, giá trị $k = 1000$ cho thấy mức độ phân nhóm tối ưu hơn so với các giá trị khác. Khi k quá nhỏ, các cụm tạo thành thường quá rộng, bao gồm nhiều từ có tính chất ngữ pháp hoặc ngữ nghĩa khác biệt, dẫn đến giảm độ đồng nhất nội tại của từng cụm. Ngược lại, khi k quá lớn, các cụm trở nên quá nhỏ và rời rạc, gây phân mảnh thông tin, khiến mô hình khó rút ra được các mẫu thống nhất có ý nghĩa. Giá trị $k = 1000$ đạt được sự cân bằng tương đối giữa độ chi tiết và tính khái quát, vừa đảm bảo phân biệt tốt các nhóm từ vựng có chức năng ngôn ngữ khác nhau, vừa giữ được tính ổn định trong việc phân tích và gán nhãn sau đó. Các cụm động từ sẽ có cùng một số tính chất như cùng khung cú pháp, cùng một số vai nghĩa, ... vì thế các cụm này sẽ được dùng để xác định các lớp động từ cho viVerbNet.

[6] `https://en.wikipedia.org/wiki/Elbow\_method\_(clustering)`

Bảng 4.1 là kết quả phân cụm khi sử dụng các mô hình biểu diễn véc tơ từ khác nhau.

Bảng 4.1: Ví dụ về các động từ trong cụm khi sử dụng đầu vào khác nhau.

Các động từ có nghĩa "sinh"			
word2vec1	**word2vec2**	**PhoBERT**	**Gemini**
đẻ	đẻ	đẻ	sinh_sản
sinh_nở	sinh_nở	sinh_sản	sinh_nở
sinh_đẻ	sinh_đẻ	sinh	sinh_đẻ
chuyển_dạ	chuyển_dạ		sinh_sôi
Các động từ có nghĩa "chết"			
word2vec1	**word2vec2**	**PhoBERT**	**Gemini**
bị_thương	bị_thương	đi	cố_chết
thiệt_mạng	thiệt_mạng	chết_chẹt	
mất_tích	mất_tích		
tử_nạn	tử_nạn	tử_vong	chết_non
chết	chết	chết	chết_hụt
chết_đuối	chết_đuối	chết_đuối	chết_đuối
bỏ_mạng	lâm_nạn		tắt_thở
mắc_kẹt	thương_vong		chết_yểu

Kết quả này cho thấy hai mô hình Word2vec1 và Word2vec2 cho kết quả tương tự nhau. Điều này có thể giải thích được vì hai mô hình được huấn luyện với cùng một thuật toán và các kho ngữ liệu được huấn luyện có cùng văn phong (các bài báo, các tác phẩm văn học). Gemini là mô hình có các cụm chưa đồng nhất về cả ý nghĩa và cú pháp, ngữ nghĩa. Trong khi đó, mô hình PhoBERT cho kết quả khác biệt nhất so với các mô hình còn lại, vì mỗi động từ sẽ được biểu diễn bằng một véc tơ khác nhau trong các ngữ cảnh khác nhau. Điều này rất hữu ích để nắm bắt các nghĩa cho từng động từ. PhoBERT có thể nhận định được từ đa nghĩa "đi" xếp nó vào cụm từ mang nghĩa "chết". Các cụm thu được khi sử dụng PhoBERT tốt hơn hẳn dựa theo các tiêu chí về: Khung cú pháp, nghĩa, khung ngữ nghĩa và các ràng buộc.

Với các kịch bản trên, việc đánh giá kết quả phân cụm dựa vào các thông tin sau: số động từ trong cụm, số động từ thực tế thuộc cụm. Một cụm được coi là đạt nếu có hơn 60% số động từ thực tế thuộc cụm đó. Chi tiết đánh giá về kết quả phân cụm được mô tả trong Bảng 4.2.

Ví dụ một số lớp con thu được từ việc phân cụm sử dụng Kmeans và PhoBERT như sau:

Bảng 4.2: Kết quả đánh giá các thuật toán phân cụm.

Mô hình	Số cụm	Số động từ trung bình	Số động từ thực tế	Tỷ lệ
Kmeans + word2vec1	200	53.4	7.3	41.5%
Kmeans + word2vec2	200	68.4	9.7	43.5%
Kmeans + PhoBERT	200	7.8	4.6	61.0%
Kmeans + Gemini	200	13.8	3.7	18.6%

- Lớp con 36: *bằng lòng, hài lòng, ưng thuận, ưng ý, vừa lòng*: các động từ thuộc nhóm này đều có chung nghĩa "cảm thấy bằng lòng vì hợp ý mình".
- Lớp con 62: *chạy chữa, chữa, chữa trị, cứu chữa* đều mang nghĩa là "làm cho khỏi bệnh hoặc hết hư hỏng".

Tuy nhiên, trong kết quả vẫn còn xuất hiện những lớp từ chưa có đặc điểm tương đồng về nghĩa và ngữ pháp, ngữ nghĩa. Ví dụ:

- Lớp con 258: *học, học hành, học tập, khoa, luyện, luyện tập, ôn, ôn luyện, tập, tập luyện, thi, tốt nghiệp*. Các động từ trong lớp 258 này đều có các đặc điểm về nghĩa là "học hỏi, trau dồi, tiếp thu, ôn luyện một kiến thức nào đó, ...". Nhưng trong lớp động từ con này vẫn có các động từ mang nghĩa khác như: "tốt nghiệp", "thi".
- Lớp con 76: *bồi dưỡng, dạy, dạy học, huấn luyện, làm, phụ đạo, rèn, thực hành*. Từ "làm" và "thực hành" không có chung đặc điểm về nghĩa với các từ còn lại.

Nhìn chung, các lớp động từ sau khi phân cụm đã có chung những đặc điểm tương đồng về nghĩa. Tuy nhiên, vẫn tồn tại một số lỗi như một số động từ trong cụm không cùng nghĩa hoặc khung cú pháp, vì thế các lớp động từ sau khi phân cụm tự động cần phải trải qua thao tác điều chỉnh, sửa đổi của các chuyên gia ngôn ngữ. Do hạn chế về mặt thời gian, việc lựa chọn, chỉnh sửa và xây dựng lớp được thực hiện cho 100 nhóm động từ cơ bản của tiếng Việt. Những lớp con có đặc điểm tương đồng về nghĩa sẽ được xếp vào trong các lớp quan hệ phân cấp. Các cụm sẽ được đặt tên theo đặc điểm nghĩa chung nhất của các động từ thành viên xuất hiện trong nhóm đó. Các lớp con và lớp nằm trong nhóm sẽ được đánh số để biểu diễn quan hệ thứ bậc.

Một nhóm động từ biểu thị một nghĩa nào đó có thể gồm một hoặc nhiều lớp động từ khác nhau. Đồng thời, các lớp động từ cũng có thể bao gồm các nhóm

con khác bên trong. Điều này tạo nên sự phân cấp của các nhóm động từ. Sự phân cấp đó được biểu diễn minh họa qua hai nhóm động từ ở Bảng 4.3.

Bảng 4.3: Một số nhóm động từ tiếng Việt.

STT	**Nhóm động từ**	**Lớp**	**Lớp con**	**Chi tiết**
1	Nghĩa "***học***"	học-1	học-1.1	học, học hành, học tập, luyện tập, luyện, ôn, ôn luyện, ôn tập, tập luyện
			học-1.2	bắt chước, học lỏm, nhái, nhại
			học-1.3	học hỏi, chắt lọc, định hình, lĩnh hội, lãnh hội, rèn giũa, mài giũa, thu nạp, tích lũy, tiếp thụ, trau dồi
6	Nghĩa "***giữ***"	giữ-6.1	giữ-6.1-1	gìn giữ, giữ gìn
			giữ-6.1-2	đúc kết, lưu giữ, lưu truyền

Một động từ có thể mang trong mình nhiều nghĩa khác nhau và nó có thể nằm trong các lớp từ con khác nhau ở các nhóm động từ hoàn toàn khác nhau. Theo Từ điển tiếng Việt của Hoàng Phê, từ "đi" mang 22 nghĩa. Trong đó có 17 nghĩa là động từ, 1 nghĩa phụ từ và 3 nghĩa cảm từ [7]. Các nghĩa của động từ "đi" cũng có sự phân bố khá đa dạng trong các lớp con động từ nói riêng và các nhóm động từ nói chung. Bảng 4.4 mô tả sự phân bố của từ "đi" trong viVerbNet.

Công việc tiếp theo sau khi hoàn thành phân cụm là xây dựng 5 thành phần của mỗi lớp động từ.

4.2.3. Xây dựng các thành phần của viVerbNet

Các thành phần của viVerbNet được thiết kế dựa vào VerbNet tiếng Anh, gồm có: Vai nghĩa, ràng buộc lựa chọn, khung cú pháp và ràng buộc ngữ nghĩa, vị từ ngữ nghĩa.

[7] `https://vlsp.hpda.vn/demo/?page=vcl`

Bảng 4.4: Sự phân bố của động từ "đi" trong viVerbNet

STT	**Nghĩa**	**Nhóm động từ**	**Lớp**	**Lớp con**
1	[người, động vật] tự di chuyển từ chỗ này đến chỗ khác bằng những bước chân nhấc lên, đặt xuống liên tiếp	Nhóm từ mang nghĩa chuyển động	chạy-14.1	chạy-14.1-1
2	[người] di chuyển đến nơi khác, không kể bằng cách gì, phương tiện gì	Nhóm từ mang nghĩa chuyển động	chạy-14.2	chạy-14.1-1
3	chết [lối nói kiêng tránh]	Nhóm từ mang nghĩa chết	chết-18	chết-18.2
4	di chuyển đến chỗ khác, nơi khác để làm một công việc nào đó	Nhóm từ mang nghĩa chuyển động	chạy-14.1	chạy-14.1-1
5	[phương tiện vận tải] di chuyển trên một bề mặt	Nhóm từ mang nghĩa chuyển động	chạy-14.1	chạy-14.1-1
9	chuyển vị trí quân cờ để tạo ra thế cờ mới [trong chơi cờ]			
10	biểu diễn, thực hiện các động tác võ thuật	Nhóm động từ mang nghĩa biểu diễn	biểu_diễn-72	
11	làm, hoạt động theo một hướng nào đó			
13	chuyển sang, bước vào một giai đoạn, một trạng thái khác	Nhóm động từ mang nghĩa bắt đầu	bắt đầu-86	
14	đem đến tặng nhân dịp lễ tết, hiếu hỉ	Động từ mang nghĩa thay đổi sở hữu	đưa-52	đưa-52.1
15	mang vào chân hoặc tay để che giữ, bảo vệ	Nhóm động từ mang nghĩa mặc (quần áo, ...)	mặc-97	
16	gắn với nhau, phù hợp với nhau			
17	đi ngoài [nói tắt]	Nhóm động từ liên quan đến quá trình của cơ thể	đại tiện-15	

4.2.3.1. Vai nghĩa

Các vai nghĩa diễn tả mối quan hệ ngữ nghĩa cơ bản giữa các vị từ và tham tố của chúng. Trong Propbank tiếng Việt [2], một tập vai nghĩa gồm 24 nhãn đã được phát triển, làm cơ sở để xây dựng các vai nghĩa cho viVerbNet. Quá trình chuyển đổi từ 24 vai nghĩa này sang 39 vai nghĩa của VerbNet diễn ra thuận lợi nhờ cách định nghĩa chi tiết và cụ thể của VerbNet [78]. Hơn nữa, sau khi tham khảo bộ vai nghĩa chuẩn hóa LIRICS [98], luận án đã xây dựng một tập vai nghĩa cốt lõi riêng cho tiếng Việt dựa trên LIRICS. Bộ vai nghĩa của viVerbNet bao gồm 29 vai, tương đồng với số vai nghĩa được sử dụng trong mô hình biểu

diễn ngữ nghĩa cho tiếng Việt đã mô tả ở Chương 3. Một ví dụ về câu trong viVerbNet được gán nhãn vai nghĩa như sau:

`Hắn đánh tôi`

`Agent V Patient`

Động từ "đánh" có nghĩa "làm cho đau, cho tổn thương bằng tác động của một lực lên cơ thể" tức là tham thể "Hắn" sẽ là người tác động được gán nhãn *Agent*; tham thể "tôi" là người bị tác động sẽ được gán nhãn *Patient*.

Một số vai nghĩa được luận án sử dụng trong viVerbNet gồm có:

- Agent (Tác nhân/Tác thể): Người chủ động của một hành động hoặc sự kiện. Ví dụ:

  ''Hắn$_{\texttt{Agent}}$ đánh tôi''.

- Asset (Một đối tượng cụ thể): Vai nghĩa chỉ giá trị là một đối tượng, tân ngữ cụ thể. Vai nghĩa này biểu thị giá trị của tham thể trước nó nhưng lại không có mối quan hệ với động từ chính của câu. Vai nghĩa này lưỡng phân với Value (Giá trị). Ví dụ:

  ''Tôi mua cái túi 20 nghìn$_{\texttt{Asset}}$''.

- Attribute (Thuộc tính): Vai nghĩa chỉ thực thể mang một tính chất nào đó được biểu thị trong câu. Ví dụ:

  ''Giá_dầu$_{\texttt{Attribute}}$ tăng vọt''.

- Beneficiary (Người hướng lợi): Vai nghĩa biểu thị kẻ được hưởng thành quả (lợi ích, bất lợi) từ một hành động do một ai đó thực hiện hoặc trạng thái, sự việc. Ví dụ:

  ''Tôi tặng mẹ$_{\texttt{Beneficiary}}$ hoa''.

- Cause (Nguyên nhân): Vai nghĩa chỉ hành thể khởi xướng sự việc, hành động nào đó. Nó tồn tại độc lập và không phụ thuộc vào sự kiện. Ví dụ:

  ''Thầy_giáo$_{\texttt{Cause}}$ giới hạn đề tài''.

- Destination (Điểm đến): Đích đến của hành động là một vị trí địa lí cụ thể. Ví dụ:

  ''Nó đi vào nhà$_{\texttt{Destination}}$''

- Experiencer (Nghiệm thể): Vai nghĩa được đặc trưng là nhận thức hoặc trải nghiệm điều gì đó. Ví dụ:

  ‘‘Tôi$_{\text{Experiencer}}$ mệt.’’.

- Extent (Mức độ): Giá trị cho biết mức độ thay đổi có thể đo lường được của các thành phần tham gia trong suốt quá trình diễn ra sự kiện. Ví dụ:

  ‘‘Giá dầu tăng 10%$_{\text{Extent}}$’’.

- Goal (Đích thể): Mục đích cuối cùng của hành động tồn tại độc lập với sự việc. Ví dụ:

  ‘‘Tôi đốt lửa để sưởi_ấm$_{\text{Goal}}$’’.

- Initial_Location (Địa điểm bắt đầu): Vai nghĩa này chỉ ra vị trí một cách cụ thể nơi mà sự kiện bắt đầu hoặc một trạng thái thành sự thật. Ví dụ:

  ‘‘Tôi đi từ Hà_Nội$_{\text{Initial_Location}}$’’.

- Instrument (Công cụ): Vai nghĩa công cụ được sử dụng để tác động hoặc tạo ra sự thay đổi trong một thứ gì đó. Thường là một công cụ được sử dụng bởi một tác nhân có tính động để mang lại sự thay đổi. Ví dụ:

  ‘‘Tôi đào đất bằng xẻng$_{\text{Instrument}}$’’.

- Location (Địa điểm): Vai nghĩa địa điểm chỉ vị trí cụ thể hay hướng không gian của trạng thái hay hành động do vị từ đảm nhiệm, thường được nhận diện bằng một giới từ đằng trước. Ví dụ:

  ‘‘Cô ấy ăn trưa ở nhà$_{\text{Location}}$’’.

- Material (Vật chất): Vai nghĩa chỉ vật chất ban đầu, cái mà thay đổi thông qua sự việc thành các thực thể mới, cụ thể hoặc trừu tượng. Vai nghĩa thường được sử dụng với các động từ mang nghĩa sáng tạo, tạo lập... Ví dụ:

  ‘‘Chiếc túi này làm bằng vải$_{\text{Material}}$’’.

- Patient (Bị thể): Tham thể tham gia trải qua một quá trình hoặc bị ảnh hưởng bởi một hành động, nhấn mạnh vào thay đổi trạng thái. Ví dụ:

  ‘‘Bình đánh Nam$_{\text{Patient}}$’’.

- Predicate (Vị ngữ): Vai nghĩa biểu thị một sự kiện, trạng thái thứ hai, phụ thuộc vào sự kiện chính, được mô tả bằng động từ thứ hai trong câu. Ví dụ:

  ‘‘Mẹ bắt tôi ăn rau$_{\text{Predicate}}$’’.

- Pivot (Chủ thể khác): Vai nghĩa Pivot chỉ chủ thể của một trạng thái tinh thần hoặc cảm xúc. Ví dụ:

  ‘‘Tôi$_{\text{Pivot}}$ yêu cô ấy.’’

- Product (Sản phẩm): Vai nghĩa chỉ kết quả cuối cùng của hoạt động, kết quả của quá trình thay đổi. Ví dụ:

  ‘‘Nghệ nhân làm gốm$_{\text{Product}}$ bằng đất sét’’

Ngoài ra, có một số nhãn không trực tiếp tham gia vào việc biểu diễn mối quan hệ ngữ nghĩa trong khung vị ngữ, như Participant, Undergoer, Place, ... và một số nhãn có tính tương tự như Theme, Patient, Agent, bao gồm Co-Theme (Đồng thể chuyển động), Co-Patient (Đồng bị thể), và Co-Agent (Đồng tác thể). Những nhãn Co-* này tham gia vào câu với chức năng và tính chất ngang bằng với những nhãn tương ứng khác.

Ví dụ:

- Tôi với Lan chăm sóc Hoa.
  Agent with Co-Agent V Patient

- Tôi ăn cơm với cá
  Agent V Patient with Co-Patient

4.2.3.2. Ràng buộc lựa chọn

Ràng buộc ngữ nghĩa hay nói cách khác là ràng buộc lựa chọn là ràng buộc giúp giới hạn các nghĩa biểu hiện có thể kết hợp của các vai nghĩa trong lớp động từ, thể hiện các bản chất của vai nghĩa đó. Việc lựa chọn các ràng buộc ngữ nghĩa cũng phụ thuộc nhiều vào đặc điểm tri nhận của mỗi ngôn ngữ. Những ràng buộc này cho biết sự tồn tại (+) hoặc không tồn tại (-) của các thuộc tính ngữ nghĩa như [concrete], [animate], [organization], ... Các toán tử logic “| (hoặc)” và “& (và)” được sử dụng để kết hợp nhiều hạn chế.

Ví dụ, các ràng buộc lựa chọn của cụm động từ “cấm, đình_chỉ, tạm_hoãn, nghiêm_cấm” tương ứng với lớp động từ VerbNet “forbid-64.4” được thể hiện

như sau:

```
Agent [+animate |+organization]
Theme []
Recipient [+animate|+organization]
```

Các ràng buộc lựa chọn ràng buộc về thuộc tính của các vai nghĩa biểu thị. Xét trong ví dụ về lớp động từ "forbid-64.4", các vai nghĩa Agent, Patient đều được giới hạn bởi hai ràng buộc [+animate] và [+organization]. Hai ràng buộc này nêu lên tính chất của tham tố là tác thể hay bị thể của hành động "cấm". Đó chính là cả tác thể và bị thể phải mang tính động (người) hoặc là một tổ chức nào đó. Toán tử "|" thể hiện ý nghĩa "hoặc A hoặc B".

Tiếng Anh và tiếng Việt có nhiều đặc điểm tri nhận giống nhau, tuy nhiên cũng có nhiều đặc điểm khác biệt trong cách tri nhận về sự vật, hiện tượng nói chung và khả năng kết hợp của các sự vật hiện tượng với các động từ hành động. Để đảm bảo tính đồng bộ giữa viVerbNet và VerbNet, luận án đã ánh xạ 75 lớp ngữ nghĩa được sử dụng cho các ràng buộc ngữ nghĩa của VCL với tập hợp 37 ràng buộc lựa chọn vai nghĩa trong VerbNet.

Các ràng buộc vai nghĩa (hay ràng buộc lựa chọn) trong VerbNet tiếng Anh sử dụng được liệt kê trong Bảng 4.5.

Bảng 4.5: Các ràng buộc vai nghĩa trong VerbNet tiếng Anh

abstract	Trừu tượng	int_control	Kiểm soát
animal	Động vật	location	Địa điểm
animate	Tính động	machine	Máy móc
body_part	Bộ phận cơ thể	nonrigid	Không rắn, có thể uống
comestible	Đồ ăn	organization	Tổ chức
communication	Sự giao tiếp	plural	Số nhiều
concrete	Cụ thể	pointy	Vật sắc nhọn
currency	Tiền tệ	refl	Từ phản thân
elongated	Vật thể dài	region	Vùng miền
eventive	Sự kiện	solid	Thuộc tính rắn
force	Lực	sound	Âm thanh
garment	Quần áo	spatial	Hướng trong không gian
human	Con người	substance	Chất lỏng
vehicle	Phương tiện	time	Thời gian
Biotic	Sinh học		

Bên cạnh đó, còn có một số ràng buộc lựa chọn được biểu diễn đồng thời với biểu diễn cú pháp trong khung ngữ nghĩa vị từ, như: dest, dest_conf, dest_dir,

dir, loc, src, path, refl, state và xuất hiện trong dấu “ ”.

Ví dụ: Tôi rót nước từ phích nước vào ấm.

Agent V Theme {+src} Initial_Location +dest_conf Destination

Cách phân biệt dest, dest_conf, dest_dir, dir được mô tả trong Bảng 4.6.

Bảng 4.6: Phân biệt dest, dest_conf, dest_dir, dir.

Tiêu chí	dest	dest_conf	dest_dir	dir
Kết hợp với src (Xuất phát điểm)	-	+	+	-
Hướng tới điểm cuối, đối tượng là người, địa điểm, khái niệm	+	-	+	-
Hướng tới điểm cuối, đối tượng là vật	-	+	-	-

Không chỉ khác biệt trong việc tri nhận các sự vật, hiện tượng mà tiếng Anh và tiếng Việt còn có sự khác biệt trong các tri nhận về không gian. Ví dụ:

`Tiếng Việt: Quả bóng ở trên cây.`

`Tiếng Anh: The ball is in/on the tree.`

Với ví dụ trên, trong tiếng Anh thực tế có thể sử dụng hai giới từ “in (trong)” và “on (trên)”, trong khi tiếng Việt chỉ sử dụng giới từ “trên”. Nếu như sử dụng giới từ “trong” thì phải cung cấp thêm thông tin cho vị trí của quả bóng. Ví dụ: “`Quả bóng ở trong tán cây`”. Sự khác biệt trong việc nhận thức không gian này cũng đóng một vai trò quan trọng trong việc biểu diễn các ràng buộc ngữ nghĩa trong phần khung cú pháp, bên cạnh việc thể hiện trực tiếp ràng buộc ngữ nghĩa cho các vai nghĩa.

Các ràng buộc ngữ nghĩa cũng liên quan đến mức độ ưu tiên, có nghĩa là ràng buộc lựa chọn có thể bao gồm nhiều ràng buộc khác và ngược lại, nhiều ràng buộc lựa chọn có thể cùng nằm trong một ràng buộc có tính bao quát hơn.

Chúng ta có thể xem xét cụ thể ràng buộc ngữ nghĩa Location với 3 ràng buộc con nằm trong nó, cụ thể như sau:

- region, diễn tả cụm giới ngữ biểu thị vùng, ví dụ: “từ dưới đất”
- place, ví dụ: “ở Hà Nội”
- object, ví dụ: “trên bàn”

4.2.3.3. Khung cú pháp và ràng buộc cú pháp

Khung cú pháp trong VerbNet mô tả ngắn gọn cấu trúc bề mặt của các thành phần cấu thành câu. Nó bao gồm các vai nghĩa tương ứng với các tham tố, động

từ chính và các ràng buộc về cú pháp. Mỗi động từ được liên kết với một hoặc nhiều khung cú pháp. Khung cú pháp cũng chỉ định các vai nghĩa xung quanh các động từ và các hạn chế cú pháp thể hiện các ràng buộc đối với các thành phần câu liên quan đến các vai nghĩa này, chẳng hạn như: số nhiều (*plural*), thông báo (*sentential*), ... Ví dụ:

```
Mẹ mắng chúng_tôi
Agent V Patient <+plural>
```

Trong tiếng Việt, một động từ có thể mang nhiều ý nghĩa khác nhau do hiện tượng nói giảm nói tránh, chơi chữ quen thuộc trong cách sử dụng ngôn ngữ của người Việt Nam. Chính vì sự chuyển hóa nghĩa này mà các động từ có thể có nhiều khung cú pháp và đối tố vị ngữ khác nhau. Ví dụ, từ "đi" trong tiếng Việt ngoài nghĩa cơ bản: "di chuyển từ nơi này đến nơi khác" (*) thì còn có một nghĩa nữa là: "rời bỏ cuộc đời, chết" (**). Với nghĩa (*) của động từ "đi", ta có khung cấu trúc và tham tố vị từ tương trùng khớp với khung cấu trúc và tham tố vị trừ của động từ "go" trong tiếng Anh. Ví dụ:

```
Anh ấy đi đến trường.
Mô tả: NP V PP. Destination
Syntax: Agent V to Destination
Semantics: Motion(During(E),Theme) Location(End(E),Destination)
```

Nghĩa (**) của động từ "đi" giống với động từ "chết" trong tiếng Việt, động từ "die" trong tiếng Anh. Ví dụ:

```
Ông ấy đã đi.
Mô tả: NP V
Syntax: Patient V
Semantics: Cause(Causer,E) alive(start(E),Patient) ?alive(result
(E),Patient)
```

viVerbNet được thiết kế sử dụng cách mô tả cú pháp giống như VerbNet. Các giới từ được phép trong mô tả cú pháp được đặt giữa dấu ngoặc nhọn. Sau đây là ví dụ sử dụng và mô tả về khung cú pháp của động từ "đi" theo nghĩa "di chuyển từ nơi này đến nơi khác" (*). Mục động từ này thuộc về cụm động từ "đi, chạy, xuôi" mà luận án đã thực hiện phân cụm có thể được ánh xạ vào một lớp con của lớp động từ "attend" trong VerbNet tiếng Anh. Ví dụ:

```
Tôi đi chợ.
Mô tả: NP V Destination
Syntax: Agent V Destination
```

`Semantics: Motion(During(E),Theme) Location(End(E),Destination)`

Quá trình xây dựng 100 lớp động từ của viVerbNet đã đối chiếu điểm khác biệt ngữ pháp giữa tiếng Anh và tiếng Việt, đồng thời đưa ra các giải pháp và đề xuất sửa đổi các ràng buộc cú pháp sao cho vừa đảm bảo tính đồng bộ giữa VN tiếng Anh và viVerbNet và bám sát theo các đặc điểm của ngữ pháp tiếng Việt. Các đặc điểm phổ biến nhất được mô tả trong các mẫu sau:

1. Agent V Patient <+plural>: Mẫu này thể hiện sự hạn chế về số lượng của vai nghĩa Patient (số nhiều/plural). Trong tiếng Việt, số nhiều được biểu thị bằng các định từ trước danh từ như "những, các, bọn, ..." hoặc bằng các số từ.

2. Pivot V Theme <+np_to_inf>: Mẫu này minh hoạ cho việc sử dụng các động từ nguyên thể. Ví dụ: "Tôi muốn anh ấy đi".

3. Agent V Theme <+sc_ing>: Mẫu này minh hoạ cho việc danh từ hoá các động từ. Trong tiếng Việt, kiểu danh động từ đầu tiên là các động từ chuyển loại thành danh từ được giữ gốc từ, tức là không có sự thay đổi bên trong cấu tạo của từ, hiện tượng này diễn ra ở cả hai ngôn ngữ, ... Ví dụ:

   - Tôi đã thỏa_thuận (V) với anh ấy, "thỏa_thuận" là một động từ;
   - Anh ấy và tôi có hai thỏa_thuận (N), "thỏa_thuận" là danh từ.

4. Pivot V Theme <+ac_ing>: Kiểu danh động từ thứ hai bao gồm thêm một từ chức năng như "sự", "việc", ... nghĩa là "sự kiện" hoặc một loại từ như "cái", "kẻ", ... ở trước động từ đó [6].

   - Kinh_tế nước_nhà phát_triển mạnh ("phát_triển" là một động từ);
   - Sự phát_triển của kinh_tế đã mang lại một bộ_mặt mới cho đất_nước. ("sự phát_triển" tương đương với một danh từ).

Việc biểu diễn các ràng buộc "*_ing" là không cần thiết bởi trong tiếng Việt không có tồn tại hiện tượng chuyển loại động từ thành danh từ bằng cách thêm hậu tố "ing". Tuy vậy, tiếng Việt vẫn nên tồn tại một ràng buộc cú pháp biểu thị cho nội dung chuyển loại từ động từ sang danh từ bằng cách kết hợp các từ như "sự, việc, niềm, cái, nỗi, ..." Chính vì vậy, ràng buộc "*_tonp" được thêm vào cho các danh từ được phái sinh từ động từ. Ví dụ:

`Tôi đã theo dõi sự phát triển của cô ấy`

`Experiencer V Stimulus <+poss_tonp>`

Khi xét về nhóm Động từ nguyên mẫu và Động từ nguyên mẫu có "to", việc đưa ra các phân biệt về "bare_inf" (nguyên mẫu) và "to_inf" (nguyên mẫu có "to") cũng không cần thiết đối với việc biểu diễn về ràng buộc cú pháp tiếng Việt. Ví dụ:

`+ac_bare_inf: He helped bake the cake (Anh ấy giúp nướng bánh)`

`+ac_to_inf: He helped to save the child (Anh ấy giúp cứu đứa_trẻ)`

Ràng buộc thành phần "ac" được hiểu là chủ ngữ là một trong những tác thể thực hiện hành động chính. Tiếp đó, khi xét về ngữ pháp tiếng Việt, sau một động từ có thể là một danh từ/cụm danh từ (bổ sung ý nghĩa về đối tượng bị ảnh hưởng trực tiếp hay gián tiếp của hành động), tính từ/cụm tính từ (bổ sung tính chất của hành động), phụ từ/cụm phụ từ (bổ sung cách thức, tính chất của hành động), cụm giới từ (bổ sung ý nghĩa về nơi chốn, thời gian, công cụ, ...), động từ/cụm động từ và thậm chí là cả một mệnh đề.

Sự xuất hiện của động từ, cụm động từ và mệnh đề ở sau động từ là vị ngữ chính của câu sẽ được luận án chú trọng tới do chúng thường có vai trò làm bổ ngữ bổ sung ý nghĩa cho động từ chính. Do vậy luận án cũng chú trọng việc biểu diễn các ràng buộc cho các phần bổ ngữ này.

Bổ ngữ mệnh đề khuyết chủ ngữ của một vị từ được xác định khi một vị từ hoặc một mệnh đề không có chủ ngữ riêng, nhưng vẫn tồn tại mối quan hệ đối với vị từ chính của câu.

- Động từ trong thành phần bổ ngữ có liên quan đến chủ ngữ của câu. Ví dụ:

  `Tôi thích vẽ tranh`

  `Pivot V Theme <+sc_Verb>`

  Động từ "vẽ" có liên quan tới chủ ngữ "tôi" và chỉ có chủ ngữ của câu tác động lên động từ nằm ở phần vai nghĩa này, phần vai nghĩa này sẽ được luận án gắn nhãn ràng buộc cú pháp "+sc_Verb". Ràng buộc này có sự lưỡng phân với ràng buộc "+ac_Verb" (chủ ngữ không phải là đối tượng duy nhất thực hiện hành động). Ví dụ:

  `Anh ấy giúp nấu cơm.`

  `Agent V Theme <+ac_Verb>`

- Động từ trong phần bổ ngữ không liên quan đến chủ ngữ. Ví dụ:

  `(*) Mẹ cho gà ăn`

Agent V Theme <+oc_Verb>

(**) Mẹ cho ăn

Agent V Theme <+oc_Verb>

Trong câu (*), động từ "ăn" trong phần bổ ngữ mệnh đề khuyết chủ ngữ có liên quan đến thành phần bổ ngữ trực tiếp "gà" của động từ chính "cho" chứ không có liên quan đến chủ ngữ "mẹ". Ở câu (**), không có phần bổ ngữ cho động từ chính của câu "cho" nhưng động từ thứ hai "ăn" vẫn không có liên quan gì tới chủ ngữ chính. Dạng này sẽ sử dụng ràng buộc "+oc_Verb" để ràng buộc cho phần vai nghĩa này. Bổ ngữ mệnh đề, hiểu một cách đơn giản thì bổ ngữ mệnh đề chính là thành phần bổ ngữ có đầy đủ cấu trúc chủ-vị.

Trong VerbNet cũng đã tồn tại một số loại nhãn với bổ ngữ là một mệnh đề như: +for_comp, +wheth_comp, +wh_comp, ... với các chức năng biểu thị về ngữ pháp cũng như ngữ nghĩa khác nhau, vì vậy mà luận án sẽ ánh xạ các ràng buộc này khi biểu diễn ràng buộc cú pháp cho tiếng Việt. Tuy nhiên, khi bàn về các loại nhãn mệnh đề, VerbNet vẫn sử dụng một số nhãn có đặc trưng về thể của động từ, theo quan điểm của luận án, để đảm bảo tính đồng bộ, luận án sẽ chuyển đổi các nhãn như sau: "wh_inf" – "Wh_Verb"; "what_inf" – "What_Verb"; "wheth_inf" – "Wheth_Verb".

Trên đây luận án đã đưa ra một số lý giải về sự khác biệt giữa ngữ pháp tiếng Anh và ngữ pháp tiếng Việt, đồng thời luận án cũng đã đưa ra một số đề xuất sửa đổi ràng buộc cú pháp.

4.2.3.4. Vị từ ngữ nghĩa

Các vị từ ngữ nghĩa biểu thị mối quan hệ giữa tham thể và các sự kiện để biểu thị ý nghĩa chính của câu. Thông tin ngữ nghĩa cho các động từ trong VerbNet được thể hiện dưới dạng kết hợp của các vị từ ngữ nghĩa, chẳng hạn như vị từ ngữ nghĩa chung (chuyển động (*motion*), liên hệ (*contact*), truyền đạt thông tin (*transfer_info*), ...), vị ngữ (*Prep, Adv, và Pred*), vị ngữ cụ thể; vị ngữ cho nhiều sự kiện.

Các vị từ ngữ nghĩa có thể thuộc các loại sau:

- Sự kiện: Biến sự kiện E hoặc một phần con của nó (start(E), during(E), end(E), result(E))

- Hằng: Một tham tố trong một vị ngữ chỉ định một thuộc tính của lớp đó nhưng không phải là một trong các vai trò. Loại tham tố này được sử dụng để cho phép các vị ngữ được sử dụng trong các lớp khác nhau.

- Vai nghĩa: Bao gồm các vai nghĩa có trong khung cú pháp và các vai nghĩa không có trong khung cú pháp cụ thể này nhưng vẫn tồn tại tiềm ẩn trong ngữ nghĩa (đứng sau toán tử ?).

- Động từ cụ thể: Được sử dụng cho các vai nghĩa khác nhau được hình thành bởi các động từ trong lớp.

Các toán tử cũng có thể được thêm vào trong cấu trúc vị từ ngữ nghĩa như sự phủ định (¬), sự vắng mặt (?) của một số vai nghĩa nhất định trong cấu trúc được mô tả.

Một số vị từ được sử dụng để mô tả các giai đoạn khác nhau trong quá trình của một sự kiện: giai đoạn chuẩn bị (Start (E)), giai đoạn diễn ra (During (E)), đỉnh điểm (End (E)) và giai đoạn kết quả (Result (E)) của một sự kiện. Sự thể hiện rõ ràng này giúp mô tả đầy đủ các thành phần ngữ nghĩa cốt lõi cũng như những thay đổi trong cấu trúc sự kiện phức tạp. Trong VerbNet, các vị từ mô tả thời gian có thể xuất hiện hoặc không, phụ thuộc vào ngữ nghĩa của động từ hoặc thậm chí là không xuất hiện đối với những động từ không xuất hiện đặc điểm về quá trình.

Ví dụ: die (Start (E), Result (E)); invest (Start (E)); exchange (Start (E), End (E), During (E)); long, seem (không có vị từ thể hiện quá trình).

VerbNet tiếng Anh sử dụng vị từ path_rel để biểu diễn cho toàn bộ sự kiện có xuất hiện sự thay đổi bao gồm: thay đổi về địa điểm (ch_of_loc), thay đổi sở hữu (ch_of_poss), chuyển đổi thông tin (tr_of_info), thay đổi trạng thái (ch_of_state),...

Ví dụ: become-109.1

`The belt came undone`

`Syntax: Patient V Result`

`Semantics: path_rel(start(E), Initial_State, Patient, ch_of_state, prep) path_rel(result(E), Result, Patient, ch_of_state, prep)`

Đối với thành phần ngữ nghĩa trong viVerbNet, luận án sử dụng cùng một tập hợp các vị từ ngữ nghĩa như VerbNet. Tập hợp vị từ ngữ nghĩa của VerbNet tiếng Anh được sử dụng trong viVerbNet có 153 nhãn vị từ ngữ nghĩa. Tiêu biểu là một số vị từ ngữ nghĩa: appear (xuất hiện); allow (cho phép); avoid (tránh);

belive (tin tưởng); benefit (lợi ích); body_motion (hoạt động của cơ thể); cause (nguyên nhân); sleep (ngủ); .... Các vị từ dùng để mô tả quá trình cũng được sử dụng trực tiếp trong quá trình biểu diễn ngữ nghĩa của động từ.

Ví dụ, khi biểu diễn ngữ nghĩa cho nhóm động từ "confine-92":

```
"We committed John."
Syntax: Agent V Theme
Semantic: path_rel(start(E),Theme,?Initial_Location,ch_of_loc,
prep),path_rel(end(E),Theme,?Destination,ch_of_loc,prep),confined
(result(E),Theme) cause(Agent,E)
```

Các vị từ ngữ nghĩa được sử dụng là:

- Vị từ quá trình: Start (Bắt đầu), End (Kết thúc), Result (Kết quả)
- Các vai nghĩa: Agent (Tác thể), Theme (Tương tác thể) và Destination (Điểm đến), Initial_Location (Điểm xuất phát)
- Các vị từ là các động từ cụ thể biểu thị ý nghĩa của hành động: Confine (Giam giữ)
- Các toán tử: Vai nghĩa Destination và Initial_Location không có hiện diện trong khung cú pháp được đánh dấu bằng toán tử (?) để biểu hiện sự vắng mặt.

Có thể hiểu sự biểu diễn vai nghĩa đó theo một tiến trình như sau: Bắt đầu khoảng thời gian (E) nào đó, John (tương tác thể) ở một địa điểm nào đó. Sau khi kết thúc khoảng thời gian (E), "John" được đưa đến một địa điểm nào đó. Hành động 'confine' (giam) là kết quả của quá trình (E), người bị giam chính là tương tác thể "John". Và người thực hiện, nguyên nhân dẫn đến kết quả đó chính là tác thể "We".

Nói chung, các nhãn vị từ ngữ nghĩa trong VerbNet sẽ được luận án áp dụng vào biểu diễn quan hệ ngữ nghĩa trong các ngữ liệu tiếng Việt.

4.2.4. Công cụ gán nhãn mạng động từ tiếng Việt

Để việc gán nhãn các lớp động từ trong tiếng Việt được đơn giản và đỡ tốn thời gian, công sức, luận án đã thiết kế công cụ gán nhãn mạng động từ tiếng Việt. Công cụ này được xây dựng để có thể sử dụng các kết quả từ các nghiên cứu trước đó, giúp việc gán nhãn các lớp động từ tiếng Việt nhanh gọn và chính xác hơn.

VerbNet cho tiếng Việt cũng sẽ có các thành phần như VerbNet cho tiếng Anh:

- Các lớp động từ (*Verb classes*): Động từ trong VerbNet được nhóm vào các lớp dựa trên các đặc điểm ngữ nghĩa và cú pháp tương tự. Thông tin trong một lớp động từ thường có cấu trúc cây của một lớp động từ đó: các lớp cha và lớp con. Các lớp động từ có thể bao gồm một hoặc nhiều lớp con.
- Thành viên (*Members*): Chứa danh sách các động từ thuộc về một lớp hoặc lớp con cụ thể.
- Vai nghĩa (*Roles*): Vai nghĩa đề cập đến mối quan hệ ngữ nghĩa giữa một vị từ và các tham tố của nó.
- Các ràng buộc lựa chọn (*Selectional Restrictions*): Mỗi vai nghĩa được liệt kê trong một lớp có thể có thêm các ràng buộc lựa chọn nhất định, cung cấp thêm thông tin về bản chất của vai trò đó.
- Khung cú pháp (*Frames*): Phần Khung cú pháp bao gồm các cấu trúc cú pháp, câu ví dụ, và các vai trò ngữ nghĩa được ánh xạ với các tham số cú pháp. Các vị từ ngữ nghĩa cũng được mô tả trong phần này, chỉ ra cách các động từ tham gia vào sự kiện.

Ngoài ra, các động từ ở Verbnet tiếng Việt sẽ được liên kết tới từ điển VCL[8] để có thể tra cứu lớp nghĩa của từng động từ tương ứng, đồng thời có thể tham khảo các ví dụ và cấu trúc cú pháp, vị từ đi kèm.

Các chức năng của công cụ gán nhãn cụm động từ gồm có:

- Đăng nhập, phân quyền (đối với người quản trị và người gán nhãn), phân các lớp động từ để người gán nhãn thực hiện.
- Tạo mới một lớp động từ.
- Chỉnh sửa các thành phần của một lớp động từ như thành phần, vai nghĩa, khung cú pháp, khung vị từ, ...
- Thống kê danh sách các lớp động từ
- Nhập (*import*) dữ liệu từ tệp xml và xuất (*export*) dữ liệu đã gán nhãn ra tệp xml.

[8] `https://vlsp.hpda.vn/demo/?page=vcl`

Công cụ gán nhãn mạng động từ tiếng Việt đã hoàn thành và được đưa vào sử dụng để gán nhãn các cụm động từ. Công cụ này giúp nâng cao chất lượng dữ liệu, tiết kiệm thời gian và công sức của các chuyên gia ngôn ngữ. Các lớp động từ được gán sẽ có cùng một chuẩn (đầu vào, đầu ra), đảm bảo tính nhất quán và chuẩn hoá trong việc gán nhãn. Từ đó tạo ra một tập các lớp động từ chất lượng cao, chính xác và chi tiết, được sử dụng để nâng cao hiệu suất của nhiều hệ thống NLP.

4.3. Ví dụ một cụm động từ trong viVerbNet

Để làm rõ kết quả các cụm động từ trong viVerbNet, mục này sẽ mô tả kĩ các thành phần của cụm động từ "học". Chi tiết về các thành phần và lớp con trong cụm được mô tả trong Bảng 4.7.

Bảng 4.7: Nhóm động từ "học" trong viVerbNet.

STT	**Nhóm động từ**	**Lớp**	**Lớp con**	**Chi tiết**
1	Nghĩa "***học***"	học-1	học-1.1	học, học hành, học tập, luyện tập, luyện, ôn, ôn luyện, ôn tập, tập luyện
			học-1.2	bắt chước, học lỏm, nhái, nhại
			học-1.3	học hỏi, chắt lọc, định hình, lĩnh hội, lãnh hội, rèn giũa, mài giũa, thu nạp, tích lũy, tiếp thụ, trau dồi

4.3.1. Vai nghĩa

Khi biểu diễn vai nghĩa cho các động từ thuộc nhóm động từ mang nghĩa "học", viVerbNet sẽ sử dụng các vai nghĩa mà VerbNet áp dụng cho lớp động từ "learn-14" để biểu diễn các tham tố tham gia vào biểu đạt nội dung ngữ nghĩa trong câu. Đó chính là: đối tượng học (*Agent*), nguồn truyền đạt (*Source*), nội dung (*Topic*).

```
Role: Agent [+animate], Topic, Source
```

4.3.2. Ràng buộc lựa chọn

Về ràng buộc lựa chọn cho vai nghĩa, lớp "learn-14" trong VerbNet sử dụng duy nhất một ràng buộc [+animate] (tính động). Tuy nhiên khi xét với lớp từ "học-1" thu được sau khi phân cụm thì ràng buộc [+tính động] không còn đủ để biểu diễn ràng buộc vai nghĩa để áp dụng vào từng trường hợp biểu diễn của các lớp động từ con trong lớp "học-1".

Khi xét về phân cấp ràng buộc chọn lựa các vai nghĩa, ràng buộc "animate" chứa ràng buộc "animal" và "human". Ràng buộc chung "animate" chỉ thích hợp sử dụng để ràng buộc cho vai Tác thể (Agent) khi biểu diễn ngữ nghĩa cho lớp con "học.1.2" vì các động từ xuất hiện trong nhóm này có tính trung tính, có thể sử dụng được cả người và động vật: "Con vẹt bắt chước tiếng người"; "trẻ em bắt chước người lớn". Không thể nào nói "Con vẹt ôn tập/lãnh hội tiếng người". Chính vì vậy cần phải bổ sung thêm ràng buộc [+human] để ràng buộc vai nghĩa Agent khi biểu diễn ngữ nghĩa của hai lớp con: "học-1.1" và "học-1.3":

```
Role: Agent [+human], Topic, Source
```

4.3.3. Khung cú pháp và ràng buộc cú pháp

Về cú biểu diễn khung cú pháp và ràng buộc cú pháp, luận án kết hợp các thông tin ngữ pháp tiếng Việt và ánh xạ với các khung cú pháp của lớp từ tương đương trong VerbNet để chọn lọc ra các khung cú pháp có thể sử dụng trong viVerbNet.

Khung cú pháp và ràng buộc của lớp "learn-14" và các lớp con của nó được biểu diễn như sau:

- learn-14

  ```
  (1) NP V NP PP. source
  syntax Agent V Topic from Source
  (2) NP V PP. source
  syntax Agent V from Source
  (3) NP V NP
  syntax Agent V Topic
  ```

- learn-14-1

  ```
  (4) NP V
  ```

```
syntax Agent V
```

- learn-14-2

```
(5) NP V that S

syntax Agent V Topic <+that_comp>
```

- learn-14-2-1

```
(6) NP V PP. topic

syntax Agent V of about Topic
```

Ta có thể sử dụng một số khung cú pháp và ràng buộc cú pháp của "learn-14" vào việc mô tả khung cú pháp và ràng buộc cú pháp cho các lớp từ con của lớp "học-1". Khung (1), (2), (3) đều có thể sử dụng để biểu diễn ngữ nghĩa cho các lớp động từ con thuộc lớp "học-1".

Khung (4) ít được sử dụng hơn vì bản chất tiếng Việt ít sử dụng khung cú pháp này vì nội dung thông báo được biểu đạt quan khung cấu trúc đó không trọn vẹn. Khung (5) và khung (6) ít được sử dụng trong tiếng Việt vì nó không tự nhiên trong cả văn nói và văn viết. Vì vậy, luận án không sử dụng hai khung này vào việc mô tả khung ngữ nghĩa vị từ của lớp "học-1".

Bên cạnh đó, tiếng Việt vẫn có sự xuất hiện của một số câu đặc biệt khuyết chủ ngữ, cấu trúc cú pháp này cũng ko được đề cập đến trong hệ thống khung cú pháp của "learn-14". Các lớp con của "học-1" đều sử dụng chung các khung cú pháp:

- Agent V Topic from Source: Tôi học tiếng Anh từ anh trai
- Agent V from Source: Tôi học từ anh trai tôi
- Agent V Topic: Tôi học tiếng Anh
- Agent V: Tôi học
- V Topic: Học tiếng Anh

4.3.4. Vị từ ngữ nghĩa

Đối với lớp nhóm động từ mang nghĩa học, mà cụ thể là lớp "học-1" với các lớp con của nó, luận án sẽ sử dụng các biểu diễn ngữ nghĩa của các khung cú pháp đã được luận án lựa chọn ánh xạ của VerbNet kết hợp với kiểm tra tiêu

chí hợp lý về khả năng biểu diễn ngữ nghĩa động từ tiếng Việt. Các vị từ được sử dụng: vị từ chung "Transfer_infor" và vị từ quá trình "During(E)". Các ngữ nghĩa của lớp "học-1" và các lớp con được biểu thị như sau:

- Agent V Topic from Source: Tôi học tiếng Anh từ anh trai

  `Transfer_info(During(E), Source, Agent, Topic)`

- Agent V from Source: Tôi học từ anh trai tôi

  `Transfer_info(During(E), Source, Agent, ? Topic)`

- Agent V Topic: Tôi học tiếng Anh

  `Transfer_info(During(E), ?Source, Agent, Topic)`

- Agent V: Tôi học

  `Transfer_info(During(E), ?Source, Agent, ?Topic)`

Biểu diễn ngữ nghĩa cho các lớp động từ là một công việc khá phức tạp, tích hợp nhiều thao tác về xác định vai nghĩa, ràng buộc chọn lựa, khung cú pháp, ràng buộc cú pháp và khung vị từ ngữ nghĩa. Tiếng Việt là một ngôn ngữ khó, các yếu tố nghĩa được biểu thị trong các động từ cũng vô cùng đa dạng. Hiện tượng từ đa nghĩa, chuyển loại từ, sự khác biệt về đặc trưng ngôn ngữ, ngữ pháp cũng tạo ra rất nhiều khó khăn trong việc chọn lựa, ánh xạ các thông tin ngữ pháp, biểu diễn ngữ nghĩa vị từ.

Một động từ có thể có một hoặc nhiều tham thể, điều này đồng nghĩa với việc cần có một hay nhiều vai nghĩa tham gia và biểu thị cho các tham thể đó. Căn cứ vào nghĩa của động từ, mà chúng ta có thể thấy tham tố thứ hai của động từ đó có thể đảm nhiệm một số vai nghĩa khác nhau, có thể là vai đích, cũng có thể là vai nguồn. Điều này ảnh hưởng nhiều đến công việc gán nhãn vai nghĩa cho các tham thể.

Các tham thể của các động từ thường là danh từ. Những đặc trưng khác biệt về ngữ pháp của tiếng Việt so với tiếng Anh cũng giúp một phần rút gọn được các ràng buộc về cú pháp trong câu. Đôi khi tham thể thứ hai có thể là một cú, lúc này cần phải ràng buộc cú bằng các khung ràng buộc phù hợp (that_comp, what_comp, ...).

Chi tiết về các cụm động từ được mô tả trong Dữ liệu phân cụm động từ của Luận án[9].

[9] `https://github.com/vietnamesedp/Thesis/tree/main/data`

4.4. Kết luận chương 4

Chương này đã trình bày chi tiết về quá trình xây dựng mạng động từ viVerbNet cho tiếng Việt, một hệ thống nhằm phân loại và nhóm các động từ theo ngữ nghĩa và cú pháp một cách có hệ thống. Trước tiên, việc khảo sát kho từ vựng VCL và VerbNet đã được thực hiện, tiếp theo là quá trình phân cụm các động từ trong VCL và xây dựng được 100 cụm động từ cơ bản cho tiếng Việt. Mỗi cụm động từ đều được phát triển đầy đủ với các thành phần quan trọng như vai nghĩa, khung cú pháp với các ràng buộc cú pháp và ngữ nghĩa, thông tin về vị từ ngữ nghĩa. Mặc dù kết quả này vẫn chưa bao quát hết toàn bộ các động từ tiếng Việt, nhưng đã đặt nền móng quan trọng cho việc xây dựng hệ thống mạng động từ trong tương lai, hỗ trợ cho việc phát triển các mô hình biểu diễn và phân tích ngữ nghĩa sâu hơn trong ngữ cảnh tiếng Việt.

KẾT LUẬN

Luận án tập trung nghiên cứu các phương pháp biểu diễn và phát triển ngữ liệu, công cụ cho bài toán phân tích cú pháp và ngữ nghĩa tiếng Việt. Cụ thể, luận án đã có những đóng góp cơ bản về hai hướng chính:

- Xây dựng các tài nguyên ngôn ngữ, gồm các lược đồ chú giải, hướng dẫn chú giải và kho ngữ liệu có chú giải theo lược đồ đã thiết kế:
  - Cú pháp phụ thuộc: Trên cơ sở tập nhãn cú pháp phụ thuộc tiếng Việt đã xây dựng trong giai đoạn trước, luận án tiến hành cập nhật, thiết kế lại và chỉnh sửa tập nhãn cũng như hướng dẫn chú giải theo phiên bản 2.0 của Dự án cú pháp phụ thuộc phổ quát (*Universal Dependency - UD*). Sau đó, luận án tiến hành thu thập và xây dựng kho ngữ liệu với hơn 9,000 câu (trong đó 3,000 câu đã được tích hợp vào kho UD hồi tháng 11 năm 2022) đã được chú giải theo quy trình chuẩn hóa, với độ đồng thuận gán nhãn đạt 91%, đảm bảo chất lượng và tính nhất quán. Kho ngữ liệu này cùng với tài liệu hướng dẫn chi tiết đã được công khai trên GitHub[10] và sử dụng trong cuộc thi về phân tích cú pháp phụ thuộc tiếng Việt tại Hội thảo Xử lí ngôn ngữ tự nhiên và tiếng nói tiếng Việt (VLSP 2020).
  - Cú pháp thành phần: Kế thừa kho ngữ liệu cú pháp thành phần Vietreebank, luận án thực hiện việc rà soát, cập nhật và chuẩn hoá các nhãn cú pháp thành phần cũng như tài liệu hướng dẫn chú giải để có một bộ nhãn phù hợp với các nghiên cứu đối sánh đa ngữ. Trên cơ sở đó, kho ngữ liệu gồm hơn 9,000 câu đã được cập nhật theo tập nhãn mới với độ đồng thuận của các nhà chú giải lên tới 94% cho thấy kho ngữ liệu được xây dựng tỉ mỉ, chính xác và có độ tin cậy cao. Kho ngữ liệu này đã được công khai và sử dụng trong cuộc thi về phân tích cú pháp thành phần tiếng Việt tại Hội thảo Xử lý ngôn ngữ tự nhiên và tiếng nói tiếng Việt (VLSP 2022 và VLSP 2023).

[10] `https://github.com/vietnamesedp/Thesis`

– Ngữ nghĩa nông (Vai nghĩa): Tập nhãn vai nghĩa cho tiếng Việt xây dựng trước đó đã được cập nhật và chỉnh sửa tương thích với khung chú giải vai nghĩa trong dự án Universal Proposition Bank 2.0. Tập nhãn này đã được sử dụng để xây dựng kho ngữ liệu tiếng Việt có chú giải vai nghĩa gồm 2,570 câu.

– Ngữ nghĩa sâu: Luận án đã thiết kế mô hình biểu diễn ngữ nghĩa cho tiếng Việt dựa trên AMR và tập vai nghĩa LIRICS. Mô hình nổi bật nhờ khả năng thể hiện đặc trưng tiếng Việt như danh từ hóa động từ, ngữ nghĩa thời gian và đồng sở chỉ liên đoạn. Kho ngữ liệu gồm 1.570 câu từ tiểu thuyết Hoàng tử bé được xây dựng theo quy trình chuẩn hóa, với độ đồng thuận cao giữa các chuyên gia[11].

– Mạng động từ tiếng Việt: Sau khi xây dựng các kho ngữ liệu và mô hình phân tích, luận án phát triển mạng động từ tiếng Việt (*viVerbNet*) dựa trên 100 cụm động từ tiêu biểu, với 5 thành phần chính: vai nghĩa, ràng buộc lựa chọn, khung và ràng buộc cú pháp, vị từ ngữ nghĩa. Các lớp động từ được ánh xạ sang VerbNet tiếng Anh, góp phần kết nối tài nguyên ngữ nghĩa song ngữ và hỗ trợ tích hợp tiếng Việt vào hệ thống đa ngữ.

- Về phương pháp và mô hình cho phân tích tiếng Việt, luận án đã thực hiện những công việc sau:

  – Đánh giá, so sánh các mô hình véc-tơ từ huấn luyện sẵn cho tiếng Việt và một số phương pháp hiện đại để cải tiến hiệu quả của bài toán phân tích cú pháp. Cụ thể, với cú pháp thành phần, kết quả tốt nhất đạt $F_1 = 90.15\%$ với mô hình HPSG kết hợp với công cụ gán nhãn Stanza. Đối với cú pháp phụ thuộc, mô hình Deep bi-affine sử dụng PhoBERT đạt $LAS = 78.05\%$ và $UAS = 85.27\%$ - kết quả tốt nhất khi huấn luyện và kiểm thử trên kho ngữ liệu cú pháp phụ thuộc đã được xây dựng.

  – Xây dựng công cụ chuyển đổi giữa cú pháp thành phần và cú pháp phụ thuộc, hỗ trợ quá trình gán nhãn dữ liệu: Thuật toán chuyển đổi từ cú pháp thành phần sang cú pháp phụ thuộc đạt kết quả $LAS = 52.63\%$ và $UAS = 66.20\%$. Với chiều ngược lại, thuật toán chuyển đổi đạt kết quả $F_1 = 80.83\%$. Công cụ này cho phép linh hoạt chuyển đổi giữa hai cách

[11] `https://github.com/vietnamesedp/Thesis/tree/main/MeaningRepresentation`

biểu diễn ngữ pháp, được sử dụng trong việc xây dựng kho ngữ liệu, giúp giảm thời gian và tiết kiệm công sức của các chuyên gia chú giải.

- Phát triển và đánh giá các thuật toán phân cụm động từ tiếng Việt: Các thuật toán phân cụm được thực hiện trên hơn 12,000 nghĩa của động từ trích từ từ điển tiếng Việt cho máy tính VCL, nhằm nhóm các động từ thành các cụm có chung những đặc điểm về ngữ pháp, ngữ nghĩa. Thuật toán phân cụm K-means đã được thử nghiệm chi tiết với nhiều kịch bản khác nhau, bao gồm thay đổi số lượng cụm, các mô hình véc tơ từ và các câu ngữ cảnh của động từ.
- Thử nghiệm áp dụng các mô hình ngôn ngữ lớn như GPT-4 và Gemini cho nhiệm vụ gán nhãn vai nghĩa và phân tích ngữ nghĩa tiếng Việt. Kết quả cho thấy các mô hình đạt độ chính xác từ 47% đến 58% trên cả hai bài toán, phản ánh tiềm năng ứng dụng của các mô hình này trong xử lý ngôn ngữ tự nhiên. Những kết quả này không chỉ minh chứng cho hiệu quả ban đầu của các mô hình ngôn ngữ lớn đối với tiếng Việt, mà còn giảm thiểu chi phí và công sức gán nhãn thủ công, đồng thời thúc đẩy quá trình phát triển các ứng dụng ngôn ngữ thông minh trong thực tiễn.

Trong tương lai, các hướng phát triển của luận án tập trung vào:

- Tiếp tục phát triển và mở rộng các kho ngữ liệu ngữ nghĩa, chia sẻ và công bố rộng rãi các tài nguyên này trong cộng đồng xử lý ngôn ngữ tiếng Việt: Luận án sẽ tiếp tục mở rộng quy mô và phạm vi của kho ngữ liệu gán nhãn ngữ nghĩa cho tiếng Việt, bổ sung thêm nhiều dạng ngữ nghĩa phức tạp và ngữ cảnh sử dụng.

- Phát triển tiếp mạng động từ viVerbNet và tăng cường khả năng liên thông liên ngữ: Luận án tiếp tục hoàn thiện viVerbNet bằng cách mở rộng lớp động từ, tinh chỉnh vai nghĩa, khung cú pháp và ràng buộc lựa chọn để nâng cao độ chính xác và tính ứng dụng. Việc ánh xạ với VerbNet tiếng Anh cũng được điều chỉnh nhằm xử lý khác biệt ngôn ngữ, tăng khả năng tích hợp tiếng Việt vào các hệ thống đa ngữ.

- Tinh chỉnh và khai thác các mô hình ngôn ngữ lớn cho các bài toán cú pháp và ngữ nghĩa tiếng Việt: Luận án sẽ tiếp tục nghiên cứu và cải tiến các phương pháp khai thác mô hình ngôn ngữ lớn (LLMs) như GPT-4, Gemini,

. . . cho các tác vụ phân tích cú pháp và ngữ nghĩa. Việc tinh chỉnh các mô hình này trên dữ liệu tiếng Việt và tích hợp với các tài nguyên ngôn ngữ đã được xây dựng sẽ giúp nâng cao chất lượng gán nhãn, giảm thiểu chi phí thủ công, đồng thời mở rộng khả năng ứng dụng của các mô hình trong các hệ thống thực tế như dịch máy, tóm tắt văn bản, hỏi đáp và trợ lý ảo.

- Cải thiện chất lượng kho ngữ liệu:
  - Đánh giá và điều chỉnh sơ đồ chú giải để tăng tính nhất quán và khả năng sử dụng lại. Xây dựng bộ tiêu chí đánh giá chất lượng chú giải và đề xuất các công cụ hỗ trợ hiệu chỉnh bán tự động.
  - Tiếp tục khảo sát và phân tích các hiện tượng ngôn ngữ đặc thù của tiếng Việt trong ngữ liệu, từ đó đề xuất các biểu diễn ngữ nghĩa phù hợp hơn cho các cấu trúc cú pháp–ngữ nghĩa độc đáo của tiếng Việt.

Tóm lại, luận án đã có những đóng góp quan trọng trong việc xây dựng một hệ thống tài nguyên ngôn ngữ tiếng Việt phong phú, đa dạng và được chú giải ở mức độ sâu, tập trung vào kho từ vựng động từ cùng các ngữ liệu chú giải cú pháp – ngữ nghĩa. Đồng thời, luận án cũng đã tiến hành nghiên cứu, phát triển và đánh giá các công cụ phân tích cú pháp và ngữ nghĩa tiên tiến, được tối ưu hóa phù hợp với đặc thù của tiếng Việt. Việc công bố rộng rãi các tài nguyên và công cụ này không chỉ góp phần thúc đẩy sự phát triển bền vững và lâu dài của lĩnh vực xử lý ngôn ngữ tự nhiên tiếng Việt, mà còn tạo nền tảng vững chắc cho các hướng nghiên cứu tiếp theo trong tương lai.

DANH MỤC CÔNG TRÌNH CÔNG BỐ

[P1] **Ha My Linh**, Nguyen Thi Minh Huyen, "*A Case Study on Meaning Representation for Vietnamese*", Proceedings of the First International Workshop on Designing Meaning Representations, pages 148-153, 2019, Italy, (*ISBN 978-1-950737-45-1*).

[P2] **Ha My Linh**, Nguyen Thi Minh Huyen, Vu Xuan Luong, Nguyen Thi Luong, Phan Thi Hue, Le Van Cuong, "*VLSP 2020 shared task: Universal dependency parsing for Vietnamese*", Proceedings of the 7th International Workshop on Vietnamese Language and Speech Processing, pages 77-83, 2020, Vietnam.

[P3] **Ha My Linh**, Le Van Cuong, Nguyen Thi Minh Huyen, "*Construction of a VerbNet style Lexicon for Vietnamese*", Proceedings of the 34th Pacific Asia Conference on Language, Information and Computation, pages 84-91, 2020, Vietnam, (*ISSN 2619-7782*).

[P4] **Ha My Linh**, Do Duy Dao, Nguyen Thi Minh Huyen, Tran Thu Trang, "*Using rules for building Vietnamese AMR-based corpus*", Một số vấn đề chọn lọc của công nghệ thông tin và truyền thông lần thứ XXIV, pages: 547-552, 2021, Vietnam.

[P5] Ishan Jindal, ALexandre Rademaker, Michał ULewicz, **Ha My Linh**, Huyen Nguyen, Khoi-Nguyen Tran, Huaiyu Zhu, and Yunyao Li, "*Universal Proposition Bank 2.0*". In Proceedings of the Thirteenth Language Resources and Evaluation Conference, pages 1700 – 1711, 2022, MarseilLe, France, European Language Resources Association, (*ISBN: 979-10-95546-72-6*).

[P6] **Ha My Linh**, Do Duy Dao, Nguyen Thi Minh Huyen, Ngo The Quyen, Doan Xuan Dung, "*VLSP 2021 - NER Challenge: Named Entity Recognition for Vietnamese*", VNU Journal of Science: Computer Science and Communication Engineering, pages 87 - 97, Vol 38.1, 2022, (*VNU Journal of Science ISSN: 0866-8612*).

[P7] The Quyen Ngo, Thi Anh Phuong Nguyen, **Ha My Linh**, Thi Minh Huyen Nguyen, Phuong Le-Hong, "*Improving Multi-label Classification of Similar Languages by Semantics-Aware Word Embeddings*", In Eleventh Workshop on NLP for Similar Languages, Varieties and DiaLects (VarDial 2024), pages 235-240, Mexico, Association for Computational Linguistics, (*ISBN 979-8-89176-104-9*).

[P8] **Ha My Linh**, Thi Minh Huyen Nguyen, The Quyen Ngo, Tuan Thanh Le, Tran Thai Dang, Viet Hoang Ngo, Xuan Dung Doan, Thi Luong Nguyen, Van Cuong Le, Thi Hue Phan, Xuan Luong Vu, "*VLSP 2022 Challenge: Vietnamese Constituency Parsing*", **accepted** to Journal of Computer Science and Cybernetics, 2025, (*ISSN 2815-5939*).

[P9] **Ha My Linh**, Pham Thi Duc, Le Ngoc Toan, Thi Minh Huyen Nguyen, "*An Investigation of ISO-TimeML Applied to Vietnamese*", Proceedings of the 38th Pacific Asia Conference on Language, Information, and Computation, PACLIC 38 (2024), Tokyo, Japan, pages 1387 - 1394.

[P10] 1 bài tạp chí quốc tế đang nộp và chờ phản biện:

(a) **Ha My Linh**, Doan Xuan Dung, Nguyen Thi Luong, Nguyen Thi Minh Huyen, Le Hong Phuong, "*Dependency parsing for Vietnamese*", **submitted** to Language Resources and Evaluation (LRE), 2023.

TÀI LIỆU THAM KHẢO

Tài liệu tiếng Việt

[1] Lâm Nhựt Khang, Võ Lê Minh Trung, Nguyễn Huỳnh Hữu Đức (2017), *Xây dựng WordNet cho tiếng Việt*, FAIR, Đà Nẵng, Việt Nam, trang 1007-1014.

[2] Hà Mỹ Linh, Nguyễn Thị Lương, Nguyễn Việt Hùng, Nguyễn Thị Minh Huyền, Lê Hồng Phương, Phan Thị Huê (2014), *Xây dựng kho ngữ liệu mẫu có gán nhãn vai nghĩa cho tiếng Việt*, Hội thảo quốc gia lần thứ XVII: Một số vấn đề chọn lọc của Công nghệ thông tin và truyền thông, Đắk-lăk, Việt Nam, trang 409-414.

[3] Nguyễn Lê Minh, Hoàng Thị Điệp, and Trần Mạnh Kế (2008), *Nghiên cứu luật hiệu chỉnh kết quả dùng phương pháp MST phân tích cú pháp phụ thuộc tiếng việt*, Kỷ yếu Hội thảo ICT.rda'08, trang 258-267.

[4] Nguyễn Minh Thuyết, Nguyễn Văn Hiệp (1998), *Thành phần câu tiếng Việt*, Nxb Đại học Quốc gia Hà Nội, Hà Nội.

[5] Nguyễn Thiện Giáp (2014), *Nghĩa học Việt ngữ*, Nhà xuất bản Giáo dục Việt Nam, 2014, 327 trang.

[6] Nguyễn Thị Bích Ngoan (2013), *So sánh đối chiếu hiện tượng danh hoá động từ trong tiếng Việt và tiếng Anh*, Tạp chí Khoa học Đại học Sư phạm TPHCM, trang 13–22.

Tài liệu tiếng Anh

[7] Abend, Omri and Rappoport, Ari (2013), *Universal Conceptual Cognitive Annotation (UCCA)*, Proceedings of ACL 2013 - 51st Annual Meeting of the Association for Computational Linguistics, Vol. 1, pp. 228–238.

[8] Abend Omri and Rappoport Ari (2017), *The State of the Art in Semantic Representation*, Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, Vol. 1, pp. 77–89.

[9] Abeillé, Anne and Clément, Lionel and Toussenel, François (2003), *Building a Treebank for French*, in Treebanks: Building and Using Parsed Corpora, edited by Anne Abeillé, Springer Netherlands, Dordrecht, pp. 165–187, ISBN: 978-94-010-0201-1, DOI: 10.1007/978-94-010-0201-1_10.

[10] Akbik, Alan and Chiticariu, Laura and Danilevsky, Marina and Li, Yunyao and Vaithyanathan, Shivakumar and Zhu, Huaiyu (2015), *Generating High Quality Proposition Banks for Multilingual Semantic Role Labeling*, Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Vol. 1: Long Papers), pp 397–407.

[11] Artstein Ron and Poesio Massimo (2008), *Survey Article: Inter-Coder Agreement for Computational Linguistics*, Computational Linguistics, Vol. 34, No. 4, pp 555–596.

[12] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin (2017), *Attention Is All You Need*, 31st Conference on Neural Information Processing Systems (NIPS 2017), Long Beach, CA, USA.

[13] Baker, C. F., Fillmore, C. J., and Lowe, J. B. (1998), *The Berkeley FrameNet Project*, 36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics, Vol. 1, pp 86–90.

[14] Banarescu, Laura, Bonial, Claire, Cai, Shu, Georgescu, Madalina, Griffitt, Kira, Hermjakob, Ulf, Knight, Kevin, Koehn, Philipp, Palmer, Martha, and

Schneider, Nathan (2013), *Abstract Meaning Representation for Sembanking*, Proceedings of the 7th Linguistic Annotation Workshop and Interoperability with Discourse, pp. 178–186.

[15] Bob Carpenter (1997), *Type-logical semantics*, MIT Press, Cambridge.

[16] Bojanowski, Piotr and Grave, Edouard and Joulin, Armand and Mikolov, Tomas (2017), *Enriching Word Vectors with Subword Information*, Transactions of the Association for Computational Linguistics, Vol. 5, pp. 135–146.

[17] Brown Peter F. and Della Pietra Vincent J. and deSouza Peter V. and Lai Jenifer C. and Mercer Robert L. (1992), *Class-Based n-gram Models of Natural Language*, Computational Linguistics, Vol 18, No. 4, pp 467–480.

[18] Brown, Susan and Dligach, Dmitriy and Palmer, Martha (2011), *VerbNet class assignment as a WSD task*, In *Proceedings of the 47th Annual Meeting of the Association for Computational Linguistics*, pp. 85–94.

[19] Cai, Shu and Knight, Kevin (2013), *Smatch: an Evaluation Metric for Semantic Feature Structures*, Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers), Sofia, Bulgaria, Association for Computational Linguistics, pp. 748–752.

[20] Candito, Marie and Amsili, Pascal and Barque, Lucie and Benamara, Farah and de Chalendar, Gaël and Djemaa, Marianne and Haas, Pauline and Huyghe, Richard and Mathieu, Yvette Yannick and Muller, Philippe and Sagot, Benoît and Vieu, Laure (2014), *Developing a French FrameNet: Methodology and First results*, Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14), pp 1372–1379.

[21] Carl Pollard and Ivan A. Sag (1994), *Head-Driven Phrase Structure Grammar*, The University of Chicago Press, Chicago.

[22] Carreras Xavier and Màrquez Lluís (2004), *Introduction to the CoNLL-2004 Shared Task: Semantic Role Labeling*, in *Proceedings of the Eighth Conference on Computational Natural Language Learning (CoNLL-2004) at HLT-NAACL 2004*, Association for Computational Linguistics, pp 89–97.

[23] Carreras Xavier and Màrquez, Lluís (2005), *Introduction to the CoNLL-2005 Shared Task: Semantic Role Labeling*, in Proceedings of the Ninth Conference on Computational Natural Language Learning (CoNLL-2005), Association for Computational Linguistics, pp 152–164.

[24] Choe, Hyonsu and Han, Jiyoon and Park, Hyejin and Oh, Tae Hwan and Kim, Hansaem (2020), *Building Korean Abstract Meaning Representation Corpus*, Proceedings of the Second International Workshop on Designing Meaning Representations, pp. 21–29.

[25] Clusmann Jan and Kolbinger Fiona and Muti Hannah and Carrero Zunamys and Eckardt Jan-Niklas and Laleh Narmin and Löffler Chiara and Schwarzkopf Sophie-Caroline and Unger Michaela and Veldhuizen, Gregory and Wagner, Sophia and Kather, Jakob (2023), *The future landscape of large language models in medicine*, Communications medicine, Vol. 3, No. 1, pp 1–8, doi: https://doi.org/10.1038/s43856-023-00370-1.

[26] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu (2020), *Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer*, Journal of Machine Learning Research, Vol. 21, No. 140, pp. 1–67.

[27] Conneau Alexis and Khandelwal Kartikay and Goyal Naman and Chaudhary Vishrav and Wenzek Guillaume and Guzman Francisco and Grave Edouard and Ott Myle and Zettlemoyer Luke and Stoyanov Veselin (2020),

Unsupervised Cross-lingual Representation Learning at Scale, Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, pp 8440–8451.

[28] Crystal, D. (1997), *A dictionary of linguistics and phonetics*, 4th edition, Cambridge, MA: Blackwell Publishing.

[29] Dat Quoc Nguyen, Dai Quoc Nguyen, Son Bao Pham, Phuong Thai Nguyen, and Minh Le Nguyen (2014), *From Treebank Conversion to Automatic Dependency Parsing for Vietnamese*, In: Métais, E., Roche, M., Teisseire, M. (eds) Natural Language Processing and Information Systems. NLDB 2014. Lecture Notes in Computer Science, vol 8455. Springer, Cham, pp. 196–207.

[30] Dat Quoc Nguyen, Mark Dras, and Mark Johnson (2016), *An empirical study for Vietnamese dependency parsing*, Proceedings of the 14th Annual Workshop of the Australasian Language Technology Association, ALTA 2016, pp 143–149.

[31] Dat Quoc Nguyen and Nguyen Anh Tuan (2020), *PhoBERT: Pre-trained language models for Vietnamese*, Findings of the Association for Computational Linguistics: EMNLP 2020, pp. 1037–1042.

[32] Day, William H. E. and Edelsbrunner, Herbert (1984), *Efficient algorithms for agglomerative hierarchical clustering methods*, *Journal of Classification*, Vol. 1, pp 7-24.

[33] Danlos Laurence and Nakamura Takuya and Pradet Quentin (2014), *Vers la création d'un VerbeNet du français*, Atelier FondamenTAL, TALN 2014, July, pp 103–108.

[34] Deng Jiawen and Zubair Areeba and Park Yejean (2023), *Limitations of large language models in medical applications*, Postgraduate Medical Journal, pp 1298-1299.

[35] Devlin, Jacob, Chang, Ming-Wei, Lee, Kenton, and Toutanova, Kristina (2018), *BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding*, NAACL-HLT 2019, pp 4171-4186.

[36] Diem Truong, Duc-Thuan Vo, and Uyen Trang Nguyen (2017), *Vietnamese Open Information Extraction*, In Proceedings of the 8th International Symposium on Information and Communication Technology (SoICT '17), Association for Computing Machinery, New York, NY, USA, pp 135–142, https://doi.org/10.1145/3155133.3155171.

[37] Dorr, Bonnie J. (1997), *Large-Scale Dictionary Construction for Foreign Language Tutoring and Interlingual Machine Translation*, Machine Translation, Vol. 12, No. 4, pp 271–322.

[38] Dozat Timothy and Manning Christopher D. (2017), *Deep Biaffine Attention for Neural Dependency Parsing*, In 5th International Conference on Learning Representations, ICLR 2017, Toulon, France, Conference Track Proceedings, OpenReview.net.

[39] Emily M. Bender, Batya Friedman (2018), *Data Statements for Natural Language Processing: Toward Mitigating System Bias and Enabling Better Science*, Transactions of the Association for Computational Linguistics, Vol. 6, pp. 587–604.

[40] Elvis Saravia, Hsien-Chi Toby Liu, Yen-Hao Huang, Junlin Wu, and Yi-Shin Chen (2018), *CARER: Contextualized Affect Representations for Emotion Recognition*, In Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, pp. 3687–3697, Brussels, Belgium.

[41] Fellbaum, C. (Ed.). (1998), *WordNet: An Electronic Lexical Database*, MIT Press, Cambridge, MA. ISBN: 978-0-262-06197-1.

[42] Francis, W. Nelson, and Kucera, Henry (1979), *Manual of Information to Accompany a Standard Corpus of Present-Day Edited American English, for Use with Digital Computers*, Brown University.

[43] Gemini Team and Rohan Anil and Sebastian Borgeaud and Jean-Baptiste Alayrac and Jiahui Yu and Radu Soricut, et al. (2024), *Gemini: A Family of Highly Capable Multimodal Models*, arXiv, url=https://arxiv.org/abs/2312.11805.

[44] Giuglea, Ana-Maria and Moschitti, Alessandro (2006), *Semantic Role Labeling via FrameNet, VerbNet and PropBank*, Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics, pp. 929–936.

[45] Gil Francopoulo, Monte George, Nicoletta Calzolari, Monica Monachini, Nuria Bel, Mandy Pet, Claudia Soria (2006), *Lexical Markup Framework (LMF)*, Proceedings of the Fifth International Conference on Language Resources and Evaluation (LREC'06), pp. 577–580.

[46] Hajič Jan and Ciaramita, Massimiliano and Johansson, Richard and Kawahara, Daisuke and Martí, Maria Antònia and Màrquez, Lluís and Meyers, Adam and Nivre, Joakim and Padó, Sebastian and Štěpánek, Jan and Straňák, Pavel and Surdeanu, Mihai and Xue, Nianwen and Zhang, Yi (2009), *The CoNLL-2009 shared task: syntactic and semantic dependencies in multiple languages*, in Proceedings of the Thirteenth Conference on Computational Natural Language Learning: Shared Task, Association for Computational Linguistics, pp 1–18.

[47] Hao Shibo and Gu Yi and Ma Haodi and Hong Joshua and Wang Zhen and Wang Daisy and Hu Zhiting (2023), *Reasoning with Language Model is Planning with World Model*, in Proceedings of the 2023 Conference on

Empirical Methods in Natural Language Processing, Association for Computational Linguistics, pp. 8154–8173.

[48] Hartigan, J. A. and Wong, M. A. (1979), *Algorithm AS 136: A K-means clustering algorithm*, Applied Statistics, Royal Statistical Society, pp. 100–108.

[49] Hensman, Svetlana and Dunnion, John (2004), *Automatically building conceptual graphs using VerbNet and WordNet*, Proceedings of the Intenational Symposium on Information and Communication Technologies, Las Vegas, Nevada, USA, June 16-18, 2004, pp 115–120.

[50] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova (2019), *BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding*, In Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), pp 4171–4186.

[51] James Pustejovsky and José M. Castaño and Robert Ingria and Roser Sauri and Robert J. Gaizauskas and Andrea Setzer and Graham Katz and Dragomir R. Radev (2003), *TimeML: Robust Specification of Event and Temporal Expressions in Text*, In Mark T. Maybury (ed.), *New Directions in Question Answering, Papers from 2003 AAAI Spring Symposium*, Stanford University, Stanford, CA, USA, pp 28–34.

[52] Jay Earley (1970), *An efficient context-free parsing algorithm*, Communications of the ACM, 13(2), pp. 94–102. DOI: `https://doi.org/10.1145/362007.362035`

[53] Jiangming Liu and Yue Zhang (2016), *Shift-Reduce Constituent Parsing with Neural Lookahead Features*, Transactions of the Association for Computational Linguistics, Vol 5, pp 45–58, `http://arxiv.org/abs/1612.00567`

[54] Jiangming Liu and Yue Zhang (2017), *In-Order Transition-based Constituent Parsing*, *Transactions of the Association for Computational Linguistics*, Vol. 5, MIT Press, Cambridge, MA, pp. 413–424.

[55] Jiang Peng and Cai Xiaodong (2024), *A Survey of Semantic Parsing Techniques*, Symmetry, Vol. 16, No. 9, Article 1201.

[56] Jinqi Lai and Wensheng Gan and Jiayang Wu and Zhenlian Qi and Philip S. Yu (2024), *Large language models in law: A survey*, AI Open, Vol. 5, pp 181–196.

[57] Ji Tao, Wu Yuanbin, and Lan Man (2019), *Graph-based Dependency Parsing with Graph Neural Networks*, Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, pp. 2475–2485, Florence, Italy.

[58] Johan Bos (2013), *The Groningen Meaning Bank*, in Proceedings of the Joint Symposium on Semantic Processing. Textual Inference and Structures in Corpora, edited by Octavian Popescu and Alberto Lavelli, Trento, Italy, Vol. 2, pp 463–496, `https://aclanthology.org/W13-3802`.

[59] Johnson, Tim (1984), *Natural language computing: the commercial applications*, The Knowledge Engineering Review, Vol. 1, No. 3, pp. 11–23.

[60] Jurafsky, D. and H. Martin (2000), *Speech and language processing: An introduction to natural language processing*, Computational linguistics, and speech recognition, New Delhi, India: Pearson Education.

[61] Kamath Aishwarya and Das Rajarshi (2019), *A Survey on Semantic Parsing*, Automated Knowledge Base Construction (AKBC), `https://openreview.net/forum?id=HylaEWcTT7`

[62] Kawahara, Daisuke and Palmer, Martha (2014), *Single Classifier Approach for Verb Sense Disambiguation based on Generalized Features*, In *Proceed-*

ings of the Ninth International Conference on Language Resources and Evaluation (LREC'14), pp. 4210–4213.

[63] Kasami Tadao (1965), *An Efficient Recognition and Syntax-Analysis Algorithm for Context-Free Languages*, Coordinated Science Laboratory, University of Illinois at Urbana-Champaign, `https://api.semanticscholar.org/CorpusID:61491815`.

[64] Kiperwasser Eliyahu and Goldberg Yoav (2016), *Simple and Accurate Dependency Parsing Using Bidirectional LSTM Feature Representations*, Transactions of the Association for Computational Linguistics, Vol. 4, pp 313–327. `https://aclanthology.org/Q16-1023`.

[65] Kiem-Hieu Nguyen (2018), *BKTreebank: Building a Vietnamese Dependency Treebank*, Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018), Miyazaki, Japan, European Language Resources Association (ELRA), pp 2164-2168.

[66] Kiet Van Nguyen and Nguyen, Ngan Luu-Thuy (2015), *Error Analysis for Vietnamese Dependency Parsing*, In 2015 Seventh International Conference on Knowledge and Systems Engineering (KSE), pp 79–84, `http://arxiv.org/abs/1911.03724`.

[67] Kingsbury, P., Palmer, M. (2002), *From TreeBank to PropBank*, Proceedings of the Third International Conference on Language Resources and Evaluation (LREC'02), pp. 1–8.

[68] Kipper, K., Korhonen, A., Ryant, N., Palmer, M. (2006), *Extending VerbNet with Novel Verb Classes*, Proceedings of the Fifth International Conference on Language Resources and Evaluation (LREC'06), pp. 1–8.

[69] K. Gaurav and S. Sebastian and V. Sowmya and R. Siva (2024), *Scope Ambiguities in Large Language Models*, Transactions of the Association for Computational Linguistics, Shanghai, China, Vol 12, pp 738–754.

[70] Klavans, Judith L. and Kan, Min-Yen (1998), *Role of Verbs in Document Analysis*, Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics, Montreal, Quebec, Canada, Association for Computational Linguistics, pp 680–686.

[71] Lam Hoang Thanh and Gabriele Picco and Yufang Hou and Young-Suk Lee and Lam M. Nguyen and Dzung T. Phan and Vanessa López and Ramon Fernandez Astudillo (2021), *Ensembling Graph Predictions for AMR Parsing*, Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, pp 8495–8505.

[72] Levin, B. (1993), *English Verb Classes and Alternations: A Preliminary Investigation*, Chicago Press, University.

[73] Li, Bin and Wen, Yuan and Qu, Weiguang and Bu, Lijun and Xue, Nianwen (2016), *Annotating the Little Prince with Chinese AMRs*, Proceedings of the 10th Linguistic Annotation Workshop held in conjunction with ACL 2016 (LAW-X 2016), pp. 7–15.

[74] Linh, Ha and Nguyen, Huyen (2019), *A Case Study on Meaning Representation for Vietnamese*, Proceedings of the First International Workshop on Designing Meaning Representations, pp. 148–153.

[75] Linh Ha, Do Dao, Nguyen Huyen, Ngo Quyen, and Doan Dung (2022), *VLSP 2021 - NER Challenge: Named Entity Recognition for Vietnamese*,

VNU Journal of Science: Computer Science and Communication Engineering, vol. 38, no. 1.

[76] Liang, Percy (2013), *Lambda Dependency-Based Compositional Semantics*, CoRR, Vol. abs/1309.4408.

[77] Liu Yinhan and Ott Myle and Goyal Naman and Du Jingfei and Joshi Mandar and Chen Danqi and Levy Omer and Lewis Mike and Zettlemoyer Luke and Stoyanov Veselin (2019), *RoBERTa: A Robustly Optimized BERT Pretraining Approach*, CoRR, abs/1907.11692, https://openreview.net/forum?id=SyxS0T4tvS.

[78] Loper, Edward and Yi, Szu-ting and Palmer, Martha (2007), *Combining lexical resources: Mapping between PropBank and VerbNet*, In Proceedings of the IWCS-7.

[79] McDonald, Ryan and Nivre, Joakim (2011), *Analyzing and Integrating Dependency Parsers*, *Computational Linguistics*, Vol. 37(1), pp 197–230.

[80] Margaret Mitchell, Simone Wu, Andrew Zaldivar, Parker Barnes, Lucy Vasserman, Ben Hutchinson, Elena Spitzer, Inioluwa Deborah Raji, Timnit Gebru (2018), *Model Cards for Model Reporting*, In Proceedings of the Conference on Fairness, Accountability, and Transparency (FAT*'19). Association for Computing Machinery, New York, USA, pp 220–229. https://doi.org/10.1145/3287560.3287596.

[81] Marie-Catherine de Marneffe, Christopher D. Manning, Joakim Nivre, Daniel Zeman (2021), *Universal Dependencies*. Computational Linguistics 2021, Vol. 47 (2), pp. 255–308, doi: https://doi.org/10.1162/coli_a_00402

[82] Matthias Feurer, Aaron Klein, Katharina Eggensperger, Jost Tobias Springenberg, Manuel Blum, and Frank Hutter (2015), *Efficient and robust automated machine learning*, In Proceedings of the 29th International Confer-

ence on Neural Information Processing Systems - Volume 2 (NIPS'15), Vol. 2. MIT Press, Cambridge, MA, USA, pp. 2755–2763.

[83] Marie-Catherine de Marneffe and Christopher D. Manning (2008), *The Stanford Typed Dependencies Representation*, Proceedings of the Workshop on Cross-Framework and Cross-Domain Parser Evaluation, Coling 2008, pp. 1-8, Coling 2008 Organizing Committee, Manchester, UK.

[84] M Kay. (1986), *Algorithm schemata and data structures in syntactic processing*, Readings in natural language processing, Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, pp. 35–70.

[85] Ma, Xuezhe and Hu, Zecong and Liu, Jingzhou and Peng, Nanyun and Neubig, Graham and Hovy, Eduard (2018), *Stack-Pointer Networks for Dependency Parsing*, Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pp. 1403–1414.

[86] Miller, G. A., Beckwith, R., Fellbaum, C., Gross, D., and Miller, K. J. (1990), *Introduction to WordNet: An On-line Lexical Database*, International Journal of Lexicography, Vol. 3, No. 4, pp. 235–244.

[87] Mikolov Tomas and Chen Kai and Corrado Greg and Dean Jeffrey, *Efficient Estimation of Word Representations in Vector Space*, In 1st International Conference on Learning Representations, ICLR 2013, Scottsdale, Arizona, USA, May 2-4, 2013, Workshop Track Proceedings.

[88] Mitchell P. Marcus, Mary Ann Marcinkiewicz, Beatrice Santorini (1993), *Building a Large Annotated Corpus of English: The Penn Treebank*, Comput. Linguist., Vol. 19(2), pp. 313–330.

[89] Nivre Joakim, de Marneffe, Marie-Catherine, Ginter Filip, Goldberg Yoav, Hajič Jan, Manning Christopher D., McDonald, Ryan, Petrov, Slav, Pyysalo, Sampo, Silveira, Natalia, Tsarfaty, Reut, Zeman, Daniel (2016),

Universal Dependencies v1: A Multilingual Treebank Collection. In Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16), Portorož, Slovenia, May 2016. European Language Resources Association (ELRA), pp. 1659–1666, `https://aclanthology.org/L16-1262`.

[90] Nivre, Joakim and de Marneffe, Marie-Catherine and Ginter, Filip and Hajič, Jan and Manning, Christopher D. and Pyysalo, Sampo and Schuster, Sebastian and Tyers, Francis and Zeman, Daniel (2020), *Universal Dependencies v2: An Evergrowing Multilingual Treebank Collection*, Proceedings of the Twelfth Language Resources and Evaluation Conference, pp. 4034–4043.

[91] Nguyen, Thi Luong, Ha, My Linh, Nguyen, Viet Hung, Nguyen, Thi Minh Huyen, and Le, Hong Phuong (2013), *Building a treebank for Vietnamese dependency parsing*, Proceedings of the 2013 RIVF International Conference on Computing & Communication Technologies - Research, Innovation, and Vision for Future (RIVF), pp. 147-151.

[92] Nguyen, Luong, Hà, Linh, Nguyen, Huyen, and Phuong, Le-Hong (2018), *Using BiLSTM in dependency parsing for Vietnamese*, *Computacion y Sistemas*, 22, pp. 853–862.

[93] Nguyen Luong Tran and Duong Minh Le and Dat Quoc Nguyen (2022), *BARTpho: Pre-trained Sequence-to-Sequence Models for Vietnamese*, Proceedings of the 23rd Annual Conference of the International Speech Communication Association.

[94] Nguyen, T. M. H., Romary, L., Rossignol, M., and Vu, X. L. (2006), *A lexicon for Vietnamese language processing*, *Language Resources and Evaluation*, 40, pp. 291–309.

[95] Nguyen Huyen T M and Nguyen Hung V and Ngo Quyen T and Vu Luong X and Tran Vu Mai and Ngo Bach X and Le Cuong A (2013), *VLSP Shared task: Sentiment Analysis*, Journal of Computer Science and Cybernetics, Vol. 34, pp. 295–310.

[96] Nguyen, Thai Phuong and Pham, Van-Lam and Nguyen, Hoang-An and Vu, Huy-Hien and Tran, Ngoc-Anh and Truong, Thi-Thu-Ha (2016), *A Two-Phase Approach for Building Vietnamese WordNet*, Proceedings of the 8th Global WordNet Conference (GWC), edited by Christiane Fellbaum, Piek Vossen, Verginica Barbu Mititelu, and Corina Forascu, pp. 261–266.

[97] OpenAI (2024), *GPT-4 Technical Report*, CoRR, Vol. abs/2303.08774.

[98] Petukhova, Volha and Bunt, Harry (2008), *LIRICS Semantic Role Annotation: Design and Evaluation of a Set of Data Categories*, Proceedings of the Sixth International Conference on Language Resources and Evaluation (LREC'08), European Language Resources Association (ELRA), Marrakech, Morocco.

[99] Percy Liang, Michael I. Jordan, and Dan Klein (2013), *Learning Dependency-Based Compositional Semantics*, Computational Linguistics, Vol. 39(2), pp. 389–446.

[100] Pennington, Jeffrey and Socher, Richard and Manning, Christopher (2014), *GloVe: Global Vectors for Word Representation*, Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), edited by Alessandro Moschitti, Bo Pang, and Walter Daelemans, pp. 1532–1543.

[101] Phuong Thai Nguyen, Luong Vu Xuan, Thi Minh Huyen Nguyen, Van Hiep Nguyen, and Phuong Le-Hong (2009), *Building a Large Syntactically-*

Annotated Corpus of Vietnamese, In Proceedings of the Third Linguistic Annotation Workshop (LAW III), Suntec, Singapore, pp. 182–185.

[102] Hong, Phuong, Pham, Hoang, Pham, Khoai, Nguyen, Huyen, Nguyen, Luong, and Nguyen, Hiep (2017), *Vietnamese Semantic Role Labelling*, VNU Journal of Science: Computer Science and Communication Engineering.

[103] Pradet, Quentin and de Chalendar, Gaël and Desormeaux Baguenier, Jeanne (2014), *WoNeF, an improved, expanded and evaluated automatic French translation of WordNet*, Proceedings of the Seventh Global Wordnet Conference (GWC2014), pp. 32–39.

[104] Qi Peng and Zhang Yuhao and Zhang Yuhui and Bolton Jason, and Manning Christopher D. (2020), *Stanza: A Python Natural Language Processing Toolkit for Many Human Languages*, Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations, pp. 101-108.

[105] Ralph M. Weischedel and Eduard H. Hovy and Mitchell P. Marcus and Martha Palmer (2017), *OntoNotes : A Large Training Corpus for Enhanced Processing*, Handbook of Natural Language Processing and Machine Translation: DARPA Global Autonomous Language Exploitation, Vol. 3, No. 3, pp. 3-4.

[106] Reppen, Randi and Ide, Nancy and Suderman, Keith (2005), *American National Corpus (ANC)*, Linguistic Data Consortium.

[107] Rousseeuw, P. J. (1987), *Silhouettes: A graphical aid to the interpretation and validation of cluster analysis*, *Journal of Computational and Applied Mathematics*, Vol. 20, pp. 53–65.

[108] Shi, Lei and Mihalcea, Rada (2005), *Putting Pieces Together: Combining FrameNet, VerbNet and WordNet for Robust Semantic Parsing*, Computational Linguistics and Intelligent Text Processing, vol. 3406, pp. 100–111.

[109] Srinivasan Iyer (2019), *Learning to Map Natural Language to General Purpose Source Code*, Thesis of Doctor of Philosophy, University of Washington.

[110] Shui Ruihao and Cao Yixin and Wang Xiang and Chua Tat-Seng (2023), *A Comprehensive Evaluation of Large Language Models on Legal Judgment Prediction*, in Findings of the Association for Computational Linguistics: EMNLP 2023, Association for Computational Linguistics, pp. 7337–7348.

[111] Strubell, Emma, Verga, Patrick, Belanger, David, and McCallum, Andrew (2018), *Linguistically-informed self-attention for semantic role labeling*, Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, pp. 5027–5038.

[112] Tai, Kai Sheng and Socher, Richard and Manning, Christopher D (2015), *Improved semantic representations from tree-structured long short-term memory networks*, in Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing, Vol. 1, pp. 1556–1566.

[113] Tesnière, Lucien (1959), *Eléments de Syntaxe Structurale*, Klincksieck.

[114] Timnit Gebru, Jamie Morgenstern, Briana Vecchione, Jennifer Wortman Vaughan, Hanna M. Wallach, Hal Daumé III, Kate Crawford (2018), *Datasheets for Datasets*, Communications of the ACM, 64. 10.1145/3458723.

[115] Tran, Tri, Pham, T., Ngo, Hung, Dien, Dinh, and Collier, Nigel (2007), *Named Entity Recognition in Vietnamese documents*, Progress in Informatics, pp. 5–13.

[116] Van-Nhat Nguyen, Ha-Thanh Nguyen, Dinh-Hieu Vo, and Le-Minh Nguyen (2018), *Relation Extraction in Vietnamese Text via Piecewise Convolution Neural Network with Word-Level Attention*, in *5th NAFOSTED Conference on Information and Computer Science (NICS)*, IEEE, pp. 99–103.

[117] Van-Hai Vu, Quang-Phuoc Nguyen, Kiem-Hieu Nguyen, Joon-Choul Shin, and Cheol-Young Ock (2020), *Korean-Vietnamese Neural Machine Translation with Named Entity Recognition and Part-of-Speech Tags*, IEICE Transactions on Information and Systems, Vol. E103.D, No. 4, pp. 866–873.

[118] Van Gysel Jens, Meagan Vigus, Jayeol Chun, Kenneth Lai, Sarah Moeller, Jiarui Yao, Tim O'Gorman, Andrew Cowell, William Croft, Chu-Ren Huang, Jan Hajic, James H. Martin, Stephan Oepen, Martha Palmer, James Pustejovsky, Rosa Vallejos, and Nianwen Xue. (2021), *Designing a uniform meaning representation for natural language processing*, KI - Künstliche Intelligenz, Vol. 35(3), pages: 343–360.

[119] Son Vu Xuan, Thanh Vu, Son Tran, and Lili Jiang (2019), *ETNLP: A Visual-Aided Systematic Approach to Select Pre-Trained Embeddings for a Downstream Task*, In Proceedings of the International Conference on Recent Advances in Natural Language Processing (RANLP 2019) Varna, Bulgaria, pp. 1285–1294,

[120] Wang Shan and Bond Francis (2013), *Building the Chinese Open Wordnet (COW): Starting from Core Synsets*, Proceedings of the 11th Workshop on Asian Language Resources, pp. 10–18.

[121] Wein, Shira, Donatelli, Lucia, Ricker, Ethan, Engstrom, Calvin, Nelson, Alex, Harter, Leonie, and Schneider, Nathan (2022), *Spanish Abstract Meaning Representation: Annotation of a General Corpus*, Northern European Journal of Language Technology, Vol. 8.

[122] White, Aaron Steven, Reisinger, Drew, Sakaguchi, Keisuke, Vieira, Tim, Zhang, Sheng, Rudinger, Rachel, Rawlins, Kyle, and Van Durme, Benjamin (2016), *Universal Decompositional Semantics on Universal Dependencies*, Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, Austin, Texas, pp. 1713–1723.

[123] Woods, William A. (1973), *Progress in Natural Language Understanding: An Application to Lunar Geology*, AFIPS National Computer Conference, AFIPS Conference Proceedings, Vol. 42, pp. 441–450.

[124] Xuan, Thao, Kawazoe, Ai, Dien, Dinh, Collier, Nigel, and Tri, Tran (2007), *Construction of a Vietnamese Corpora for Named Entity Recognition*, In Large Scale Semantic Access to Content (Text, Image, Video, and Sound) (RIAO '07), Paris, FRA, pp. 719–724.

[125] Xue Naiwen and Xia, Fei and Chiou, Fu-Dong and Palmer, Marta (2005), *The Penn Chinese TreeBank: Phrase structure annotation of a large corpus*, Natural Language Engineering, Vol. 11, No. 2, pp. 207–238, DOI: 10.1017/S135132490400364X.

[126] Yanshan Wang, Sijia Liu, Majid Rastegar-Mojarad, Liwei Wang, Feichen Shen, Fei Liu, and Hongfang Liu (2017), *Dependency and AMR Embeddings for Drug-Drug Interaction Extraction from Biomedical Literature*, Proceedings of the 8th ACM International Conference on Bioinformatics, Computational Biology, and Health Informatics, pp. 36–43.

[127] Yang Kaiyu, and Deng Jia (2020), *Strongly Incremental Constituency Parsing with Graph Neural Networks*, Proceedings of the 34th International Conference on Neural Information Processing Systems, NIPS'20, pp. 1820–1831.

[128] You Liping and Liu Kaiying (2005), *Building Chinese FrameNet database*, 2005 International Conference on Natural Language Processing and Knowledge Engineering, pp. 301–306.

[129] Y. Shuguang and C. Feipeng and Y. Yiming and Z. Zude (2024), *A Study on Semantic Understanding of Large Language Models from the Perspective of Ambiguity Resolution*, in Proceedings of the 2023 International Joint Conference on Robotics and Artificial Intelligence, Association for Computing Machinery, No. 6, pp 165-170.

[130] Yu Zhang and Houquan Zhou and Zhenghua Li (2020), *Fast and Accurate Neural CRF Constituency Parsing*, Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence (IJCAI-20), pp 4046–4053.

[131] Zhou Junru, and Zhao Hai (2019), "Head-Driven Phrase Structure Grammar Parsing on Penn Treebank," Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, pp. 2396–2408, Florence, Italy.

[132] Zhou, Jie and Xu, Wei (2015), *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 (Volume 1: Long Papers), pp 1127–1137.

PHỤ LỤC

So sánh tập nhãn phụ thuộc tiếng Việt với tập nhãn phụ thuộc phổ quát (UD).

STT	Nhãn tiếng Việt	Nhãn UD	Ý nghĩa
1	acl	acl	Định ngữ là mệnh đề.
2	acl:subj		Định ngữ là mệnh đề: danh từ là chủ ngữ.
3	acl:tonp		Danh từ hoá.
4	acl:tmod		Định ngữ là mệnh đề cho danh từ thời gian.
5	acl:relcl	acl:relcl	Định ngữ là mệnh đề quan hệ.
6	acomp	xcomp	Bổ ngữ tính từ.
7	advmod	advmod	Phụ ngữ.
8	advmod:neg	advmod	Phụ từ phủ định.
9	advmod:adj		Phụ ngữ gốc là tính từ.
10	advmod:dir		Phụ ngữ chỉ hướng.
11	advcl	advcl	Trạng ngữ mệnh đề.
12	advcl:objective		Trạng ngữ mệnh đề mục đích.
13	amod	amod	adjectival modifier
14	aux	aux	Trợ động từ tình thái.
15	aux:pass	aux:pass	Trợ từ bị động của động từ chính.
16	appos	appos	Định ngữ dạng chêm nghĩa tương đương.
17	appos:nmod		Định ngữ dạng chêm nghĩa bổ sung.
18	case	case	Giới từ trước danh ngữ.
19	cc	cc	Liên từ đẳng lập.
20	ccomp	ccomp	Bổ ngữ mệnh đề.
21	conj	conj	Quan hệ liên hợp
22	cop		Hệ từ.
23	clf	clf	Danh từ chỉ loại.
24	compound	compound	Từ ghép.
25	compound:adj		Từ ghép tính từ với tính từ.
26	compound:amod		Từ ghép danh từ với tính từ.
27	compound:apr		Từ ghép tính từ với trợ từ/phụ từ.
28	compound:dir		Từ ghép động từ với động từ chỉ hướng.
29	compound:verbnoun		Từ ghép động từ với danh từ.
30	compound:redup	compound:redup	Từ ghép dạng láy.
31	compound:prt	compound:prt	Từ ghép động từ với trợ từ.
32	compound:pron		Từ ghép danh từ với đại từ.
33	compound:svc	compound:svc	Từ ghép động từ với động từ.
34	compound:vmod		Từ ghép danh từ với động từ.
35	compound:atov		Từ ghép tính từ với động từ.
36	compound:Z		Từ ghép với yếu tố Z.
37	csubj	csubj	Mệnh đề là chủ ngữ của câu.

38	csubj:asubj		Chủ ngữ tính từ.
39	csubj:vsubj		Chủ ngữ động từ.
40	csubj:pass	csubj:pass	Chủ ngữ mệnh đề bị động.
41	dep	dep	Quan hệ phụ thuộc chung.
42	det	det	Từ hạn định.
43	det:pmod		Định ngữ hạn định đại từ.
44	det:clf		Định ngữ là danh từ chỉ loại.
45	discourse	discourse	Tình thái từ.
46	dislocated	dislocated	Thành phần bị di chuyển.
47	expl	expl	Hư từ là thành phần chêm vào.
48	fixed	fixed	Ngữ cố định.
49	flat	flat	Tổ hợp từ.
50	flat:date		Tổ hợp từ ngày tháng.
51	flat:redup		Tổ hợp từ láy.
52	flat:number		Tổ hợp từ số lượng.
53	flat:foreign	flat:foreign	Từ nước ngoài.
54	flat:time		Tổ hợp từ thời gian
55	flat:name	flat:name	Tổ hợp từ trong tên riêng.
56	iobj	iobj	Bổ ngữ gián tiếp.
57	list	list	Quan hệ liệt kê.
58	mark	mark	Kết từ trước mệnh đề.
59	mark:pcom	mark:pcomp	Kết từ trước mệnh đề mục đích.
60	nmod	nmod	Định ngữ danh từ.
61	nmod:poss	nmod:poss	Định ngữ danh từ sở hữu.
62	nsubj	nsubj	Chủ ngữ danh từ.
63	nsubj:nn		Chủ ngữ danh từ trong trường hợp vị từ là danh từ.
64	nsubj:pass	nsubj:pass	Chủ ngữ danh từ bị động.
65	nsubj:xsubj	nsubj:xsubj	Chủ ngữ của xcomp là bổ ngữ.
66	nummod	nummod	Đinh ngữ số lượng.
67	obj	obj	Bổ ngữ trực tiếp.
68	obl	obl	Trạng ngữ.
69	obl:about		Bổ ngữ trả lời "về cái gì".
70	obl:adj		Bổ ngữ danh từ cho tính từ.
71	obl:adv		Danh từ phụ cho phó từ.
72	obl:agent	obl:agent	Bổ ngữ là chủ thể trong cấu trúc bị động.
73	obl:comp		Bổ ngữ có giới từ khác.
74	obl:iobj		Trạng ngữ đích đến trong động từ trao tặng.
75	obl:tmod	obl:tmod	Bổ ngữ thời gian.
76	obl:with		Bổ ngữ trả lời "với ai".
77	parataxis	parataxis	Thành phần đẳng lập.
78	punct	punct	Dấu câu.
79	remnant		Truy vết tỉnh lược.
80	reparandum	reparandum	Quan hệ sửa chữa.
81	root	root	Gốc.
82	vocative	vocative	Quan hệ xưng hô.
83	xcomp	xcomp	Bổ ngữ mệnh đề khuyết.
84	xcomp:adj		Bổ ngữ mệnh đề cho tính từ.